Reports Distribution
File Copy

AD 678567

Gulf General Atomic
Incorporated

P.O. Box 608, San Diego, California 92112

GAMD-8497
Category A

AN EULERIAN METHOD FOR CALCULATING STRENGTH DEPENDENT DEFORMATION

PART THREE

Work done by:

J. K. Dienes
M. W Evans
L. J. Hageman
W. E. Johnson
J. M. Walsh

Report written by:

J. K. Dienes
M. W Evans
L. J. Hageman
W. E. Johnson
J. M. Walsh

DDC
DEC 3 1968

This document, which was prepared primarily for internal use at Gulf General Atomic, may contain preliminary or incomplete data. It is informal and is subject to revision or correction; it does not, therefore, represent a final report.

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

Advanced Research Projects Agency
ARPA Order No. 71-62
Ballistic Research Laboratories
Contract No. DA-04-495-AMC-1481(X)
GGA Project 6003

February 2, 1968

Reproduced by the
CLEARINGHOUSE
for Federal Scientific & Technical
Information Springfield Va. 22151

176

AN EULERIAN METHOD FOR CALCULATING STRENGTH DEPENDENT DEFORMATION

by

J. K. Dienes, M. W. Evans, L. J. Hageman,
W. E. Johnson, and J. M. Walsh

PART THREE

THE FORTRAN IV PROGRAM AND INSTRUCTIONS FOR ITS USE

by

J. K. Dienes, M. W. Evans, L. J. Hageman,
W. E. Johnson, and J. M. Walsh

TABLE OF CONTENTS

1. INTRODUCTION 1

2. DESCRIPTION OF THE PROGRAM 3

2.1 RPM 3

2.2 INPUT 3

2.3 CARDS 7

2.4 SETUP 7

2.5 CDT 7

2.6 ES 8

2.7 EDIT 8

2.8 MAP 10

2.9 PH1 10

2.10 PH3 10

2.11 PH2 10

2.12 REZONE 10

2.13 ERROR 11

2.14 TAPE DUMPS 11

3. SET-UP OF PROBLEMS 12

3.1 Definition of RPM Input Variables 13

3.1.1 Identification 13
3.1.2 Dimensions 14
3.1.3 Projectile 15
3.1.4 Target 15
3.1.5 Sphere 16
3.1.6 Filler 16
3.1.7 Physical Constants 16
3.1.8 Calculational Constants 17
3.1.9 Output 21
3.1.10 Stopping 22

3.2 Format and order of OIL-RPM Input Cards 22

3.3 OIL-RPM Input for "Special Setups" 23

4. SAMPLE PROBLEM 27

4.1 OIL-RPM Output 32

4.2 UNIVAC 1108 Control Cards 41

TABLE OF CONTENTS (Continued)

5. RESTART OF PROBLEMS 42

6. LISTING OF THE FORTRAN IV PROGRAM 45

7. DICTIONARY 138

APPENDIX A 169

APPENDIX B 171

1. INTRODUCTION

In PART ONE of the OIL-RPM report the equations of motion were derived from first principles and discussed from a fundamental point of view. PART TWO then described the difference equations, the equation of state, the constitutive equations and the differencing scheme for velocity gradients used in formulating the FORTRAN IV program. Here, in PART THREE, the computer program is listed in detail and the general information necessary to set up and run problems is provided. A dictionary of terms used in the program is also included, as well as a number of tables and charts describing the program in outline form. Forms for setting up a sample problem are also given. This information provides the user with a capability to run his own problems and, in addition, the intent is to describe the code in enough detail that any changes the user may require can be readily made.

The program is the result of contributions by quite a number of people whose names are cited in the references. Briefly, the techniques described here originated from the PIC particle-in-cell method developed at Los Alamos by Evans and Harlow(1) and programmed in machine language. The code was rewritten in FORTRAN and modified for application in the ORION program by Johnson(2) and the General Atomic version was called SHELL. At the suggestion of B. E. Freeman a continuous version was developed by Walsh and Johnson(3) for the solution of hypervelocity impact problems and called OIL. It made use of the equation of state programmed by Tillotson(4). To develop OIL, the particles were replaced by a continuously varying mass in each cell. The capability to run multi-material problems could not be conveniently retained (the particles in PIC could be of different materials). However, the cost of running problems was significantly reduced and the flow profiles were substantially smoother, making it practical to run impact and explosion problems out to a time when the shock pressure was down to a few kilobars. The lowest pressure that could be resolved with PIC was several hundred kilobars.

In order to compute crater size from an impact directly, the shear strength of the cratered material had to be accounted for. The necessary changes to OIL were made in an experimental way by Johnson, Walsh, and Dienes in 1965. This modified version of OIL was called OIL-RPM.(5)

Subsequently, when it had been shown that crater size could be calculated with satisfactory accuracy, the program was streamlined by the authors so it could be used in production runs, and additional editing features were incorporated. The current version has been used in calculating over 50 different problems and the results have generally compared well with experiments, analytical solutions and other hydrodynamic codes.

2. DESCRIPTION OF THE PROGRAM

The equations of motion are integrated in Phases 1, 2 and 3 which account for the effect of pressure, transport, and shear stresses respectively. In addition to these subroutines, ten others have been introduced for the various peripheral tasks. These include INPUT, SETUP, CARDS, CDT (computes time step), EDIT, MAP (provides displays), ES (equation of state), REZONE and ERROR. "RPM" is used to denote both the routine controlling the main flow and the entire program; the choice is generally made clear by the context. A summary of the subroutines is given in Table 1 in the order in which they appear in the listing provided in Section 6. Included in the table are the names of the subroutines calling and called by each of the others. A few general comments are made in the paragraphs below on each of the subroutines.

2.1. RPM

The overall flow of the program is controlled by RPM, as shown in the flow chart of Fig. 1. RPM controls whether additional information is printed at intermediate phases of the calculation cycle for diagnostic purposes and debugging. The variable "INTER" controls these intermediate prints. When INTER = 0, no intermediate prints are made. When INTER $\neq$ 0 EDIT is called and on print cycles EDIT prints are made after PH1 and PH3 as well as after CDT. Details of the Phase 2 calculation are obtained in addition to the EDIT prints by putting INTER = 7, which causes printing of the energy and mass transported as each cell is processed. For debugging of Phase 3 difficulties one puts INTER = 99 and thereby obtains detailed prints of stresses, strain rates and a few other parameters. These options should be used with extreme caution since an intermediate print uses considerable paper.

2.2. INPUT

Instructions for running problems are interpreted by INPUT, which can either start or restart a calculation. It calls SETUP and CARDS, as necessary, to prescribe the initial conditions and to read the input deck. A flow chart showing the relation of INPUT, CARDS, and SETUP is provided in Fig. 2.

TABLE 1

ORDER OF SUBROUTINES

Name	Called From	Calls
RPM	--	INPUT, CDT, EDIT PH1, PH2, PH3
INPUT	RPM	SETUP, CARDS
CARDS	INPUT, SETUP	--
SETUP	INPUT	CARDS, ERROR
CDT	RPM, EDIT	ES, ERROR
ES	CDT	--
EDIT	RPM, ERROR	MAP, REZONE, ERROR, CDT
MAP	EDIT	--
PH1	RPM	--
PH3	RPM	--
PH2	RPM	ERROR
REZONE	EDIT	--
ERROR	SETUP, CDT EDIT, PH2	EDIT

FLOW OF THE CONTROL SUBROUTINE "RPM"

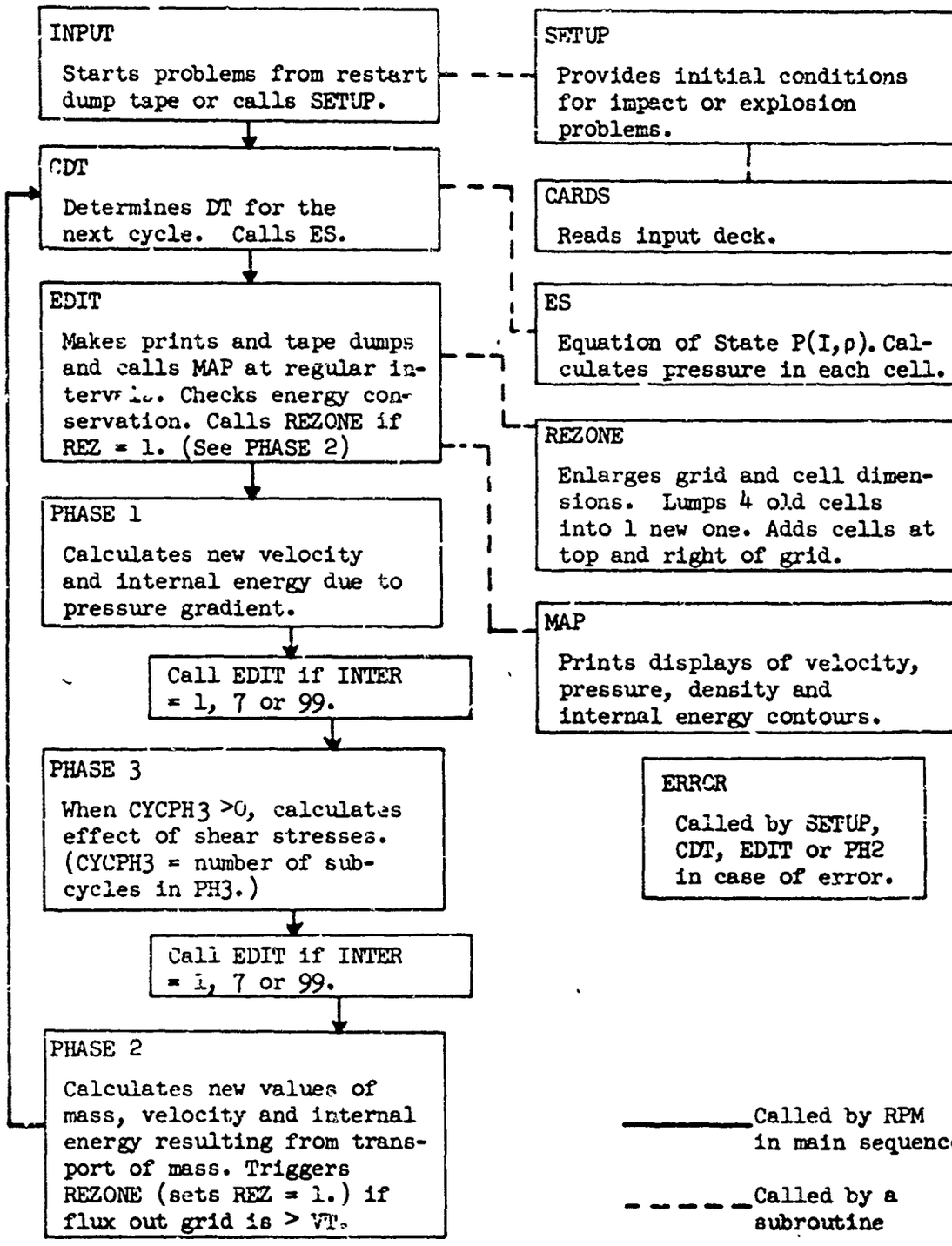


Fig. 1--Flow chart of main sequence

FLOW DIAGRAM OF INPUT, CARDS AND SETUP

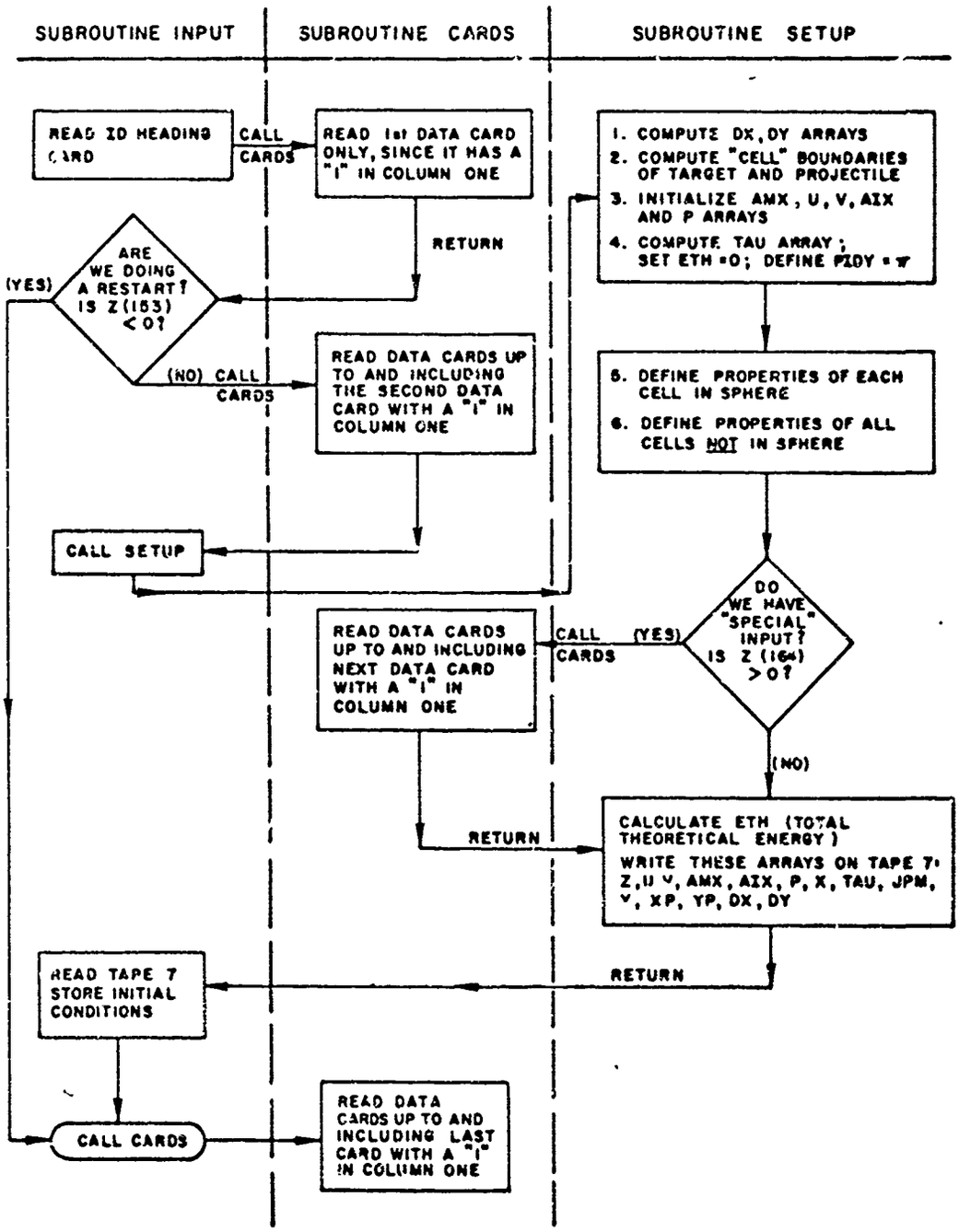


Fig. 2

2.3. CARDS

The reading of the input deck is called by this simple subroutine. Details are given in Fig. 2.

2.4. SETUP

The initial conditions for running problems are generated by SETUP. The most commonly used options are those in which a sphere or a cylinder (called the projectile) hits a target of finite thickness. The target of infinite thickness is an important special case. Another option is a geometry in which a sphere hits a thin plate (called the projectile) which has a filler behind it (typically void or of underdense "foamy" material) and then hits a second thin plate behind the filler. Further details are given in Section 3 describing how to set up various problems.

In an earlier report[3] describing the OIL code a description and FORTRAN listing of a special generator code, CLAM, was given. This program provides a very general method for specifying initial conditions and for setting up OIL-RPM problems. A few changes in CLAM are required to make the write statements for the CLAM dump tape compatible with the READ statements in OIL-RPM.

2.5. CDT

The principal function of this subroutine is to compute a time step which ensures stability of the finite difference equations. This is done by finding the minimum of D/w for all the cells. Here D denotes the minimum of the radial and axial cell dimensions, and w denotes the maximum of the radial velocity, axial velocity and sound speed. For vaporized material sound speed is computed by $\sqrt{\gamma p/\rho}$ and for solid materials by the approximate relation $C = C_0 + \bar{B}\sqrt{p}$ where p is the pressure in the cell. The coefficient $\bar{B}$ is obtained by determining a typical slope for the isentropes in Ref. 4 and using the relation

$$C = V\sqrt{-dp/dV}$$

to evaluate $\bar{B}$ at a particular point. The pressure array is updated in CDT by calling the equation of state subroutine ES.

The array JPM(I) which determines the location of the pressure maximum and hence the cells on which deviator stresses act is also computed in CDT.

Unrealistic behavior in free surface cells containing a small amount of mass would occur, in the absence of a special treatment, when a neighboring cell has a high pressure. This causes large accelerations of the mass in the cell containing a free surface, and eventually velocities which are physically unrealistic. This difficulty is alleviated by reducing the pressure computed by the equation of state by a factor which is the ratio of the smallest mass in an adjacent cell to the mass in the cell itself. This factor reduces the pressure at the interface to a value which accounts for the position of the free surface.

2.6. ES

The equation of state subroutine is called by CDT to evaluate the pressure as a function of density and internal energy. The general method was originally described in Ref. 4, but a number of modifications have been made since that report was written. A general discussion is given in Section 4 of PART TWO. Values of the parameters for a number of materials are listed in Table 2. Parameters for some materials not given in Ref. 4 are supplied in Refs 6 and 7.

2.7. EDIT

The pressure, velocities, density, specific internal energy, and mass for each cell are displayed by EDIT in a "long" print. It also prints out the total internal energy, kinetic energy, axial and radial momentum and mass above and below a dividing line which is the top of the cell whose J index is JPROJ. The changes in energy due to evaporation and losses out the boundaries are also accounted for in the EDIT prints. The crater depth is calculated by "packing down" material in each column to its normal density. This describes in a rough way the extent of the crater even while it still contains low density material.

"Short" EDIT prints display the integrated quantities, and the cell variables for the one column of cells that is adjacent to the axis. These

TABLE 2

EQUATION OF STATE CONSTANTS FOR SEVERAL MATERIALS

	a	b	A dynes/cm^2	B dynes/cm^2	E_0 ergs/g	α	β	E_s	ρ g/cm^3	E'_s	
W	.5	1.04	3.08 x 10^{12}	2.5 x 10^{12}	.225 x 10^{12}	10	10	1.11 x 10^{10}	19.17	5.6 x 10^{10}	Ref. 4
Cu	.5	1.5	1.39	1.1	.325	5	5	1.38	8.9	6.9	
Fe	.5	1.5	1.28	1.05	.095	5	5	2.44	7.9	10.2	
Al	.5	1.63	.75	.65	.05	5	5	3.0	2.7	15.0	
Be	.55	.62	1.7	.55	.175	5	5	10.0	1.8	46.0	
Ti	.5	.60	1.03	.5	.07	5	5	3.5	4.5	12.5	
Ni	.5	1.33	1.91	1.5	.09	5	5	2.85	8.9	9.4	
Mo	.5	1.02	2.71	1.65	.045	5	5	2.8	10.2	9.0	
Th	.4	.86	.53	.5	.025	9	.88	2.0	11.7		
Granite*	.5	1.3	.18	.18	.16	5	5	3.5	2.7	.18 x 10^{12}	Ref. 1
Andesite*	.5	1.3	.18	.18	.15	5	5	3.5	2.7	.15	
Wet Tuff	.5	1.3	.10	.06	.11	5	5	3.2	2.0	.16	
Dry Tuff	.5	1.3	.045	.03	.06	5	5	3.2	1.7	.18	
Oil Shale	.5	1.0	.28	.11	.11	5	5	3.2	2.3	.15	
Dolomite	.5	.5	.85	.30	.10	5	5	2.5	2.8	.14	
Limestone	.5	.6	.4	.67	.10	5	5	2.5	2.7	.14	
Halite	.5	.6	.25	.30	.05	5	5	2.0	2.2	.15	
CH_2	.6	2.0	.075	.024	.07	10	5	2.4	.9	18.0	Ref. 3
Pb	.4	2.4	.466	.0025	.02	10	2	.26	11.3	.97	

*These fits include additional parameters

short prints require only a few pages of printing, and hence are normally called for at more frequent intervals than the "long" prints.

Tracer points are positioned at the center of every other cell in SETUP, and the positions of these points are updated in PH2. The current position of each tracer point is printed in both the long and the short prints, providing the basis for a Lagrangian description of the flow. The positions are written on the restart dump tape and can be used by plot routines to make a plot of material deformation.

2.8. MAP

This sburoutine is called by EDIT and displays the properties of each cell in the active grid using an alphabetic scale. One obtains contour maps of the density, pressure, radial and axial velocities, and internal energy in the active grid.

2.9. PH1

The effect of the pressure gradient in updating the velocities and the internal energy is computed here. The calculation method is described in detail in Section 3.2 of PART TWO.

2.10. PH3

The deviator stresses acting on each cell edge and the hoop stress are determined in PH3 and the resulting velocity and energy increments are computed. Details are given in Section 5 of PART TWO. If CYCPH3 is -1, Phase 3 is bypassed and the effect of strength is not accounted for in the calculation. In this case the code is "hydrodynamic" in the classical sense.

2.11. PH2

Mass transport and the associated flux of momentum and energy are accounted for in PH2. The tracer points are also moved with velocities obtained by a simple weighting scheme.

2.12. REZONE

The masses of four cells are lumped into one in this subroutine. The JPM, DX, DY, X, Y, and TAU arrays are adjusted accordingly. Momentum and total energy are conserved, thereby converting some kinetic energy into

internal in a process loosely called "thermalizing." Every other tracer point is deleted in rezone, and new tracer points are placed in the added cells, retaining constant the total number of tracer points.

2.13. ERROR

This subroutine, which is called in the case of certain error conditions tested on by the code, prints a message identifying the error conditions, calls EDIT for a long print and tape dump, and then calls EXIT.

2.14. TAPE DUMPS

Each OIL-RPM tape dump consists of eight or nine records depending on whether tracer points are used. (See list below.) The first record contains three words: 555.0, the value of CYCLE at the time of the dump, and the value of N3 (N3 = 1 when tracer points are used; otherwise, N3 = 0). The last record also contains three words: 666.0, 666.0, 666.0. However, before each dump after cycle 0, TAPE 7 is backspaced one record and this last record is written over. Therefore, this last record remains only on the last dump of a run and in that case is followed by an end of file mark.

On cycle 0, after all input cards but the last have been read and the properties of all cells have been defined, SETUP does a tape dump. Thereafter, all tape dumps are made by EDIT at set intervals defined by NDUMP7. However, when NODUMP $\neq$ 0. all tape dumps after cycle 0 are suppressed. This makes it possible to restart a problem from a dump tape without writing on the dump tape which is sometimes useful in special studies.

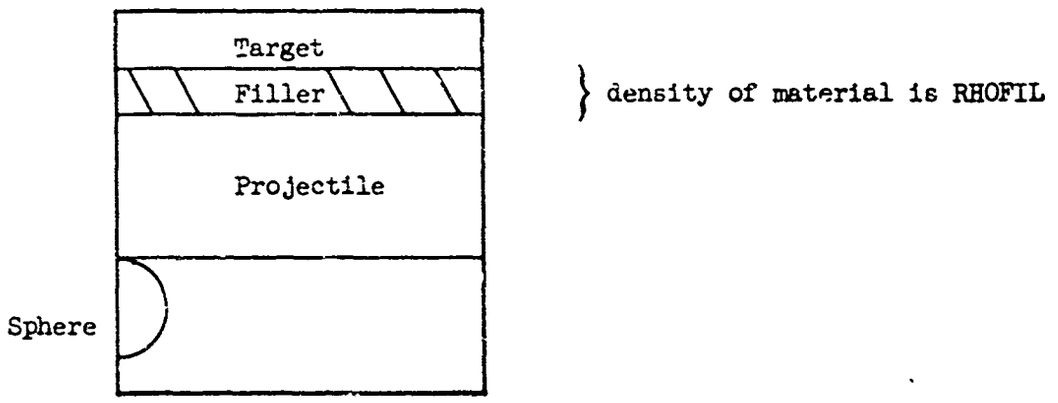
Record Number	Description
1	555.0, CYCLE, N3
2	Z(I), I = 1, MZT
3	U(I), V(I), AMX(I), AIX(I), P(I), I = 1, KMAX
4	X(0),(X(I), TAU(I), JPM(I), I = 1, IMAX)
5	Y(I), I = 0, JMAX
6	((XP(I,J), YP(I,J), I = 1, II), J = 1, JJ)
7	DX(I), I = 1, IMAX
8	DY(I), I = 1, JMAX
9	666.0, 666.0, 666.0

3. SET-UP OF PROBLEMS

Most calculations performed with the OIL-RPM code involve either a sphere or a cylinder hitting a target. Since the cylinder (which is called a "projectile" in the code) can be stationary, the set-up can also provide for a sphere hitting a two-layer target with or without filling material between the layers, as sketched in the diagram below.

The geometries described by the code are limited by the following requirements:

1. The Y-axis of the "sphere" must be on the Y-axis of the grid, i.e. the code cannot setup problems involving toroids.
2. The center of the sphere must coincide with a cell boundary. (Note: the center can be at the origin of the grid.)
3. The edge of the sphere should be contiguous with the projectile or target.
4. The projectile package is assumed to be below the target package.
5. The filler material can be placed only between the projectile package and the target package and extends out to the right boundary of the grid.
6. The right and top boundaries of the grid are transmittive, although the bottom boundary can be either reflective or transmittive.



The input cards for setting up an OIL-RPM problem define (1) the dimensions of the grid as well as the packages (sphere, projectile, and target); (2) the properties of the packages (density, velocity, and specific internal energy); (3) the physical constants used in the equation of state and yield-strength calculation; (4) the calculational constants used in defining various cutoffs and flags; (5) the frequency of printing and of writing on the restart tape; and (6) the time or cycle at which to stop execution.

Most of the input variables are equivalenced to an element in the Z-array, the first array in Blank Common. The variables, therefore, are identified on the input cards by their location in Blank Common (i.e., in the Z-array). The list that follows gives the variable name associated with each input number, its location in Blank Common, and a brief description of its function in the code.

Following this list will be a discussion of the format and order of the input cards, and the RPM Setup Sheets illustrating the input for a typical impact problem.

3.1. DEFINITION OF RPM INPUT VARIABLES

3.1.1. Identification

Variable Name	Location in Blank Common	Definition
PK(1)	151	The problem number can be any number with at most 2 places to the left and at most 4 to the right of the decimal point. (Range: 00.001 to 99.9999)
PROB	1	The problem number is repeated if this is a new problem just being set up. (It is not repeated for restarts from tape. See "RPM Input for Restart.")

3.1.2. Dimensions

Variable Name	Location in Blank Common	Definition
IMAX	33	The number of columns in the calculational mesh. IMAX ≤ 50 for the standard version. Unless IMAX = 1 for 1-D problem, IMAX must be even in order to rezone the grid.
JMAX	35	The number of rows in the calculational mesh. In the standard version JMAX ≤ 100, and IMAX times JMAX ≤ 2500. JMAX must be even to rezone the grid.
I1	47	The right-most column to be calculated initially. It should be two columns beyond the last column with non-zero energy (kinetic or internal). (I1 ≤ IMAX)
I2	48	The top row to be calculated initially. It should be two rows above the last column with non-zero energy. (I2 ≤ JMAX) I1 and I2 define the "active grid." I1 and I2 are automatically adjusted as the problem runs.
DXF	136	The value of all the DX's if DX is constant.
DYF	137	The value of all the DY's if DY is constant.
X(I)	269	The number of cms. from the axis to the right side of column I. The x-array is included in the input deck only if the radial dimension of the cells varies.
Y(J)	166	The number of cms. from the bottom of the grid to the top of row J. The Y-array is included in the input deck only if the axial dimension of the cells varies.

3.1.3. Projectile

Variable Name	Location in Blank Common	Definition
PRYBOT	67	The number of cms. from the bottom of the grid to the bottom of the projectile.
PRYTOP	68	The number of cms. from the bottom of the grid to the top of the projectile.
PRXRT	69	The number of cms. from the axis to the right of the projectile.
VINI	112	Initial axial velocity of the projectile.
PROJU	73	Initial radial velocity of the projectile.
PROJI	16	Initial specific internal energy of the projectile.
RHINI	111	Initial density of the projectile.

3.1.4. Target

Variable Name	Location in Blank Common	Definition
TAYBOT	88	The number of cms. between the bottom of the grid and the bottom of the target.
TAYTOP	89	The number of cms. between the bottom of the grid and the top of the target.
TAXRT	107	The number of cms. between the axis and the right of the target.
TARGV	52	Initial axial velocity of the target.
TARGI	72	Initial specific internal energy of the target.
RHINIT	15	Initial density of the target.

3.1.5. Sphere

Variable Name	Location in Blank Common	Definition
RADIUS	162	The radius (in cms.) of the sphere.
YCENTR	163	The number of cms. from the bottom of the grid to the center of the sphere. (Note: The center of the sphere must be on a cell boundary and on the axis of the grid.)
VINSPH	102	Initial axial velocity of the sphere.
SIESPH	101	Initial specific internal energy of the sphere.
RHOSPH	100	Initial density of the sphere.
RHOOUT	103	Initial density of material in the outside part of a cell cut by the sphere boundary.

3.1.6. Filler

Variable Name	Location in Blank Common	Definition
RHOFIL	51	Initial density of material between projectile and target.

3.1.7. Physical Constants

Variable Name	Location in Blank Common	Definition
AMDM	21	The lowest compression of material considered "solid." If $\rho/\rho_o \leq$ AMDM, and material is cold (specific internal energy < ES') stresses and pressure are zero.
CZERO(Y_o)	42	Parameters used in yield-strength equation:
STK1(Y_1)	11	$Y = (Y_o + Y_1\mu + Y_2\mu^2) \cdot (1 - E/E_o)$ where
STK2(Y_2)	28	$\mu = \rho/\rho_o - 1$, and E = specific internal energy.
STEZ(E_o)	29	However, (1) If $Y < 0$, stresses are set to 0. (2) If $E > E_o$, then $Y = 0$.

Variable Name	Location in Blank Common	Definition	
BBAR	149	A constant used to approximate sound speed for the calculation of DT. $C = C_o + \bar{B} \cdot \sqrt{P(K)}$, where $C_o = \sqrt{A/\rho_o}$.	
RHOZ	115	The normal density of the material. This value is used in the equation of state, the calculations of PMIN and C_o, the strength calculations, and the "slaving" process.	
ESA(a)	116	Gruneisen ratio at high energy, ~ .5.	Equation of State constants, See Ref. 4 or Section 4, PART TWO
ESEZ(E_o)	117	A constant in Gruneisen ratio.	
ESB(b)	118	A constant in Gruneisen ratio.	
ESCAPA(A)	119	Bulk modulus.	
ESESP(ES')	120	Heat to vaporize material; must be larger than ESES.	
ESESQ(ESQ)	121	Used to test whether material is "cold;" usually identical to ESESP.	
ESES(ES)	122	Heat to bring material to vapor temperature; must be smaller than ESESP.	
ESALPH(α)	123	Usually 5.	
ESBETA(β)	124	Usually 5.	
ESCAPB(B)	125	Usually of the same order as A.	

3.1.8. Calculational Constants

Variable Name	Location in Blank Common	Definition
RHOMIN	138	The smallest density a cell can have and still influence the calculation of DT. (Usually RHOMIN = RHOZ * 10^{-3}.)

Variable Name	Location in Blank Common	Definition
DMIN	24	The allowable relative error in the energy sum. The error is the difference of the current total energy of all the cells and the total energy computed on cycle zero but adjusted for energy "evaporated" and lost across boundaries. The relative error is the difference divided by the total energy. If it exceeds DMIN, ERROR is called and the calculation is terminated.
DTMIN	144	The minimum value of DT (after STAB = FINAL) for the calculation to continue. DTMIN may be zero.
EVAP	75	This variable controls the "evaporation" of mass. If ρ < EVAP * RHINI, the cell mass is evaporated. The mass, energy and momentum of evaporated cells are accumulated in PH2 and printed in EDIT. (Usually EVAP = 10^{-4}.)
ROEPS	110	The "round-off epsilon" used in setting to zero certain calculated differences which could be due simply to machine round-off. (Usually ROEPS = 10^{-5} or 10^{-6}.)
STAB	139	The stability fraction used in determining DT. The input value of STAB is its initial value. If FINAL > 0., STAB is doubled on each cycle until it equals FINAL. However, if FINAL = 0., the initial value of STAB is used throughout the run. STAB is usually $\leq 10^{-3}$, but when all the energy is initially internal, setting STAB $\sim 10^{-8}$ is recommended.

Variable Name	Location in Blank Common	Definition
FINAL	113	FINAL is used in determining DT. If FINAL is > 0., then it is the largest value the stability fraction (STAB) will have. If FINAL = 0., the stability fraction will have the same value for each cycle. (FINAL is usually ~ .4.)
JSTR	25	JSTR (J strength) gives the value of I2 (active grid, J direction) at which stress calculations (PH3) are turned on and tensions are allowed.
N6	56	N6 specifies the J index of the cell behind which tensions (negative pressures) are to be allowed. If N6 = 0., tensions are allowed everywhere.
CYCPH3	70	CYCPH3 = -1. if no stress calculation is wanted. Otherwise, it is the number of subcycles of PH3 per time step. (Usually about 4.)
NUMREZ	12	The maximum number of times the grid will be rezoned.
NMPMAX	85	The maximum number of tracer points to be used.
Y2	81	Y2 = -2. if tracer points are to be calculated. Y2 = 0. if no tracer points are to be calculated.
REZFCT	71	REZFCT = 1. if rezones are allowed. REZFCT = 0. if rezones are not allowed.
SS4	130	SS4 = 1. if a rezone is to be forced on the second cycle of a run. (Often used to test the setup of problems to be rezoned.)

Variable Name	Location in Blank Common	Definition
SN	65	SN = 0., if negative specific internal energy is to be set to zero. SN = 1., if negative specific internal energy is to be left along.
CVIS	27	CVIS = 0., if the bottom boundary of the grid is to be reflective. CVIS = -1., if the bottom boundary of the grid is to be transmittive.
INTER	87	INTER = 0., gives no intermediate prints. INTER = 1., gives EDIT prints after PH1 and PH3, as well as CDT. INTER = 7., gives, in addition to the extra EDIT prints, details of PH2 calculations. INTER = 99., gives, in addition to the extra EDIT prints, details of PH3 stress and strain rates. CAUTION: INTER = 7., or = 99., gives <u>many</u> pages of output.
IVARDX	83	IVARDX = 0., if DX is constant and the X array is to be calculated from the value of DXF. IVARDX = 1., if DX varies and the X array is included in the input deck.
IVARDY	54	IVARDY = 0., if DY is to be constant and the Y array is to be calculated from the value of DYF. IVARDY = 1., if DY varies and the Y array is included in the input deck.

3.1.9. Output

Variable Name	Location in Blank Common	Definition
JPROJ	147	JPROJ is usually assigned the value of J at the top of the projectile. In EDIT, JPROJ is used as the zero in calculating the crater depth and is the division for the printout of total energy, mass and momentum.
PRDELT	45	The number of seconds between EDIT prints when printing on time. Otherwise 0.
IPCYCL	49	The number of cycles between EDIT prints when printing on cycles. Otherwise 0.
PRLIM	44	PRLIM is the time or cycle at which the EDIT print interval is to be increased. PRLIM is multiplied by PRFACT each time the print interval is adjusted.
PRFACT	46	The factor by which the print interval is increased. PRDELT (or IPCYCL) and PRLIM are multiplied by PRFACT when T = PRLIM.
NUMSCA	43	NUMSCA is the number of times the code will increase the interval time (or number of cycles) between EDIT prints.
NFRELP	5	NFRELP indicates the frequency of "long" EDIT prints. (A "long" print gives the velocities, pressure, mass, energy, density, and compression of all cells in the active grid; the "short" print gives this information only for the cells in the first column of the active grid.) A "long" print will occur every NFRELP short prints.

Variable Name	Location in Blank Common	Definition
NDUMP7	6	NDUMP7 indicates the frequency of "tape dumps," where most of Blank Common is written on an output tape. A tape dump will occur every NDUMP7 EDIT prints. These "tape dumps" are used for restarting problems and for making automatic plots of tracer points.
NODUMP	96	NODUMP = 1. allows the user to pick up a run at some intermediate point on the restart tape without writing over the subsequent dumps on that tape.

3.1.10. Stopping

Variable Name	Location in Blank Common	Definition
ICSTOP	7	The cycle for execution to stop when stopping on cycles.
TSTOP	50	The value of T for execution to stop when stopping on time rather than cycles. NOTE: This card, because of its "1" in column one, must always be included in the initial input deck. If stopping on cycles set to zero.
Z(150)	150	Dummy end card. Used in the RPM input deck for setting up problems. (Do not include this card in an input deck when restarting a problem.)

3.2. FORMAT AND ORDER OF OIL-RPM INPUT CARDS

Except for the ID header card, all RPM input cards have the same format and are normally punched on 7-word data cards.

The RPM setup sheets provide the information to be punched in Col. 1, Cols. 2-6, and Col. 7 for each input variable. The values of the variables are punched in the seven 9-space fields in Cols. 8-70. These values must

be punched with a decimal point even when they define integer variables. If the E-format is used, the exponent must be right-adjusted in the field. Only those variables which occur in consecutive order in Blank Common can be punched on the same card.

Col. 1	Cols. 2-6	Col. 7	Cols. 8-16	Cols. 17-25 etc.
Punch "1" on first and last two cards of deck. (See Setup Sheet.) Punch "2" on all cards defining integer variables. Otherwise no punch.	N Location in Blank Common of variable defined in Cols. 8-16.	Number of variables being defined on this card.	Z(N) Value of variable equi-valenced to Z(N).	Z(N+1) Value of variable equivalenced to Z(N+1).

For a normal setup deck, the only data cards which must be in a specific order are those listed on the Setup Sheet with a "1" in column one, and they must be the first and the last two cards in the deck.

3.3. OIL-RPM INPUT FOR "SPECIAL SETUPS"

The properties (density, velocity, internal energy) of each cell are defined by the subroutine SETUP according to the input parameters associated with the sphere, cylinder, filler and target (e.g., SIESPH, VINI, RHOFIL, TARGV). The RPM "special setup" allows the user to assign a special mass (not density), velocity or internal energy to specific cells in the grid.

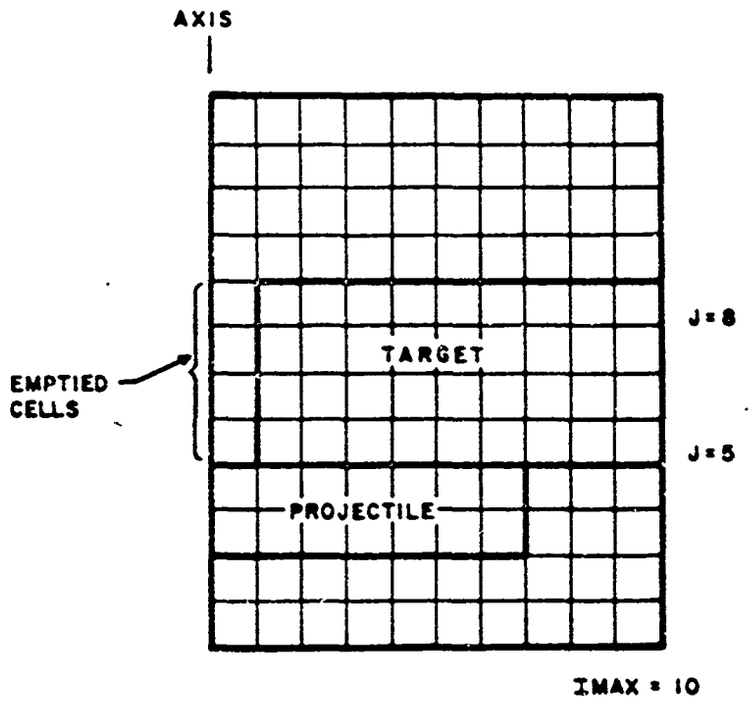
To assign special properties to a specific cell the user must do the following:

1. Fine the I and J of the cell, then compute its K-index. (K = (J - 1)* IMAX + I + 1).
2. Find the location in Blank Common of AMX(K), U(K), V(K), and AIX(K), whichever is being specifically defined. (See Table 3, page 26, "Location of Arrays in Blank Common.")
3. Using the format of the other RPM input cards, punch a card with the location and value for each property being assigned.

Precede these input cards by a flag card which has a "1" in column one and which sets Z(164) = PK(14) = 1. (See Figure 2, page " Flow Diagrams of INPUT, CARDS and SETUP.") Place the special input cards in the

RPM input deck just before the card that defines ICSTOP(Z(7)). The flag card (PK(14) = 1.) must be the second card with a "1" in the column one in the input deck.

EXAMPLE: Suppose one column of cells in the target next to the axis is to be empty. The target is to range from J = 5 to J = 8, and IMAS = 10. (NOTE: This problem requires special input because RPM places all packages next to the axis of the grid.)



1. Compute the K-index of the empty cells.

I = 1,	J = 5,	K = 4*10 + 1 + 1 = 42
I = 1,	J = 6	K = 52
I = 1,	J = 7	K = 62
I = 1,	J = 8	K = 72

2. Find the location in Blank Common of members of the AMX array which store the masses of these cells. Table 3 indicates that 482 is the location of AMX (1) in Blank Common.

Variable Name	Location in Blank Common
AMX(42)	523
AMX(52)	533
AMX(62)	543
AMX(72)	553

3. The input deck would be organized as follows:

Description of card	Col. 1	Col. 2-6	Col. 7	Col. 8-16
ID CARD				IMPACT
PK(1)	1	151	1	3.2
Z(1)=PROB	blank		1	3.2
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
NODUMP	2	96	1	0.
PK(14)	1	164	1	1.
AMX(42)	blank	523	1	0.
AMX(52)	blank	533	1	0.
AMX(62)	blank	543	1	0.
AMX(72)	blank	553	1	0.
ICSTOP	2	7	1	0.
TSTOP	1	50	1	1. - 06
Dummy End	1	150	1	0.

TABLE 3

LOCATION OF ARRAYS IN BLANK COMMON*

Name of Array	Location of First Member of Array
Z	1
PK	151
YY Note: YY(1) = Y(0)	166
XX Note: XX(1) = X(0)	270
DDX Note: DDX(1) = DX(0)	324
DDY Note: DDY(1) = DY(0)	378
AMX	482
AIX	2984
U	5486
V	7988
P	10490
TAU	12992
JPM	13044
UL	13096
PL	13300
XP	13504
YP	14830
CMXP	16156
CMYP	16161

* These location numbers should not be used if the dimension of any array in Blank Common is changed.

4. SAMPLE PROBLEM

The impact of an aluminum sphere on a target of like material was described in an earlier report,[8] and the setup of the problem will be described in this section as an example. The sphere diameter was .4763 cms (3/16 in.), and its velocity was 7.35 km/sec. The zoning was chosen so that the sphere radius contained exactly 6 cells, the free surface of the target was 16 zones above the bottom of the grid, and the cells were square. It was found that after 16 microseconds the velocities were down to a value where the subsequent material motion would be negligible. The grid was rezoned twice during the computation. About two hours of computer time and about 450 cycles were required to complete the problem.

Details of the setup are given by the OIL-RPM Setup Sheets which list the appropriate values for the input parameters and describe the format of the data cards. Also, the main section of the printed output are illustrated and briefly discussed in addition to the control cards for the UNIVAX 1108.

7-WORD DATA
OIL-RPM SETUP SHEET

	2 LOC	7 N	8 1	17 2	26 3	35 4	44 5	6	62 7	71	73
					STANDARD	CRATER					
1	151	1	13.735								
	1	1	13.735								
2	33	1	42.								
2	35	1	54.								
2	47	1	8.								
2	48	1	18.								
	136	1	.03997								
	137	1	.03997								

PK(1) = Problem number in the form 92.3456.

PROB = Problem number (exactly the same as previous card).

IMAX = Number of columns. An <u>even</u> number ≤ 50. unless IMAX = 1.

JMAX = Number of rows. An <u>even</u> number ≤ 100.

} IMAX * JMAX ≤ 2500 unless variable dimensions are changed.

I1 = Number of columns in "active" grid + 2. (If IMAX = 1., I1 = 1.)

I2 = Number of rows in "active" grid + 2.

DXF = DX if it is constant. If DX varies DXF = 0.

DYF = DY if it is constant. If DY varies DYF = 0.

Insert X-array only if DX varies. (See Page 31) Insert Y-array only if DY varies. (See page 31)

	67	1	-1.
	68	1	0.
	69	1	0.
	112	1	0.
	73	1	0.
	16	1	0.
	111	1	2.7
	88	1	.6352
	89	1	1.0 +10
	107	1	1.0 +10
	52	1	0.
	72	1	0.
	15	1	2.7

PRYBOT = Y-value (cm) of bottom of projectile. PRYBOT = -1., if no projectile.

PRYTOP = Y-value (cm) of top of projectile.

PRXRT = X-value (cm) of right of projectile.

VINI = Initial axial velocity of projectile.

PROJU = Initial radial velocity of projectile.

PROJI = Initial specific internal energy of projectile.

RHINI = Initial density of projectile.

TAYBOT = Y-value (cm) of bottom of target. TAYBOT = -1., if no target.

TAYTOP = Y-value (cm) of top of target.

TAXRT = X-value (cm) of right of target.

TARGV = Initial axial velocity of target.

TARGI = Initial specific internal energy of target.

RHINIT = Initial density of target.

7-WORD DATA
OIL-RPM SETUP SHEET

2 LOC	7 N	8 1	17 2	26 3	35 4	44 5	6	62 7	71	73	
462	1	.2382									RADIUS = radius (cm) of sphere. (RADIUS need not fall on cell boundary.)
163	1	397									YCENTR = Y-center (cm) of sphere. (YCENTR must coincide with a cell boundary.)
102	1	7.35			+05						VINSPH = Initial velocity of sphere.
101	1	0.									SIESPH = Initial specific internal energy of sphere.
100	1	2.7									RHOSPH = Initial density of sphere.
103	1	0.									RHOOUT = Initial density of material adjacent to sphere.
51	1	0.									RHOFIL = Initial density of filling material between projectile and target.
21	1	.97									AMDM = Allowable expansion of material considered "solid." (Usually between .9 & .99.)
42	1	2.39			+09						CZERO (Y_0)
11	1	2.4			+10						STK1 (Y_1)
28	1	4.1			+10						STK2 (Y_2)
29	1	7.9			+09						STEZ (E_0)
149	1	.5									BBAR = Constant in "local sound speed" calculation. ($C = C_0 + BBAR * \sqrt{P(K)}$)
115	1	2.7									RHOZ = Normal density (ρ_0) for the equation of state.
116	5	.5	5.0	+10	1.63		7.52	+11	1.3	+11	a, E_0, b, A, ES'
121	5	3.0	3.0	+10	5.0		5.9		6.5	+11	ESQ, ES, α, β, B
138	1	1.0			-03						RHOMIN = Minimum density for influencing DT. Usually 10^{-3} * RHOZ.
24	1	1.0			-03						DMIN = Allowable relative error for total energy sum. Usually 10^{-3}.
144	1	1.0			-09						DTMIN = Minimum value of DT for program to continue execution. DTMIN may be 0.
75	1	1.0			-04						EVAP = Minimum allowable compression. If ρ/ρ_0 < EVAP, mass of cell is "evaporated."
110	1	1.0			-06						RCEPS = Round-off epsilon. Usually 10^{-5} or 10^{-6}.
139	1	1.0			-03						STAB = Initial value of "stability fraction." Usually < 10^{-3}.
143	1	.4									FINAL = Final value of "stability fraction." Usually = .4.

Used in strength calculation

$$Y = [Y_0 + \mu\,(Y_1 + Y_2\mu)](1 - E/E_c)$$
$$\mu = \rho/\rho_c - 1$$

Equation of State Constants. Normally ES=ESQ.

7-WORD DATA
OIL-RPM SETUP SHEET

	2 LOC	7 N	8 1	17 2	26 3	35 4	44 5	6	62 7	71	73
2	25	1	98.								
2	56	1	1.4.								
	70	1	4.								
2	12	1	2.								
2	85	1	567.								
	81	1	-2.								
	71	1	1.								
	130	1	9.								
	65	1	1.								
	27	1	-1.								
2	87	1	9.								
2	83	1	9.								
2	54	1	0.								
2	147	1	16.								
	45	1	2.					-97			
2	49	1	0.								
	44	1	1.					-96			
	46	1	2.								
2	43	1	10.								
2	5	1	5.								
2	6	1	1.								
2	96	1	9.								
2	7	1	0.								
1	50	1	1.0					-96			
1	150	1	0.								

JSTR = Value of I2 which triggers calculation of strength and negative pressures.

N6 = Value of J below which negative pressures are not allowed.

CYCPH3 = Number of times of subcycle PH3. CYCPH3 = -1., for no PH3.

NUMREZ = Maximum number of rezones allowed.

NMPMAX = Maximum number of tracer points desired. NMPMAX ≤ 1250.

Y2 = -2., if tracer points are wanted. Y2 = 0. for no tracer points.

REZFCT = 1. for allowing rezones. REZFCT = 0. for no rezones.

SS4 = 1. for forcing rezone on second cycle of run. Usually SS4 = 0.

SN = 1. leave negative S.I.E. alone. SN = 0. sets negative S.I.E. to 0.

CVIS = -1. for transmittive bottom boundary. CVIS = 0. for reflective bottom boundary.

INTER = Intermediate print flag.

IVARDX = 1. if DX varies and the X-array is read in. IVARDX = 0. if DX is constant.

IVARDY = 1. if DY varies and the Y-array is read in. IVARDY = 0. if DY is constant.

JPROJ = Usually the J-value of top row of projectile. Used in EDIT print, crater depth calculation.

PRDELT = Time between EDIT prints. PRDELT = 0. if printing on cycles.

IPCYCL = Cycles between EDIT prints. IPCYCL = 0. if printing on time.

PRLIM = Time or cycle at which to increase interval between prints.

PRFACT = Factor by which to increase print interval.

NUMSCA = Maximum number of times print interval can be increased.

NFRELP = Ratio of "short" prints to "long" prints.

NDUMP7 = Ratio of prints to tape dumps.

NODUMP = 0. allows tape dumps. NODUMP = 1. suppresses tape dumps after cycle 0.

ICSTOP = Cycle to stop execution. ICSTOP = 0. if stopping on time.

TSTOP = Time (value of T) to stop execution. TSTOP = 0. if stopping on cycles.

Dummy end card.

7-WORD DATA

OIL-RPM SETUP SHEET--X AND Y ARRAYS*

2 LOC	7 N	8 1	17 2	26 3	35 4	44 5	6	62 7	71	73
270	7									
277	7									
284	7									
291	7									
298	7									
305	7									
312	7									
319	2									
165	7									
173	7									
180	7									
187	7									
194	7									
201	7									
208	7									
215	7									
222	7									
229	7									
236	7									
243	7									
250	7									
257	7									
264	3									

X(0),... ,X(50)
Input these values only if DX varies.

Y(0),... , Y(100)
Input these values only if DY varies.

*These location numbers (Cols. 2-6) must be changed if the dimension of Z, PK, or YY is changed.

4.1. OIL-RPM Output

The pages of OIL-RPM output which follow were produced by the sample problem, "Standard Crater," described above. The printed headings make most of the listing self-explanatory. Sections needing further description are numbered in the listing and discussed below.

1. The first few pages of the output for a setup run* display the input deck and describe the initial conditions of the problem. (Each time the CARDS routine is called "INPUT CARDS" is printed.)

2. When a sphere is placed, SETUP assures that the value of YCENTR is on a cell boundary and prints the input and adjusted values of YCENTR.

3. On every cycle subroutine CDT prints the value of T, the time, and DT, the time step. The integers following "CDT" in the printout are the I and J of the cell controlling the time step. MAXCUV represents the maximum sound speed or velocity in the active grid. Likewise, MAXUV represents the maximum velocity. UMIN and PMIN are velocity and pressure cutoffs, respectively, used in MAP and PH2.

4. The first page printed by EDIT gives the total energies, mass and momenta of the cells above the JPROJ row and of the cells below and in the JPROJ row. (JPROJ is an input parameter, usually the J of the top row of the projectile package.)

5. On <u>each</u> cycle EDIT calculates the relative error in the total energy sum. On <u>print</u> cycles EDIT prints the maximum error calculated and the number of the cycle in which the maximum error occurred.

6. The total work done due to stresses calculated in PH3 is printed under "PLASTIC-WORK."

7. Also printed is a running total of the mass, energy, and axial and radial momentum lost when material crosses a transmittive boundary or is "evaporated" in PH2.

---

*For a restart run only the input deck is printed.

8. The "J OF PRESSURE MAXIMUM" describes the shock front by giving the location of the peak pressure in each column.

9. The tracer point coordinates are printed in centimeter units. The I and J of the cell in which a tracer point originated is also printed. This enables the user to follow the movement of a given tracer point. However, if the mass of a cell is "evaporated" the coordinates of the tracer points in that cell are set to zero and thereafter are not printed out.

10. The range of values assigned to the symbols on the contour maps are adjusted as the calculation proceeds. Therefore, a given symbol in the compression map, for instance, does not represent the same compression from one print to the next. A negative pressure, velocity, or internal energy is denoted by a symbol representing its absolute value and preceded by a minus sign.

11. The crater depth is measured from JPROJ. After a crater is formed, the negative values for the crater depth will describe the crater's "lip". The volume of the crater is printed along with the volume of a hemisphere of radius equal to the crater's depth in column one.

12. The J-index, radial velocity, axial velocity, pressure, mass, density, specific internal energy, compression, and distance (in cms.) from the bottom of the grid are printed for non-empty cells. The "long" EDIT print lists these properties for <u>every</u> column of the active grid, whereas the "short" EDIT print describes the non-empty cells in column one only.

13. Several error conditions are tested for during execution, and when one is detected, subroutine ERROR is called. ERROR in turn identifies the error test, prints the entire Z-array, and calls EDIT to do a long print and tape dump before stopping.

STANDARD CRATER

INPUT CARDS

(1.)

1	151	1	1.375500+01

INPUT CARDS

-0	1	1	1.375500+01				
2	35	1	4.200000+01				
2	35	1	5.390000+01				
2	47	1	6.000000+00				
2	46	1	1.600000+01				
-0	136	1	3.970000-02				
-0	137	1	3.970000-02				
-0	67	1	-1.000000+00				
-0	68	1	0.000000				
-0	69	1	0.000000				
-0	112	1	0.000000				
-0	75	1	0.000000				
-0	16	1	0.000000				
-0	111	1	2.700000+00				
-0	38	1	6.352000-01				
-0	89	1	1.000000+10				
-0	107	1	1.000000+10				
-0	52	1	0.000000				
-0	72	1	0.000000				
-0	15	1	2.700000+00				
-0	162	1	2.382000-01				
-0	163	1	3.970000-01				
-0	102	1	7.350000+05				
-0	101	1	0.000000				
-0	100	1	2.700000+00				
-0	103	1	0.000000				
-0	51	1	0.000000				
-0	21	1	9.700000-01				
-0	42	1	2.390000+09				
-0	11	1	2.400000+10				
-0	28	1	4.100000+10				
-0	29	1	7.000000+09				
-0	149	1	5.000000-01				
-0	115	1	2.700000+00				
-0	116	5	5.000000-01	5.000000+10	1.630000+00	7.520000+11	1.300000+11
-0	121	5	5.000000+10	3.000000+10	5.000000+00	5.000000+00	6.500000+11
-0	138	1	1.000000-03				
-0	24	1	1.000000-03				
-0	144	1	1.000000-09				
-0	75	1	1.000000-04				
-0	110	1	1.000000-06				
-0	139	1	1.000000-03				

```
 -0    113  1   4.000000-01
  2     25  1   2.000000+01
  2     56  1   1.000000+01
 -0     70  1   4.000000+00
  2     12  1   2.000000+00
  2     85  1   5.070000+02
 -0     81  1  -2.000000+00
 -0     71  1   1.000000+00
 -0    130  1   0.000000
 -0     65  1   1.000000+00
 -0     27  1  -1.000000+00
  2     87  1   0.000000
  2     85  1   0.000000
  2     54  1   0.000000
  2    147  1   1.600000+01
 -0     45  1   1.000000-06
  2     49  1   0.000000
 -0     44  1   1.000000-06
 -0     46  1   2.000000+00
  2     43  1   1.000000+01
  2      5  1   5.000000+00
  2      6  1   1.000000+00
  2     96  1   0.000000
  2      7  1   0.000000
  1     50  1   1.000000-06
```

(2.) INPUT YCENTR = 3.970000-01 ADJUSTED YCENTR = 3.970000-01

INITIAL CONDITIONS

	SPHERE		PACKAGE 1	PACKAGE 2	FILLER
DENSITY	2.7000+00		2.7000+00	2.7000+00	0.0000
S.A.E.	0.0000		0.0000	0.0000	
V	7.3500+05		0.0000	0.0000	
U			0.0000		
RADIUS	2.3820-01	TOP	0.0000	1.0000+10	
YCENTER	3.9700-01	BOTTOM	-1.0000+00	6.3520-01	
		RIGHT	0.0000	1.0000+10	

INPUT CARDS

```
  1    150  1   0.000000

          NUMREZ= 2  JSTR= 28  N6= 10  IMAX= 42  JMAX= 54  I1=  8  I2= 18  JPROJ= 16  NMPMAX=  567  INTFR= 0  NUMSCA=10
IPCYCL=  0  ICSTOP=   0  NFRELP=  5  NDUMP7=  1  NODUMP= 0  IVAROX= 0  IVARDY= 0
```

③

```
CUT 17  12  T= 5.7330342-07  DT= 1.7105428-08  MAXCUV= 9.2836024+05  MAXUV= 9.2836024+05  UMIN= 9.2836023-01  PMIN= 1.3228399+06
CUT 17  11  T= 5.9640684-07  DT= 1.7160432-08  MAXCUV= 9.2430734+05  MAXUV= 9.2430734+05  UMIN= 9.2430732-01  PMIN= 1.3170649+06
CUT 17  11  T= 6.0758927-07  DT= 1.7960208-08  MAXCUV= 8.8417683+05  MAXUV= 8.8417683+05  UMIN= 8.8417681-01  PMIN= 1.2598821+06
CUT 16  12  T= 6.2554947-07  DT= 1.7412566-08  MAXCUV= 9.1198503+05  MAXUV= 9.1198503+05  UMIN= 9.1198502-01  PMIN= 1.2995066+06
CUT 16  12  T= 6.4296203-07  DT= 1.6202935-08  MAXCUV= 8.7238675+05  MAXUV= 8.7238675+05  UMIN= 8.7238675-01  PMIN= 1.2430821+06
CUT 16  11  T= 6.6116496-07  DT= 1.8313324-08  MAXCUV= 8.6712820+05  MAXUV= 8.6712820+05  UMIN= 8.6712819-01  PMIN= 1.2355892+06
CUT 16  11  T= 6.7947629-07  DT= 1.9107477-08  MAXCUV= 8.3108822+05  MAXUV= 8.3108822+05  UMIN= 8.3108822-01  PMIN= 1.1842350+06
CUT 10  26  T= 6.9858576-07  DT= 1.9526046-08  MAXCUV= 8.1324770+05  MAXUV= 7.9660646+05  UMIN= 8.1324768-01  PMIN= 1.1588137+06
CUT  9  27  T= 7.1811641-07  DT= 1.9712737-08  MAXCUV= 8.0557049+05  MAXUV= 7.6116204+05  UMIN= 8.0557048-01  PMIN= 1.1478743+06
CUT 10  27  T= 7.3782514-07  DT= 1.9694613-08  MAXCUV= 8.0630364+05  MAXUV= 7.2520310+05  UMIN= 8.0630363-01  PMIN= 1.1489189+06
CUT  9  28  T= 7.5751995-07  DT= 1.9847581-08  MAXCUV= 8.0009749+05  MAXUV= 7.1909469+05  UMIN= 8.0009748-01  PMIN= 1.1400757+06
CUT 11  27  T= 7.7736752-07  DT= 1.9906493-08  MAXCUV= 7.9772964+05  MAXUV= 6.8658896+05  UMIN= 7.9772963-01  PMIN= 1.1367017+06
CUT 10  28  T= 7.9727402-07  DT= 1.9937062-08  MAXCUV= 7.9648253+05  MAXUV= 6.7095737+05  UMIN= 7.9648252-01  PMIN= 1.1349247+06
CUT 11  28  T= 8.1721167-07  DT= 2.0008527-08  MAXCUV= 7.9050096+05  MAXUV= 6.5599599+05  UMIN= 7.9050095-01  PMIN= 1.1264013+06
CUT 10  29  T= 8.3730020-07  DT= 2.0092037-08  MAXCUV= 7.9036296+05  MAXUV= 6.6331750+05  UMIN= 7.9036284-01  PMIN= 1.1262046+06
CUT 12  28  T= 8.5739223-07  DT= 2.0253253-08  MAXCUV= 7.8407156+05  MAXUV= 6.5883274+05  UMIN= 7.8407155-01  PMIN= 1.1172400+06
CUT 11  29  T= 8.7704547-07  DT= 2.0244983-08  MAXCUV= 7.8439185+05  MAXUV= 6.5202171+05  UMIN= 7.8439184-01  PMIN= 1.1176964+06
CUT 10  30  T= 8.9789040-07  DT= 2.0351241-08  MAXCUV= 7.8029636+05  MAXUV= 6.4354466+05  UMIN= 7.8029636-01  PMIN= 1.1118606+06
CUT 12  29  T= 9.1824169-07  DT= 2.0382380-08  MAXCUV= 7.7910425+05  MAXUV= 6.3772611+05  UMIN= 7.7910423-01  PMIN= 1.1101619+06
CUT 10  31  T= 9.3862407-07  DT= 2.0512106-08  MAXCUV= 7.7417693+05  MAXUV= 6.3620342+05  UMIN= 7.7417693-01  PMIN= 1.1031409+06
CUT 12  30  T= 9.5913017-07  DT= 2.0516787-08  MAXCUV= 7.7400029+05  MAXUV= 6.3356148+05  UMIN= 7.7400029-01  PMIN= 1.1028893+06
CUT 11  31  T= 9.7905495-07  DT= 2.0635466-08  MAXCUV= 7.6954887+05  MAXUV= 6.2956331+05  UMIN= 7.6954886-01  PMIN= 1.0965463+06
CUT 13  30  T= 9.9999999-07  DT= 2.0698330-08  MAXCUV= 7.6721163+05  MAXUV= 6.2401386+05  UMIN= 7.6721162-01  PMIN= 1.0932159+06
```

(5.) (6.)

PROBLEM	TIME	CYCLE	TOT.EN.THEOR.	MAX.REL.ERROR-CYCLE	IE SET TO ZERO-PH1	IE SET TO ZERO-PH2	PLASTIC-WORK
13.7350	9.9999999-07	70	3.9972263+10	-8.3513908-06 70	0.0000000	-5.6878873+00	3.0655210+08

(4.)

	IE	KE	TOT.EN. (SUM)	MASS	MV	MV(POSITIVE)	MU	MU(POSITIVE)
J.GT.JPROJ	1.4164357+10	2.1862779+10	3.6047136+10	3.5606988+01	1.2888103+05	1.5801904+05	2.0767698+05	2.0768153+05
J.LE.JPROJ	1.9007798+09	2.0240066+09	3.9248052+09	1.2177211-01	-1.5975029+04	3.9293887+01	1.2597252+04	1.2621583+04
TOTALS	1.6065156+10	2.3886786+10	3.9971941+10	3.5728759+01	1.1290600+05	1.5805834+05	2.2027423+05	2.2030311+05

(7.)

BOUNDARY	BOTTOM	RIGHT	TOP	$EVAPORATED$
MASS OUT	0.0000000	0.0000000	0.0000000	8.0971186-04
ENERGY OUT	0.0000000	0.0000000	0.0000000	1.3158280+09
MU OUT	0.0000000	0.0000000	0.0000000	5.3098244+02
MV OUT	0.0000000	0.0000000	0.0000000	-5.6164385+02
WORK DONE	0.0000000	0.0000000	0.0000000	

(8.)

J OF PRESSURE-MAXIMUM

34	34	34	34	34	34	33	33	32	32	31	31	30	29	28	27	25	24	0	0	0	0	0	0	0
0	0	0	0																					

TAPE 7 DUMP ON CYCLE 70

(9.)

TRACER POINTS - INITIAL LOCATION IN CELL COORDINATES (I,J) - CURRENT LOCATION IN CM. COORDINATES (X,Y)

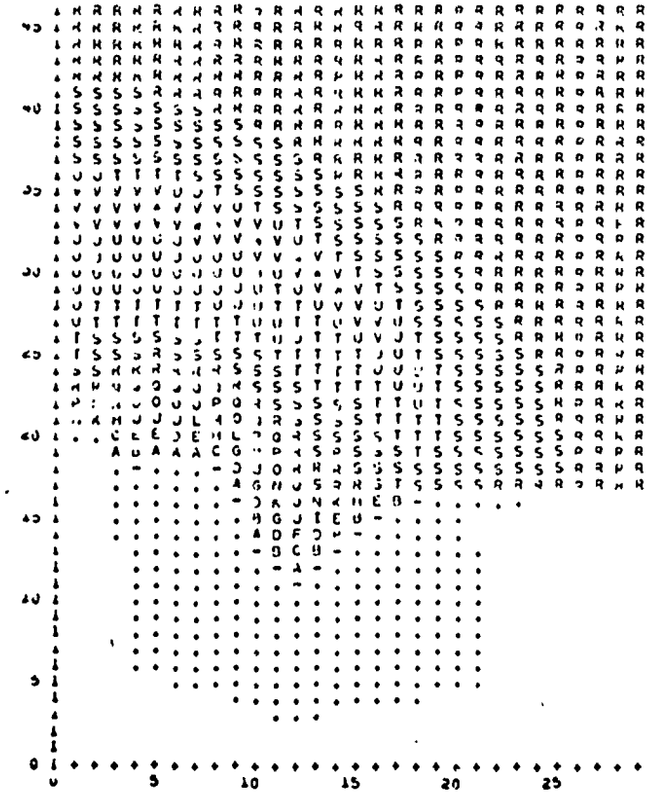
I	J	X	Y	I	J	X	Y	I	J	X	Y	I	J	X	Y	I	J	X	Y
1	7	3.17-02	8.39-01	3	7	1.57-01	8.51-01	5	7	2.52-01	8.29-01	1	9	2.65-02	8.82-01	3	9	1.60-01	8.78-01
5	9	2.05-01	8.26-01	1	11	2.55-02	9.16-01	3	11	1.65-01	9.03-01	5	11	3.13-01	8.22-01	1	13	2.93-02	9.47-01
3	13	1.76-01	9.25-01	5	13	3.47-01	8.00-01	1	15	4.08-02	9.71-01	3	15	1.93-01	9.40-01	5	15	3.82-01	7.48-01
1	17	4.09-02	9.65-01	3	17	2.17-01	9.41-01	5	17	3.83-01	7.55-01	7	17	4.21-01	5.79-01	9	17	4.46-01	5.57-01
11	17	4.75-01	5.76-01	13	17	5.20-01	6.12-01	15	17	5.85-01	6.34-01	17	17	6.57-01	6.48-01	19	17	7.34-01	6.54-01
21	17	8.14-01	6.55-01	23	17	8.93-01	6.55-01	25	17	9.73-01	6.55-01	27	17	1.05+00	6.55-01	29	17	1.13+00	6.55-01
31	17	1.21+00	6.55-01	33	17	1.29+00	6.55-01	35	17	1.37+00	6.55-01	37	17	1.45+00	6.55-01	39	17	1.53+00	6.55-01
41	17	1.61+00	6.55-01	1	19	4.55-02	1.00+00	3	19	1.99-01	9.70-01	5	19	3.39-01	8.68-01	7	19	4.05-01	7.66-01
9	19	4.43-01	7.26-01	11	19	4.81-01	7.15-01	13	19	5.32-01	7.19-01	15	19	5.96-01	7.25-01	17	19	6.64-01	7.31-01
19	19	7.37-01	7.34-01	21	19	8.14-01	7.34-01	23	19	8.93-01	7.34-01	25	19	9.73-01	7.34-01	27	19	1.05+00	7.34-01
29	19	1.13+00	7.34-01	31	19	1.21+00	7.34-01	33	19	1.29+00	7.34-01	35	19	1.37+00	7.34-01	37	19	1.45+00	7.34-01
39	19	1.53+00	7.34-01	41	19	1.61+00	7.34-01	1	21	3.97-02	1.02+00	3	21	1.71-01	1.00+00	5	21	2.91-01	9.46-01
7	21	3.70-01	8.85-01	9	21	4.25-01	8.44-01	11	21	4.79-01	8.25-01	13	21	5.36-01	8.17-01	15	21	5.99-01	8.14-01
17	21	6.67-01	8.15-01	19	21	7.38-01	8.14-01	21	21	8.14-01	8.14-01	23	21	8.93-01	8.14-01	25	21	9.73-01	8.14-01
27	21	1.05+00	8.14-01	29	21	1.13+00	8.14-01	31	21	1.21+00	8.14-01	33	21	1.29+00	8.14-01	35	21	1.37+00	8.14-01
37	21	1.45+00	8.14-01	39	21	1.53+00	8.14-01	41	21	1.61+00	8.14-01	1	23	2.69-02	1.05+00	3	23	1.48-01	1.04+00

(10.)

COMPRESSIVE

SYMBOL	.	-	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
MAXIMUM VALUE	.05	.10	.15	.20	.25	.30	.35	.40	.45	.50	.55	.60	.65	.70	.75	.80	.85	.90	.95	1.00

SYMBOL	S	T	U	V	W	X	Y	Z
MAXIMUM VALUE	1.05	1.10	1.15	1.20	1.25	1.30	1.35	1.40



(11.)

DEPTH OF CRATER MEASURED FROM JPROJ

I	J OF CRATER BOTTOM	R	DEPTH IN CM, D(I)
1	20.4	1.9850-02	1.7399-01
2	20.6	5.9550-02	1.6264-01
3	20.4	9.9250-02	1.7633-01
4	20.1	1.3895-01	1.6342-01
5	20.2	1.7865-01	1.6594-01
6	20.2	2.1835-01	1.6736-01
7	20.1	2.5805-01	1.6258-01
8	19.6	2.9775-01	1.4321-01
9	18.5	3.3745-01	1.0159-01
10	16.8	3.7715-01	3.1806-02
11	14.9	4.1685-01	-4.3254-02
12	13.6	4.5655-01	-8.1702-02
13	14.1	4.9625-01	-7.8552-02
14	14.8	5.3595-01	-4.9472-02
15	15.3	5.7565-01	-2.4230-02
16	15.6	6.1535-01	-1.6319-02
17	15.5	6.5505-01	-7.2039-03
18	15.9	6.9475-01	-3.1113-03
19	16.0	7.3445-01	-1.3145-03
20	16.0	7.7415-01	-6.1621-04
21	16.0	8.1385-01	-1.9196-04
22	16.0	8.5355-01	-4.1127-05
23	16.0	8.9325-01	-9.1419-06

CRATER VOLUME
6.2737-02

CRATER VOLUME BASED ON (2/3) * PI * D(1)**3
1.1032-02

(12.)

I = 1 R(I) = .040 DR(I) = .3970000-01

J	U	V	P	AMA	RHO	AIF	COMP	Z
46	-0.000000	-0.000000	0.000000	5.307440-04	2.700000+00	-0.000000	1.000000+00	1.826700+00
45	-0.000000	-0.000000	0.000000	5.307440-04	2.700000+00	-0.000000	1.000000+00	1.746800+00
44	-0.000000	7.357571-01	0.000000	5.307448-04	2.700004+00	-0.000000	1.000001+00	1.746800+00
43	1.953747-01	4.320798+00	9.412925+06	5.307507-04	2.700034+00	1.012434+01	1.000013+00	1.707109+00
42	1.742703-01	2.196922+01	4.051160+07	5.307726-04	2.700145+00	2.208105+02	1.000054+00	1.667400+00
41	1.539405+00	9.450982+01	1.732542+08	5.308603-04	2.700622+00	4.323703+03	1.000234+00	1.627700+00
40	8.010741+00	4.029706+02	7.217932+08	5.312527-04	2.702586+00	7.767690+04	1.000953+00	1.588000+00
39	2.291742+01	1.044659+03	2.913524+09	5.327881-04	2.710399+00	1.433507+06	1.003451+00	1.548300+00
38	4.959536+01	6.376933+03	1.099897+10	5.383272-04	2.734577+00	2.090943+07	1.014288+00	1.508600+00
37	3.350307+02	2.214134+04	3.771528+10	5.552879-04	2.824859+00	2.484650+08	1.046244+00	1.468900+00
36	1.042815+03	6.008044+04	1.041870+11	5.909807-04	3.006476+00	1.465623+09	1.113510+00	1.429200+00
35	2.333843+03	1.088163+05	1.915145+11	6.294958-04	3.202370+00	4.469485+09	1.146063+00	1.389500+00
34	3.703115+03	1.321037+05	2.098914+11	6.365202-04	3.238134+00	5.928257+09	1.199309+00	1.349800+00
33	4.049302+03	1.243888+05	1.645387+11	6.193575-04	3.150794+00	3.220242+09	1.166961+00	1.310100+00
32	3.751942+03	1.174583+05	1.242793+11	6.028731-04	3.066935+00	1.254604+09	1.135902+00	1.270400+00
31	3.032057+03	1.211031+05	1.106218+11	6.000588-04	3.052618+00	2.042385+08	1.130599+00	1.230700+00
30	4.088259+03	1.376011+05	1.153654+11	6.035482-04	3.070369+00	-5.420134+08	1.137174+00	1.191000+00
29	4.259260+03	1.539992+05	1.124955+11	6.018959-04	3.061963+00	-1.400760+09	1.134061+00	1.151300+00
28	4.860599+03	1.882008+05	1.020793+11	5.958797-04	3.031358+00	-2.051599+09	1.122725+00	1.111600+00
27	6.564849+03	1.893803+05	8.726971+10	5.907145-04	3.005082+00	-8.423065+08	1.112903+00	1.071900+00
26	5.351507+03	2.250674+05	5.335653+10	5.830847-04	2.966267+00	2.091332+09	1.098617+00	1.032200+00
25	6.828365+03	2.094380+05	7.942581+10	5.653011-04	2.875799+00	4.104941+09	1.065111+00	9.924099-01
24	3.928348+03	3.175320+05	6.214010+10	5.493360-04	2.794581+00	6.403131+09	1.033032+00	9.527099-01
23	2.719302+03	3.898200+05	1.803368+10	5.194259-04	2.642422+00	6.615177+09	9.786750-01	9.130099-01
22	7.010433+03	4.280370+05	0.000000	4.550866-04	2.315126+00	5.384096+09	8.574360-01	8.733099-01
21	7.870803+03	4.632327+05	0.000000	4.115700-04	2.093738+00	4.255820+09	7.758546-01	8.336099-01
20	8.139330+03	4.699922+05	0.000000	3.030748-06	1.541802-02	4.091441+09	5.710376-03	7.939099-01

*** ERROR EXIT - SEE STATEMENT NUMBER 65 IN CDT

(13.)

I= 8 J= 16 K=705

Z-BLOCK

I	REAL FORMAT Z(I)	INTEGER FORMAT Z(I)
1	.137350+02	17831957620
2	-.100000+01	-17381195776
3	.000000	0
4	.000000	0
5	.500000-38	5
6	.500000-38	1
7	.000000	0
8	.314159+01	17553718998
9	.000000	0
10	.000000	0
11	.240000+11	21971239664
12	.500000-38	2
13	.000000	0
14	.000000	0
15	.270000+01	17536901606
16	.000000	0
17	.000000	0
18	.166740+01	17425984211
19	.500000-38	1
20	.500000-38	2
21	.970000-00	17310060380
22	.000000	0
23	.000000	0
24	.100000-02	16040629108
25	.500000-38	28
26	.000000	0
27	-.100000+01	-17381195776
28	.410000+11	22091785517
29	.700000+10	21718429208
30	-.500000-38	-1
31	.000000	0
32	.000000	0
33	.500000-38	42
34	.500000-38	43
35	.600000-38	54
36	.600000-38	55
37	.540000-38	2269
38	.540000-38	2270
39	.000000	0
40	.000000	0
41	.000000	0
42	.239000+10	21549523980
43	.500000-38	10
44	.100000-05	14700101096

4.2. UNIVAC 1108 CONTROL CARDS

```
∇ ASG P = 1184                          Program Tape
∇ ASG 7 = 2213                          Restart Tape*
∇ XQT CUR
  IN
  TRI P
  TOC
∇P HDG                  STANDARD CRATER
∇ XQT RPM
```

STANDARD CRATER

```
1        151      1      13.735
.        .        .      .
.        .        .      .
         65       1      1.0
.        .        .      .
.        .        .      .
1        50       1      1.0       -06
```

} Input deck

---

* Can assign a drum area for tape dumps instead of a tape by using:

∇D ASG 7 = 100000

5. RESTART OF PROBLEMS

Periodically during a calculation RPM writes on tape the problem parameters and the current state of the material in each cell. By reading this tape the user can "restart" and continue a calculation from an intermediate point. Because the initial conditions are saved on tape, SETUP is not called and the following three cards are the only ones which must be in the restart input deck:

Description of Card	Col. 1	Col. 2-6	Col. 7	Col. 8-16	Col. 17-25	Col. 26-34
ID CARD				IMPACT		
PK(1),PK(2), PK(3)	1	151	3	3.2	88.	-1.
TSTOP	1	50	1	1. - 06		

PK(1) = The same problem number used when the problem was initially setup. (NOTE: This can be any number between 00.0000 and 99.9999.)

PK(2) = The restart cycle number. The problem can be restarted on any cycle which is marked as a tape dump cycle in the printed output.

PK(3) = The restart flag. If it is -1. EDIT makes a long print of the restart cycle. However the user may wish to avoid making a long print on the restart cycle. He can do this by setting PK(3) = -2.

TSTOP = The new time at which execution will stop.

Other variables besides TSTOP may be redefined when restarting a problem. The copy of the "Setup Sheet for OIL-RPM Restarts" on page 44 lists those variables most likely to be redefined at an intermediate point in the calculation.

The cards in a restart input deck can be in any order as long as the first card is the ID card, the second card defines PK(1), PK(2) and PK(3) and has a "1" in column one, and the last card has a "1" in column one.

Examples:

1.

Description of Card	Col. 1	Col. 2-6*	Col. 7	Col. 8-16	Col. 17-25	Col. 26-34
ID CARD				IMPACT 1		
PK(1),PK(2) PK(3)	1	151	3	26.1	32.	-1.
INTER	2	87	1	0.		
RHOMIN		138	1	1. -03		
ICSTOP	2	7	1	135.		
TSTOP	1	50	1	0.		

2.

Description of Card	Col. 1	Col. 2-6*	Col. 7	Col. 8-16	Col. 17-25	Col. 26-34
ID CARD				IMPACT 2		
PK(1),PK(2) PK(3)	1	151	3	35.013	2019.	-2.
TSTOP	1	50	1	1.0 -05		

3.

Description of Card	Col. 1	Col. 2-6*	Col. 7	Col. 8-16	Col. 17-25	Col. 26-34
ID CARD				IMPACT 3		
PK(1),PK(2) PK(3)	1	151	3	8.2	128.	-2.
ICSTOP	2	7	1	200.		
TSTOP	1	50	1	0.		

*All numbers must be right-adjusted within the field.

7-WORD DATA
SETUP SHEET FOR OIL-RPM "RESTART"

	2 LOC	7 N	8 1	17 2	26 3	35 4	44 5	6	62 7	71	73
					ID	CARD					
1	151	3				(a) Problem number. (b) Pickup cycle #. (c) Restart flag (-1. or -2.)					
	70	1		CYCPH3 = Number of times to subcycle PH3.							
	21	1		AMDM = Allowable expansion of material considered "solid": ~.95 to .99.							
	158	1		RHOMIN = Minimum ρ for influencing DT.							
	113	1		FINAL = Final value of "stability fraction."							
	24	1		DMIN = Allowable relative error for total energy sum: ~ 10^{-4}.							
	144	1		DTMIN = Minimum for DT.							
	55	1		VT = Minimum Δρ from one cell across a grid boundary to trigger rezone.							
	65	1		SN = 0., when negative S.I.E. is to be set to 0. If SN = 1., negative S.I.E. left alone.							
	75	1		EVAP = ρ/ρ_o of cells to be "evaporated."							
	110	1		ROEPS = Round-off epsilon.							
2	147	1		JPROJ = Used in EDIT print and crater depth calculation.							
2	5	1		NFRELP = Ratio of "short" print frequency to "long" print frequency.							
2	6	1		NDUMP7 = Ratio of print frequency to tape dump frequency.							
2	7	1		ICSTOP = Cycle to stop. Set to 0. if stopping on time.							
1	50	1		TSTOP = Time (value of T) to stop. Set to 0. if stopping on cycles. (This must be the last card of a restart deck.)							

6. LISTING OF THE FORTRAN IV PROGRAM

This section consists simply of a listing of the FORTRAN IV program.

```
C     RPM
C     ***************************************************************
C
C
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),
     3          GAMC(102),
     4          PK(15),   Z(150)   ,
     5          XP(26,51),YP(26,51),
     6          PL(204)  ,UL(204)  ,PR(204)  ,
     7           RSN(52),   RST(52),
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)
C            *** DIMENSIONED ARRAYS
C            *** Z-BLOCK IS SAVED ON TAPE.
      COMMON     Z
      COMMON    PK
      COMMON  YY,      XX
      COMMON  DDX,     DDY
      COMMON  AMX,     AIX,     U,     V,     P
      COMMON  TAU,     JPM
      COMMON  UL ,     PL
      COMMON  XP ,     YP,      CMXP, CMYP
C            *** NON-DIMENSIONED VARIABLES
      COMMON         AID    ,AMMV   ,AMMY   ,AMPY   ,AMUR   ,AMUT   ,AMVR   ,
     1AMVT   ,DELTL ,DELER ,DELET ,DELM   ,DTODX ,DXYMIN,EAMMP ,EAMPY ,
     2E      ,ERDUMP,I      ,I3     ,IWS    ,J      ,K      ,KA     ,KB     ,
     3LL     ,MD    ,ME     ,MZT    ,NERR   ,NK     ,NPRINT,
     4NR     ,NRZ   ,NULLE  ,PIDTS  ,SIEMIN,SNR    ,SNT    ,STR    ,SOLID  ,
     5SUM    ,TESTRH,TWOPI  ,URR    ,WS     ,WSA    ,WSB    ,WSC    ,WFLAGF,
     6WFLAGL,WFLAGP
      COMMON    LAST
C
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE
C                X(0), Y(0), DX(0), DY(0)
C
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))
C
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY
C
      EQUIVALENCE          (UL,FLEFT),          (UL(103),YAMC),
     1                     (PL,GAMC,PR),        (PL(103),SIGC)
C
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY
C
      EQUIVALENCE          (UL,RSN),
     1                     (PL,RST),            (P,UK),
     2                     (P(157),VK),         (P(313),SNB),
     3                     (P(365),STB),        (P(417),RHO)
C
C            *** SPECIAL EQUIVALENCES FOR EDIT
C
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)
C
C            *** Z-STORAGE EQUIVALENCES
```

```
C
      EQUIVALENCE                          (Z(  1),PROB   ),(Z(  2),CYCLE ),
     1(Z(  3),DY     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),
     2(Z(  7),ICSTOP),(Z(  8),PIDY   ), (Z(  9),TOPMU ),(Z( 10),RTMU  ),
     3(Z( 11),STK1   ),(Z( 12),NUMREZ), (Z( 13),ETH   ),(Z( 14),UN14  ),
     4(Z( 15),RHINIT),(Z( 16),PROJI ), (Z( 17),UN17  ),(Z( 18),XMAX  ),
     5(Z( 19),NZ     ),(Z( 20),NREZ  ), (Z( 21),AMDM  ),(Z( 22),UVMAX ),
     6(Z( 23),UN23   ),(Z( 24),DMIN  ), (Z( 25),JSTR  ),(Z( 26),DTNA  ),
     7(Z( 27),CVIS   ),(Z( 28),STK2  ), (Z( 29),STEZ  ),(Z( 30),NC    ),
     8(Z( 31),UN31   ),(Z( 32),NRC   ), (Z( 33),IMAX  ),(Z( 34),IMAXA ),
     9(Z( 35),JMAX   ),(Z( 36),JMAXA ), (Z( 37),KMAX  ),(Z( 38),KMAXA )
      EQUIVALENCE
     1(Z( 39),BOTM   ),(Z( 40),BOTMV ), (Z( 41),NUMSPT),(Z( 42),CZERO ),
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ), (Z( 45),PRDELT),(Z( 46),PRFACT)
      EQUIVALENCE
     1(Z( 47),I1     ),(Z( 48),I2    ), (Z( 49),IPCYCL),(Z( 50),TSTOP ),
     2(Z( 51),RHOFIL),(Z( 52),TARGV ), (Z( 53),N3    ),(Z( 54),IVARDY),
     3(Z( 55),VT     ),(Z( 56),N6    ), (Z( 57),RTM   ),(Z( 58),RTMV  ),
     4(Z( 59),UN59   ),(Z( 60),N10   ), (Z( 61),N11   ),(Z( 62),GAMMA ),
     5(Z( 63),TOPM   ),(Z( 64),BOTMU ), (Z( 65),SN    ),(Z( 66),TOPMV ),
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP), (Z( 69),PRXRT ),(Z( 70),CYCPH3),
     7(Z( 71),REZFCT),(Z( 72),TARGI ), (Z( 73),PROJU ),(Z( 74),BBOUND),
     8(Z( 75),EVAP   ),(Z( 76),ECK   ), (Z( 77),NECYCL),(Z( 78),II    ),
     9(Z( 79),JJ     ),(Z( 80),NMP   ), (Z( 81),Y2    ),(Z( 82),EZPH1 )
      EQUIVALENCE
     1(Z( 83),IVARDX),(Z( 84),T      ), (Z( 85),NMPMAX),(Z( 86),PMIN  ),
     2(Z( 87),INTER ),(Z( 88),TAYBOT), (Z( 89),TAYTOP),(Z( 90),IEMAP ),
     3(Z( 91),MC     ),(Z( 92),MR    ), (Z( 93),MZ    ),(Z( 94),MB    )
      EQUIVALENCE
     1(Z( 95),REZ    ),(Z( 96),NODUMP), (Z( 97),UN97  ),(Z( 98),UN98  ),
     2(Z( 99),UN99   ),(Z(100),EVAPM ), (Z(101),EVAPEN),(Z(102),EVAPMU),
     3(Z(103),EVAPMV),(Z(104),EZPH2 ), (Z(105),SNL   ),(Z(106),STL   ),
     4(Z(107),TAXRT ),(Z(108),IDNMAP), (Z(109),IPRMAP),(Z(110),ROEPS ),
     5(Z(111),RHINI ),(Z(112),VINI  ), (Z(113),FINAL ),(Z(114),IVMAP ),
     6(Z(115),RHOZ  ),(Z(116),ESA   ), (Z(117),ESEZ  ),(Z(118),ESB   ),
     7(Z(119),ESCAPA),(Z(120),ESESP ), (Z(121),ESESQ ),(Z(122),ESES  ),
     8(Z(123),ESALPH),(Z(124),ESBETA), (Z(125),ESCAPB),(Z(126),IUMAP ),
     9(Z(127),SS1    ),(Z(128),SS2   ), (Z(129),UMIN  ),(Z(130),SS4   )
      EQUIVALENCE
     1(Z(131),PRTIME),(Z(132),EOR    ), (Z(133),EOT   ),(Z(134),EOB   ),
     2(Z(135),EMOR   ),(Z(136),DXF   ), (Z(137),DYF   ),(Z(138),RHOMIN),
     3(Z(139),STAB), (Z(140),XIENRG),  (Z(141),XKENRG), (Z(142),XTENRG),
     4(Z(143),STT    ),(Z(144),DTMIN ), (Z(145),TRNSFC),(Z(146),EMOT  ),
     5(Z(147),JPROJ ),(Z(148),CNAUT ), (Z(149),BBAR  ),(Z(150),EMOB  )
C
C     ..............................................................
C
C     END OF COMMON
C
C     ..............................................................
C
C           *** INIALIZE BLANK COMMON
      LAST = 1
      IQ = 0
5     IQ = IQ+1
      Z(IQ) = 0.
      IF(LAST.NE.0) GO TO 5
```

```
10    CALL INPUT
20    CALL CDT
      CALL EDIT
C           *** ASK WFLAGL WHETHER THIS IS THE LAST CYCLE.
C               WFLAGL IS SET IN EDIT.
      IF (WFLAGL.GT.0.) GO TO 40
      CALL PH1
C           *** NPRINT=1 ON EDIT PRINT CYCLES.
C               INTER.NE.0 WHEN INTERMEDIATE EDIT PRINTS ARE WANTED.
      IF (INTER.NE.0.AND.NPRINT.EQ.1) CALL EDIT
C           *** CYCPH3=-1. WHEN PHASE 3 IS NOT USED. OTHERWISE,
C               CYCPH3=NUMBER OF TIMES PHASE 3 CALCULATIONS ARE
C               SUBCYCLED.
      IF (CYCPH3.EQ.(-1.)) GO TO 30
      CALL PH3
      IF (INTER.NE.0.AND.NPRINT.EQ.1) CALL EDIT
30    CALL PH2
      GO TO 20
40    CALL EXIT
      END
```

```
      SUBROUTINE INPUT                                                  INP  10
C     ..................................................................INP  20
C                                                                       INP  30
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,      INP  40
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,               INP  50
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),  INP  60
     3          GAMC(102),                                             INP  70
     4          PK(15),   Z(150)   ,                                   INP  80
     5          XP(26,51),YP(26,51),                                   INP  90
     6          PL(204)  ,UL(204)  ,PR(204)  ,                         INP 100
     7           RSN(52),   RST(52),                                   INP 110
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,               INP 120
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,               INP 130
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)      INP 140
C            *** DIMENSIONED ARRAYS                                     INP 150
C            *** Z-BLOCK IS SAVED ON TAPE.                              INP 160
      COMMON     Z                                                     INP 170
      COMMON    PK                                                     INP 180
      COMMON  YY,      XX                                              INP 190
      COMMON  DDX,     DDY                                             INP 200
      COMMON  AMX,     AIX,     U,     V,     P                       INP 210
      COMMON  TAU,     JPM                                             INP 220
      COMMON  UL ,     PL                                              INP 230
      COMMON  XP ,     YP,      CMXP, CMYP                             INP 240
C            *** NON-DIMENSIONED VARIABLES                              INP 250
      COMMON         AID    ,AMMV  ,AMMY  ,AMPY  ,AMUR  ,AMUT  ,AMVR  ,INP 260
     1AMVT  ,DELEB ,DELER ,DELET ,DELM  ,DTODX ,DXYMIN,EAMMP ,EAMPY ,   INP 270
     2E     ,ERDUMP,I     ,I3    ,IWS   ,J     ,K     ,KA    ,KB    ,   INP 280
     3LL    ,MD    ,ME    ,MZT   ,NERR  ,NK    ,NPRINT,                INP 290
     4NR    ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR   ,SNT   ,STR   ,SOLID ,  INP 300
     5SUM   ,TESTRH,TWOPI ,URR   ,WS    ,WSA   ,WSB   ,WSC   ,WFLAGF,   INP 310
     6WFLAGL,WFLAGP                                                     INP 320
C                                                                       INP 330
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE              INP 340
C                X(0), Y(0), DX(0), DY(0)                               INP 350
C                                                                       INP 360
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))                         INP 370
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))                     INP 380
C                                                                       INP 390
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY                      INP 400
C                                                                       INP 410
      EQUIVALENCE          (UL,FLEFT),         (UL(103),YAMC),          INP 420
     1                     (PL,GAMC,PR),       (PL(103),SIGC)           INP 430
C                                                                       INP 440
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY                      INP 450
C                                                                       INP 460
      EQUIVALENCE          (UL,RSN),                                   INP 470
     1                     (PL,RST),           (P,UK),                 INP 480
     2                     (P(157),VK),        (P(313),SNB),           INP 490
     3                     (P(365),STB),       (P(417),RHO)            INP 500
C                                                                       INP 510
C            *** SPECIAL EQUIVALENCES FOR EDIT                          INP 520
C                                                                       INP 530
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)                             INP 540
C                                                                       INP 550
C            *** Z-STORAGE EQUIVALENCES                                 INP 560
C                                                                       INP 570
      EQUIVALENCE                      (Z(  1),PROB  ),(Z(  2),CYCLE ), INP 580
```

```
     1(Z(  3),DT     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  INP 590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ), (Z(  9),TOPMU ),(Z( 10),RTMU  ),  INP 600
     3(Z( 11),STK1  ),(Z( 12),NUMREZ), (Z( 13),ETH   ),(Z( 14),UN14  ),  INP 610
     4(Z( 15),RHINIT),(Z( 16),PROJI ), (Z( 17)-UN17  ),(Z( 18),XMAX  ),  INP 620
     5(Z( 19),NZ     ),(Z( 20),NREZ  ), (Z( 21),AMDM  ),(Z( 22),UVMAX ),  INP 630
     6(Z( 23),UN23   ),(Z( 24),DMIN  ), (Z( 25),JSTR  ),(Z( 26),DTNA  ),  INP 640
     7(Z( 27),CVIS   ),(Z( 28),STK2  ), (Z( 29),STEZ  ),(Z( 30),NC    ),  INP 650
     8(Z( 31),UN31   ),(Z( 32),NRC   ), (Z( 33),IMAX  ),(Z( 34),IMAXA ),  INP 660
     9(Z( 35),JMAX   ),(Z( 36),JMAXA ), (Z( 37),KMAX  ),(Z( 38),KMAXA )  INP 670
      EQUIVALENCE                                                        INP 680
     1(Z( 39),BOTM   ),(Z( 40),BOTMV ), (Z( 41),NUMSPT),(Z( 42),CZERO ),  INP 690
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ), (Z( 45),PRDELT),(Z( 46),PRFACT)   INP 700
      EQUIVALENCE                                                        INP 710
     1(Z( 47),I1     ),(Z( 48),I2    ), (Z( 49),IPCYCL),(Z( 50),TSTOP ),  INP 720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ), (Z( 53),N3    ),(Z( 54),IVARDY),  INP 730
     3(Z( 55),VT     ),(Z( 56),N6    ), (Z( 57),RTM   ),(Z( 58),RTMV  ),  INP 740
     4(Z( 59),UN59   ),(Z( 60),N10   ), (Z( 61),N11   ),(Z( 62),GAMMA ),  INP 750
     5(Z( 63),TOPM   ),(Z( 64),BOTMU ), (Z( 65),SN    ),(Z( 66),TOPMV ),  INP 760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP), (Z( 69),PRXRT ),(Z( 70),CYCPH3),  INP 770
     7(Z( 71),REZFCT),(Z( 72),TARGI ), (Z( 73),PROJU ),(Z( 74),BBOUND),  INP 780
     8(Z( 75),EVAP   ),(Z( 76),ECK   ), (Z( 77),NECYCL),(Z( 78),II    ),  INP 790
     9(Z( 79),JJ     ),(Z( 80),NMP   ), (Z( 81),Y2    ),(Z( 82),EZPH1 )  INP 800
      EQUIVALENCE                                                        INP 810
     1(Z( 83),IVARDX),(Z( 84),T      ), (Z( 85),NMPMAX),(Z( 86),PMIN  ),  INP 820
     2(Z( 87),INTER ),(Z( 88),TAYBOT), (Z( 89),TAYTOP),(Z( 90),IEMAP ),  INP 830
     3(Z( 91),MC     ),(Z( 92),MR    ), (Z( 93),MZ    ),(Z( 94),MB    )  INP 840
      EQUIVALENCE                                                        INP 850
     1(Z( 95),REZ    ),(Z( 96),NODUMP), (Z( 97),UN97  ),(Z( 98),UN98  ),  INP 860
     2(Z( 99),UN99   ),(Z(100),EVAPM ), (Z(101),EVAPEN),(Z(102),EVAPMU),  INP 870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ), (Z(105),SNL   ),(Z(106),STL   ),  INP 880
     4(Z(107),TAXRT ),(Z(108),IDNMAP), (Z(109),IPRMAP),(Z(110),ROEPS ),  INP 890
     5(Z(111),RHINI ),(Z(112),VINI  ), (Z(113),FINAL ),(Z(114),IVMAP ),  INP 900
     6(Z(115),RHOZ  ),(Z(116),ESA   ), (Z(117),ESEZ  ),(Z(118),ESB   ),  INP 910
     7(Z(119),ESCAPA),(Z(120),ESESP ), (Z(121),ESESQ ),(Z(122),ESES  ),  INP 920
     8(Z(123),ESALPH),(Z(124),ESBETA), (Z(125),ESCAPB),(Z(126),IUMAP  ),  INP 930
     9(Z(127),SS1    ),(Z(128),SS2   ), (Z(129),UMIN  ),(Z(130),SS4   )  INP 940
      EQUIVALENCE                                                        INP 950
     1(Z(131),PRTIME),(Z(132),EOR    ), (Z(133),EOT   ),(Z(134),EOB   ),  INP 960
     2(Z(135),EMOR   ),(Z(136),DXF   ), (Z(137),DYF   ),(Z(138),RHOMIN),  INP 970
     3(Z(139),STAB), (Z(140),XIENRG),  (Z(141),XKENRG), (Z(142),XTENRG),INP 980
     4(Z(143),STT    ),(Z(144),DTMIN ), (Z(145),TRNSFC),(Z(146),EMOT  ),  INP 990
     5(Z(147),JPROJ ),(Z(148),CNAUT ), (Z(149),BBAR  ),(Z(150),EMOB   )  INP1000
C                                                                        INP1010
C     ..............  ..............................................INP1020
C                                                                        INP1030
C     END OF COMMON                                                      INP1040
C                                                                        INP1050
C     ..............................................................INP1060
C                                                                        INP1070
C            *** MZT MUST EQUAL NUMBER OF WORDS IN Z-ARRAY.              INP1080
      MZT=150                                                            INP1090
C            *** SET WFLAGF=1. TO SAY THIS IS FIRST CYCLE OF THIS RUN.   INP1095
      WFLAGF=1.                                                          INP1100
C            *** READ AND PRINT ID HEADING CARD (FIRST CARD IN           INP1110
C                INPUT DECK)                                             INP1115
      READ (5,370) IWS                                                   INP1120
      WRITE (6,370) IWS                                                  INP1130
```

```
C             *** CARDS ROUTINE WILL READ AND PRINT FIRST DATA CARD.      INP1140
      CALL CARDS                                                          INP1150
C             *** PK(3).LT.0. MEANS THIS PROBLEM IS BEING RESTARTED FROM INP1152
C                 TAPE 7 AND SETUP IS NOT NEEDED.                         INP1155
      IF (PK(3).LT.0.) GO TO 70                                           INP1160
C             *** SINCE THE SETUP ROUTINE WRITES ON TAPE 7, MAKE          INP1170
C                 SURE THIS IS A SETUP AND NOT A RESTART RUN  TO AVOID    INP1180
C                 WRITING OVER A GOOD DUMP TAPE.                          INP1190
      CALL CARDS                                                          INP1190
C             *** Z(1)=PROB IS DEFINED BY THE SECOND CARD OF A SETUP      INP1200
C                 DECK,BUT IS NOT DEFINED IN A RESTART DECK.              INP1205
C                 THEREFORE IF Z(1)=0., THIS IS A RESTART RUN, AND IF     INP1210
C                 Z(1).NE.0.,THIS IS A SETUP RUN.                         INP1215
5     IF(PROB.EQ.0.)GO TO 230                                             INP1220
      CALL SETUP                                                          INP1230
      GO TO 70                                                            INP1240
10    CONTINUE                                                            INP1250
      CALL CARDS                                                          INP1260
C             *** INITIALIZE P-STORAGE.                                   INP1265
20    DO 30 K=1,KMAXA                                                     INP1270
30    P(K)=0.0                                                            INP1280
C             *** SET T AND NC SO THEY WILL EQUAL ZERO ON FIRST EDIT      INP1282
C                 PRINT AFTER BEING INCREMENTED BY CDT.                   INP1284
      T=T-DTNA                                                            INP1290
      NC=NC-1                                                             INP1300
C             *** CHECK FATAL INPUT ERRORS.                               INP1305
32    IF(RHOZ.LE.0.) GO TO 260                                            INP1310
34    IF(ESCAPA.LT.0.) GO TO 270                                          INP1320
36    IF(IMAX.EQ.0.OR.JMAX.EQ.0) GO TO 280                                INP1340
C             *** DEFINE CONSTANTS USED THROUGHOUT CALCULATION.           INP1345
      CNAUT=SQRT(ESCAPA/RHOZ)                                             INP1350
      WS=ESESP-ESES                                                       INP1360
      IF (WS.LE.0.) WS=1.                                                 INP1370
      SS1=1./WS                                                           INP1380
      TESTRH = .2*RHOZ                                                    INP1390
      CYCLE=NC                                                            INP1400
      NRZ=NREZ-NUMREZ                                                     INP1420
      SOLID=AMDM*RHOZ                                                     INP1430
      GAMMA=ESA+1.                                                        INP1440
      TWOPI=2.*PIDY                                                       INP1450
      PMIN=10.**6                                                         INP1460
      TRNSFC=.4                                                           INP1470
      VT=10.**(-5)                                                        INP1475
      SS2=1.                                                              INP1480
C             *** SET NUMBER OF SYMBOLS TO BE USED IN PRINTED CONTOUR     INP1482
C                 MAPS.                                                   INP1484
      IDNMAP=28.                                                          INP1490
      IPRMAP=26.                                                          INP1500
      IVMAP=26.                                                           INP1510
      IUMAP=26.                                                           INP1520
      IEMAP=26.                                                           INP1530
C             *** PRINT VALUES OF MOST INPUT PARAMETERS.                  INP1555
      WRITE (6,310) NUMREZ,JSTR,N6,IMAX,JMAX,I1,I2,JPROJ,NMPMAX,INTER,NUINP1560
     1MSCA,IPCYCL,ICSTOP,NFRELP,NDUMP7,NODUMP,IVARDX,IVARDY              INP1570
      WRITE(6,320) DXF,DYF,RHOMIN,TESTRH,RHOZ,RHINI,RHINIT,AMDM,SOLID, VINP1580
     1T,EVAP,ROEPS,SN,BBAR,CNAUT,FINAL,STAB,DMIN,CVIS,SS2,CYCPH3,CZERO,SINP1590
     2TK1,STK2,STEZ,ESA,ESB,ESCAPA,ESCAPB,ESALPH,ESBETA,ESEZ,ESES,ESESP,INP1600
```

```
      3ESESQ,REZFCT,SS4,Y2,TRNSFC,DTMIN,PRDELT,PRFACT,PRLIM,TSTOP        INP1610
C            *** PRINT DX,DY ARRAYS WHEN THE CELL DIMENSIONS ARE         INP1620
C                VARIABLE.                                               INP1625
      IF (IVARDX.EQ.0) GO TO 40                                          INP1630
      WRITE (6,330)                                                      INP1640
      WRITE (6,350) (I,DX(I),I=1,IMAX)                                   INP1650
40    IF (IVARDY.EQ.0) GO TO 50                                          INP1660
      WRITE (6,340)                                                      INP1670
      WRITE (6,350) (J,DY(J),J=1,JMAX)                                   INP1680
50    CONTINUE                                                           INP1690
C            *** WHEN T.GT.0., PROBLEM IS BEING RESTARTED.               INP1695
      IF (T.GT.0.) GO TO 60                                              INP1700
C            *** DEFINE  TIME OF FIRST EDIT PRINT AFTER CYCLE 0.         INP1705
      PRTIME=PRDELT                                                      INP1710
      GO TO 300                                                          INP1720
C            *** PRDELT = 0. WHEN PRINTING ON CYCLES RATHER TIME.        INP1725
60    IF (PRDELT.EQ.0.) GO TO 300                                        INP1730
C            *** DEFINE  TIME OF FIRST EDIT PRINT AFTER RESTART CYCLE.   INP1735
      IWS=T/PRDELT+1.                                                    INP1740
      PRTIME=FLOAT(IWS)*PRDELT                                           INP1750
      GO TO 300                                                          INP1760
C            *** READ DUMP TAPE 7.                                       INP1770
70    CONTINUE                                                           INP1800
      IWS=0                                                              INP1810
80    REWIND 7                                                           INP1820
90    READ (7) PR(1),PR(2),N3                                            INP1830
C            *** NR = NUMBER OF RECORDS WRITTEN BY EACH TAPE DUMP.       INP1832
C                WHEN N3=1,TRACER POINTS ARE BEING USED AND MAKE UP      INP1834
C                ANOTHER RECORD IN EACH TAPE DUMP.                       INP1836
      NR=N3+7                                                            INP1840
C            *** FIRST WORD OF FIRST RECORD OF EACH DUMP SHOULD BE       INP1842
C                555.0. TEST THIS THREE TIMES BEFORE EXITING.            INP1844
      IF (PR(1)-555.0) 100,110,100                                       INP1850
100   IWS=IWS+1                                                          INP1860
      IF (MOD(IWS,3)) 220,220,80                                         INP1870
110   IF (PR(2)) 100,120,120                                             INP1880
C            *** WHEN SETTING UP A PROBLEM PR(2) = PK(2) = 0. WHEN       INP1882
C                RESTARTING A PROBLEM, TAPE 7 IS READ UNTIL              INP1884
C                PR(2).GE.PK(2), THE RESTART CYCLE NUMBER.               INP1886
120   IF (PK(2)-PR(2)) 150,150,130                                       INP1890
130   DO 140 L=2,NR                                                      INP1900
140   READ (7)                                                           INP1910
      GO TO 90                                                           INP1920
150   READ (7) (Z(I),I=1,MZT)                                            INP1930
C            *** MAKE SURE PROBLEM NUMBER ON TAPE (PROB) MATCHES         INP1932
C                PROBLEM NUMBER ON INPUT CARDS (PK(1)).                  INP1934
      IF (ABS(PROB-PK(1))-.01) 160,160,210                               INP1940
160   READ (7) (U(I),V(I),AMX(I),AIX(I),P(I),I=1,KMAXA)                  INP1950
      READ (7) X(0),(X(I),TAU(I),JPM(I),I=1,IMAX)                        INP1960
      READ (7) (Y(I),I=0,JMAX)                                           INP1970
C            *** Y2=-1. WHEN TRACER POINTS ARE USED.                     INP1980
      IF (Y2.GT.(-1.)) GO TO 170                                         INP1990
      READ (7) ((XP(I,J),YP(I,J),I=1,II),J=1,JJ)                         INP2000
170   READ (7) (DX(I),I=1,IMAX)                                          INP2010
      READ (7) (DY(J),J=1,JMAX)                                          INP2020
      READ (7) PR(1),PR(2),PR(3)                                         INP2030
C            *** THE FIRST WORD OF THE LAST RECORD OF EACH DUMP SHOULD   INP2032
```

```
      3ESESQ,REZFCT,SS4,Y2,TRNSFC,DTMIN,PRDELT,PRFACT,PRLIM,TSTOP
C            *** PRINT DX,DY ARRAYS WHEN THE CELL DIMENSIONS ARE
C                VARIABLE.
      IF (IVARDX.EQ.0) GO TO 40
      WRITE (6,330)
      WRITE (6,350) (I,DX(I),I=1,IMAX)
40    IF (IVARDY.EQ.0) GO TO 50
      WRITE (6,340)
      WRITE (6,350) (J,DY(J),J=1,JMAX)
50    CONTINUE
C            *** WHEN T.GT.0., PROBLEM IS BEING RESTARTED.
      IF (T.GT.0.) GO TO 60
C            *** DEFINE  TIME OF FIRST EDIT PRINT AFTER CYCLE 0.
      PRTIME=PRDELT
      GO TO 300
C            *** PRDELT = 0. WHEN PRINTING ON CYCLES RATHER TIME.
60    IF (PRDELT.EQ.0.) GO TO 300
C            *** DEFINE  TIME OF FIRST EDIT PRINT AFTER RESTART CYCLE.
      IWS=T/PRDELT+1.
      PRTIME=FLOAT(IWS)*PRDELT
      GO TO 300
C            *** READ DUMP TAPE 7.
70    CONTINUE
      IWS=0
80    REWIND 7
90    READ (7) PR(1),PR(2),N3
C            *** NR = NUMBER OF RECORDS WRITTEN BY EACH TAPE DUMP.
C                WHEN N3=1,TRACER POINTS ARE BEING USED AND MAKE UP
C                ANOTHER RECORD IN EACH TAPE DUMP.
      NR=N3+7
C            *** FIRST WORD OF FIRST RECORD OF EACH DUMP SHOULD BE
C                555.0. TEST THIS THREE TIMES BEFORE EXITING.
      IF (PR(1)-555.0) 100,110,100
100   IWS=IWS+1
      IF (MOD(IWS,3)) 220,220,80
110   IF (PR(2)) 100,120,120
C            *** WHEN SETTING UP A PROBLEM PR(2) = PK(2) = 0. WHEN
C                RESTARTING A PROBLEM, TAPE 7 IS READ UNTIL
C                PR(2).GE.PK(2), THE RESTART CYCLE NUMBER.
120   IF (PK(2)-PR(2)) 150,150,130
130   DO 140 L=2,NR
140   READ (7)
      GO TO 90
150   READ (7) (Z(I),I=1,MZT)
C            *** MAKE SURE PROBLEM NUMBER ON TAPE (PROB) MATCHES
C                PROBLEM NUMBER ON INPUT CARDS (PK(1)).
      IF (ABS(PROB-PK(1))-.01) 160,160,210
160   READ (7) (U(I),V(I),AMX(I),AIX(I),P(I),I=1,KMAXA)
      READ (7) X(0),(X(I),TAU(I),JPM(I),I=1,IMAX)
      READ (7) (Y(I),I=0,JMAX)
C            *** Y2=-1. WHEN TRACER POINTS ARE USED.
      IF (Y2.GT.(-1.)) GO TO 170
      READ (7) ((XP(I,J),YP(I,J),I=1,II),J=1,JJ)
170   READ (7) (DX(I),I=1,IMAX)
      READ (7) (DY(J),J=1,JMAX)
      READ (7) PR(1),PR(2),PR(3)
C            *** THE FIRST WORD OF THE LAST RECORD OF EACH DUMP SHOULD
```

```
C                 BE 555.0 OR 666.0.                                    INP2034
175   IF(PR(1)-555.0) 240,10,180                                        INP2040
180   IF(PR(2)-666.0) 250,10,250                                        INP2050
200   CALL CARDS                                                        INP2070
      CALL SETUP                                                        INP2080
      GO TO 20                                                          INP2090
C            *** PROBLEM NUMBER ON TAPE 7 NOT THE SAME AS PROBLEM       INP2092
C                NUMBER ON INPUT CARDS.                                 INP2094
210   NK=150                                                            INP2100
      GO TO 290                                                         INP2110
C            *** CANNOT FIND FIRST WORD OF FIRST RECORD.                INP2115
220   NK=100                                                            INP2120
      GO TO 290                                                         INP2130
C            *** NOT A RESTART AND YET Z(1) = 0.                        INP2135
230   NK=5                                                              INP2140
      GO TO 290                                                         INP2150
C            *** FIRST WORD OF LAST RECORD IS INCORRECT.                INP2155
240   NK=175                                                            INP2160
      GO TO 290                                                         INP2170
C            *** FIRST WORD OF LAST RECORD IS INCORRECT.                INP2175
250   NK=180                                                            INP2180
      GO TO 290                                                         INP2190
C                *** RHOZ.LE.0.                                         INP2200
260   NK=32                                                             INP2210
      GO TO 290                                                         INP2220
C                *** ESCAPA.LT.0.                                       INP2230
270   NK=34                                                             INP2240
      GO TO 290                                                         INP2250
C                 *** IMAX OR JMAX IS ZERO                              INP2260
280   NK=36                                                             INP2270
290   NR=1                                                              INP2280
C            *** PRINT FIRST THREE WORDS OF DUMP (PR(1),PR(2),N3)       INP2282
C                AND Z(151),Z(152),Z(153).                              INP2284
      WRITE(6,360) PR(1), Z(151), PR(2), Z(152), N3, Z(153)
      CALL ERROR                                                        INP2300
300   RETURN                                                            INP2310
C                                                                       INP2320
310   FORMAT (//12X,9H  NUMREZ=,I2,7H  JSTR=,I3,5H  N6=I3,7H  IMAX=,I3,7INP2330
     1H  JMAX=,I3,5H  I1=,I3,5H  I2=,I3,8H  JPROJ=,I3,9H  NMPMAX=,I5,8H INP2340
     2 INTER=,I2,9H  NUMSCA=,I2,/8H IPCYCL=,I3,9H  ICSTOP=,I4,9H  NFRELPINP2350
     3=,I3,9H  NDUMP7=,I3,9H  NODUMP=,I2,9H  IVARDX=,I2,9H  IVARDY=,I2//INP2360
     4)                                                                 INP2370
320   FORMAT (1X,120H          DXF          DYF       RHOMIN       TESTRH   INP2380
     1      RHOZ        RHINI       RHI[illegible]          AMDM        SOLID          INP2390
     2 VT/1X,1P10E12.4//1X,120H                    .PS          SN      INP2400
     3   BBAR        CNAUT        FINAL         STAB        DMIN        CVIINP2410
     4S          SS2/1X,1P10E12.4//1X,120H       CYCPH3        CZERO        INP2420
     5 STK1         STK2         STEZ          ESA          ESB       ESCAPA INP2430
     6     ESCAPB       ESALPH/1X,1P10E12.4//1X,96H       ESBETA         ESINP2440
     7EZ         ESES        ESESP        ESESQ       REZFCT          SS4    INP2450
     8      Y2/1X,1P8E12.4//1X,72H       TRNSFC        DTMIN       PRDELT  INP24[illegible]
     9    PRFACT        PRLIM        TSTOP/1X,1P6E12.4)                    INP2470
330   FORMAT (//7(3H  I,6X,2HDR,7X))                                    INP2480
340   FORMAT (//7(3H  J,6X,2HDZ,7X))                                    INP2490
350   FORMAT (7(I4,2X,1PE9.3,3X))                                       INP2500
360   FORMAT (1H1,5X,72H*** CHECK FIRST RECORD OF THE DUMP AND FIRST DAT
     1A CARD OF THE INPUT DECK // 4X,7HON TAPE,41X,8HON CARDS / 4X,
```

```
      24HWS =,F6.1,4X,7H(555.0),24X, 8HZ(151) =,F8.4,3X,16H(PROBLEM NUMBE
      3R) / 8H CYCLE =,F6.1,4X,18H(CYCLE BEING READ),13X, 8HZ(152) =F5.1,
      46X,15H(RESTART CYCLE) / 4X,4HN3 =,I4,6X,19H(TRACER POINT FLAG) ,
      512X, 8HZ(153) =,F5.1,6X,14H(RESTART FLAG))
  370  FORMAT (I1,71H                                                    INP2520
      1                    )                                             INP2530
       END                                                               INP2540-
```

```
      SUBROUTINE CARDS                                                  CRD  10
      DIMENSION TABLE(1),CARD(7),LABLE(1)                               CRD  20
      DIMENSION INPERR(1)                                               CRD  30
      COMMON      TABLE                                                 CRD  40
      EQUIVALENCE(TABLE(1),LABLE(1))                                    CRD  50
      INPERR=0                                                          CRD  60
      WRITE (6,80)                                                      CRD  70
 10   READ (5,90) IEND,LOC,NUMWPC,(CARD(I),I=1,NUMWPC)                  CRD  80
      WRITE (6,100) IEND,LOC,NUMWPC,(CARD(I),I=1,NUMWPC)                CRD  90
      IF (NUMWPC.LT.1) GO TO 50                                         CRD 100
      IF (LOC.LT.1) GO TO 70                                            CRD 110
      DO 30 I=1,NUMWPC                                                  CRD 120
      J=LOC+I-1                                                         CRD 130
      IF (IEND.NE.2) GO TO 20                                           CRD 140
      LABLE(J)=IFIX(CARD(I))                                            CRD 150
      GO TO 30                                                          CRD 160
 20   TABLE(J)=CARD(I)                                                  CRD 170
 30   CONTINUE                                                          CRD 180
 40   IF (IEND.NE.1) GO TO 10                                           CRD 190
      IF (INPERR.EQ.0) RETURN                                           CRD 200
      STOP                                                              CRD 210
 50   IF (LOC.NE.0) GO TO 70                                            CRD 220
      DO 60 I=1,7                                                       CRD 230
      IF (CARD(I).NE.0.) GO TO 70                                       CRD 240
 60   CONTINUE                                                          CRD 250
      WRITE (6,120)                                                     CRD 260
      GO TO 40                                                          CRD 270
 70   WRITE (6,110)                                                     CRD 280
      INPERR=1                                                          CRD 290
      GO TO 40                                                          CRD 300
C                 FORMATS                                               CRD 310
C                                                                       CRD 320
 80   FORMAT (/18H        INPUT CARDS///)                               CRD 330
 90   FORMAT (I1,I5,I1,0P7E9.4)                                         CRD 340
 100  FORMAT (1H I4,I7,I3,1P7E14.6)                                     CRD 350
 110  FORMAT (//42H **** ERROR ON PRECEDING DATA CARD *******/)         CRD 360
 120  FORMAT (//18H BLANK CARD ******/)                                 CRD 370
      END                                                               CRD 380-
```

```
      SUBROUTINE SETUP                                                  SET  10
C     ..................................................................SET  20
C                                                                       SET  30
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,      SET  40
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,               SET  50
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),  SET  60
     3          GAMC(102),                                             SET  70
     4          PK(15),   Z(150)   ,                                   SET  80
     5          XP(26,51),YP(26,51),                                   SET  90
     6          PL(204)  ,UL(204)  ,PR(204)  ,                         SET 100
     7           RSN(52),   RST(52),                                   SET 110
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,               SET 120
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,               SET 130
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)      SET 140
C           *** DIMENSIONED ARRAYS                                      SET 150
C           *** Z-BLOCK IS SAVED ON TAPE.                               SET 160
      COMMON     Z                                                     SET 170
      COMMON    PK                                                     SET 180
      COMMON  YY,      XX                                              SET 190
      COMMON  DDX,     DDY                                             SET 200
      COMMON  AMX,     AIX,     U,     V,     P                       SET 210
      COMMON  TAU,     JPM                                             SET 220
      COMMON  UL ,     PL                                              SET 230
      COMMON  XP ,     YP,      CMXP, CMYP                             SET 240
C           *** NON-DIMENSIONED VARIABLES                               SET 250
      COMMON         AID    ,AMMV  ,AMMY  ,AMPY  ,AMUR  ,AMUT  ,AMVR  ,SET 260
     1AMVT  ,DELEB ,DELER ,DELET ,DELM  ,DTODX ,DXYMIN,EAMMP ,EAMPY ,   SET 270
     2E     ,ERDUMP,I     ,I3    ,IWS   ,J     ,K     ,KA    ,KB    ,   SET 280
     3LL    ,MD    ,ME    ,MZT   ,NERR  ,NK    ,NPRINT,                SET 290
     4NR    ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR   ,SNT   ,STR   ,SOLID ,  SET 300
     5SUM   ,TCSTRH,TWOPI ,URR   ,WS    ,WSA   ,WSB   ,WSC   ,WFLAGF,   SET 310
     6WFLAGL,WFLAGP                                                     SET 320
C                                                                       SET 330
C           *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE               SET 340
C               X(0), Y(0), DX(0), DY(0)                                SET 350
C                                                                       SET 360
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))                         SET 370
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))                     SET 380
C                                                                       SET 390
C           *** SPECIAL EQUIVALENCES FOR PH2 ONLY                       SET 400
C                                                                       SET 410
      EQUIVALENCE          (UL,FLEFT),         (UL(103),YAMC),          SET 420
     1                     (PL,GAMC,PR),       (PL(103),SIGC)           SET 430
C                                                                       SET 440
C           *** SPECIAL EQUIVALENCES FOR PH3 ONLY                       SET 450
C                                                                       SET 460
      EQUIVALENCE          (UL,RSN),                                   SET 470
     1                     (PL,RST),           (P,UK),                 SET 480
     2                     (P(157),VK),        (P(313),SNB),           SET 490
     3                     (P(365),STB),       (P(417),RHO)            SET 500
C                                                                       SET 510
C           *** SPECIAL EQUIVALENCES FOR EDIT                           SET 520
C                                                                       SET 530
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)                             SET 540
C                                                                       SET 550
C           *** Z-STORAGE EQUIVALENCES                                  SET 560
C                                                                       SET 570
      EQUIVALENCE                        (Z(  1),PROB  ),(Z(  2),CYCLE ), SET 580
```

```
     1(Z(  3),DT     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  SET 590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ), (Z(  9),TOPMU ),(Z( 10),RTMU  ), SET 600
     3(Z( 11),STK1  ),(Z( 12),NUMREZ), (Z( 13),ETH   ),(Z( 14),UN14  ), SET 610
     4(Z( 15),RHINIT),(Z( 16),PROJI ), (Z( 17),UN17  ),(Z( 18),XMAX  ), SET 620
     5(Z( 19),NZ    ),(Z( 20),NREZ  ), (Z( 21),AMDM  ),(Z( 22),UVMAX ), SET 630
     6(Z( 23),UN23  ),(Z( 24),DMIN  ), (Z( 25),JSTR  ),(Z( 26),DTNA  ), SET 640
     7(Z( 27),CVIS  ),(Z( 28),STK2  ), (Z( 29),STEZ  ),(Z( 30),NC    ), SET 650
     8(Z( 31),UN31  ),(Z( 32),NRC   ), (Z( 33),IMAX  ),(Z( 34),IMAXA ), SET 660
     9(Z( 35),JMAX  ),(Z( 36),JMAXA ), (Z( 37),KMAX  ),(Z( 38),KMAXA )  SET 670
      EQUIVALENCE                                                       SET 680
     1(Z( 39),BOTM  ),(Z( 40),BOTMV ), (Z( 41),NUMSPT),(Z( 42),CZERO ), SET 690
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ), (Z( 45),PRDELT),(Z( 46),PRFACT)  SET 700
      EQUIVALENCE                                                       SET 710
     1(Z( 47),I1    ),(Z( 48),I2    ), (Z( 49),IPCYCL),(Z( 50),TSTOP ), SET 720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ), (Z( 53),N3    ),(Z( 54),IVARDY), SET 730
     3(Z( 55),VT    ),(Z( 56),N6    ), (Z( 57),RTM   ),(Z( 58),RTMV  ), SET 740
     4(Z( 59),UN59  ),(Z( 60),N10   ), (Z( 61),N11   ),(Z( 62),GAMMA ), SET 750
     5(Z( 63),TOPM  ),(Z( 64),BOTMU ), (Z( 65),SN    ),(Z( 66),TOPMV ), SET 760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP), (Z( 69),PRXRT ),(Z( 70),CYCPH3), SET 770
     7(Z( 71),REZFCT),(Z( 72),TARGI ), (Z( 73),PROJU ),(Z( 74),BBOUND), SET 780
     8(Z( 75),EVAP  ),(Z( 76),ECK   ), (Z( 77),NECYCL),(Z( 78),II    ), SET 790
     9(Z( 79),JJ    ),(Z( 80),NMP   ), (Z( 81),Y2    ),(Z( 82),EZPH1 )  SET 800
      EQUIVALENCE                                                       SET 810
     1(Z( 83),IVARDX),(Z( 84),T     ), (Z( 85),NMPMAX),(Z( 86),PMIN  ), SET 820
     2(Z( 87),INTER ),(Z( 88),TAYBOT), (Z( 89),TAYTOP),(Z( 90),IEMAP ), SET 830
     3(Z( 91),MC    ),(Z( 92),MR    ), (Z( 93),MZ    ),(Z( 94),MB    )  SET 840
      EQUIVALENCE                                                       SET 850
     1(Z( 95),REZ   ),(Z( 96),NODUMP), (Z( 97),UN97  ),(Z( 98),UN98  ), SET 860
     2(Z( 99),UN99  ),(Z(100),EVAPM ), (Z(101),EVAPEN),(Z(102),EVAPMU), SET 870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ), (Z(105),SNL   ),(Z(106),STL   ), SET 880
     4(Z(107),TAXRT ),(Z(108),IDNMAP), (Z(109),IPRMAP),(Z(110),ROEPS ), SET 890
     5(Z(111),RHINI ),(Z(112),VINI  ), (Z(113),FINAL ),(Z(114),IVMAP ), SET 900
     6(Z(115),RHOZ  ),(Z(116),ESA   ), (Z(117),ESEZ  ),(Z(118),ESB   ), SET 910
     7(Z(119),ESCAPA),(Z(120),ESESP ), (Z(121),ESESQ ),(Z(122),ESES  ), SET 920
     8(Z(123),ESALPH),(Z(124),ESBETA), (Z(125),ESCAPB),(Z(126),IUMAP ), SET 930
     9(Z(127),SS1   ),(Z(128),SS2   ), (Z(129),UMIN  ),(Z(130),SS4   )  SET 940
      EQUIVALENCE                                                       SET 950
     1(Z(131),PRTIME),(Z(132),EOR   ), (Z(133),EOT   ),(Z(134),EOB   ), SET 960
     2(Z(135),EMOR  ),(Z(136),DXF   ), (Z(137),DYF   ),(Z(138),RHOMIN), SET 970
     3(Z(139),STAB), (Z(140),XIENRG), (Z(141),XKENRG), (Z(142),XTENRG), SET 980
     4(Z(143),STT   ),(Z(144),DTMIN ), (Z(145),TRNSFC),(Z(146),EMOT  ), SET 990
     5(Z(147),JPROJ ),(Z(148),CNAUT ), (Z(149),BBAR  ),(Z(150),EMOB  )  SET1000
C                                                                       SET1010
C          *** SPECIAL EQUIVALENCES FOR SETUP ONLY                      SET1020
C                                                                       SET1030
      EQUIVALENCE  (RADIUS,PK(12)), (YCENTR,PK(13)), (RHOSPH,Z(100)),   SET1040
     1             (SIESPH,Z(101)), (VINSPH,Z(102)),(RHOOUT,Z(103))     SET1050
C                                                                       SET1060
      COMMON/ SPHERE / ADDVL, AREAFC, ISPHMX, JCENTR, JSPHBT, JSPHTP,   SET1070
     1                 RSQRD, VOLSPH, XL2   , XR2   , YBOTTM, YC2   ,  SET1080
     2                YDIFFB, YDIFFI, YDIFFO, YDIFFT, YLINTA, YLINTB,   SET1090
     3                YLOWER, YRINTA, YRINTB, YTOP  , YUPPER            SET1100
C                                                                       SET1110
C     ..................................................................SET1120
C                                                                       SET1130
C     END OF COMMON                                                     SET1140
C                                                                       SET1150
```

```
C     ***************************************************************SET1160
C                                                                    SET1170
      IF (IVARDY.GT.0) GO TO 30                                      SET1180
C                                                                    SET1190
C            *** IF DY VARIABLE,Y(I) WILL BE READ IN RATHER THAN     SET1200
C                CALCULATED.                                         SET1210
C                                                                    SET1220
      Y(1)=DYF                                                       SET1230
      DO 10 J=2,JMAX                                                 SET1240
      Y(J)=Y(J-1)+DYF                                                SET1250
10    CONTINUE                                                       SET1260
C                                                                    SET1270
C            *** IF DY VARIABLE, DY(I) WILL BE CALCULATED FROM THE Y(I) SET1280
C                READ IN.   IF DY CONSTANT, DY(I) WILL EQUAL DYF FOR SET1290
C                ALL I.                                              SET1300
C                                                                    SET1310
      DO 20 I=1,JMAX                                                 SET1320
20    DY(I)=DYF                                                      SET1330
      GO TO 50                                                       SET1340
C                                                                    SET1350
C              *** CALCULATE VARIABLE DY(I).                         SET1360
30    DO 40 I=1,JMAX                                                 SET1370
40    DY(I)=Y(I)-Y(I-1)                                              SET1380
50    CONTINUE                                                       SET1390
C                                                                    SET1400
C           ***  IF DX VARIABLE, X(I) WILL BE READ IN RATHER THAN    SET1410
C                CALCULATED                                          SET1420
C                                                                    SET1430
      IF (IVARDX.GT.0) GO TO 80                                      SET1440
      X(1)=DXF                                                       SET1450
      DO 60 I=2,IMAX                                                 SET1460
      X(I)=X(I-1)+DXF                                                SET1470
60    CONTINUE                                                       SET1480
C                                                                    SET1490
C            *** IF DX VARIABLE, DX(I)WILL BE CALCULATED FROM        SET1500
C                THE X(I) READ IN. IF DX CONSTANT, DX(I) WILL        SET1520
C                EQUAL DXF FOR ALL I.                                SET1525
C                                                                    SET1530
      DO 70 I=1,IMAX                                                 SET1540
      DX(I)=DXF                                                      SET1550
70    CONTINUE                                                       SET1560
      GO TO 100                                                      SET1570
C                                                                    SET1580
C              ***  CALCULATE VARIABLE DX(I)                         SET1590
C                                                                    SET1600
80    DO 90 I=1,IMAX                                                 SET1610
90    DX(I)=X(I)-X(I-1)                                              SET1620
C            *** MAKE SURE DX AND DY ARRAYS HAVE BEEN DEFINED.       SET1625
95    IF (DX(1).GT.0..AND.DY(1).GT.0.) GO TO 100                     SET1630
      GO TO 770                                                      SET1640
100   CONTINUE                                                       SET1650
C                                                                    SET1660
C            *** PK(3) = -3. WHEN RESTARTING FROM A CLAM TAPE.       SET1662
C                PROPERTIES OF CELLS HAVE ALREADY BEEN DEFINED       SET1664
C                BUT TRACER POINTS HAVE NOT.                         SET1666
      IF (PK(3).EQ.(-3.)) GO TO 700                                  SET1670
C            *** PRYBOT=-1. MEANS THE PROJECTILE PACKAGE             SET1672
```

```
C                 IS NOT BEING USED.                                    SET1674
      IF (PRYBOT.LT.0.) GO TO 200                                       SET1680
C           *** DEFINE CELL BOUNDARIES (MB,MC,MR) OF PROJECTILE         SET1690
C                 PACKAGE.                                              SET1695
      IF (IVARDY.GT.0) GO TO 110                                        SET1700
C           *** CALCULATION FOR CONSTANT DY.                            SET1705
      MB=INT(PRYBOT/DYF+.5)+1                                           SET1710
      IF (MB.GT.JMAX) GO TO 200                                         SET1720
      M=1                                                               SET1730
      MC=INT(PRYTOP/DYF+.5)                                             SET1740
      IF (MC.GT.JMAX) MC=JMAX                                           SET1750
      GO TO 160                                                         SET1760
C           *** CALCULATION FOR VARIABLE DY.                            SET1765
110   DYSUM=0.                                                          SET1770
      I=0                                                               SET1780
C           *** SEARCH FOR J-VALUE OF BOTTOM OF PROJECTILE (MB).        SET1785
      IF (PRYBOT.EQ.0.) GO TO 130                                       SET1790
      DO 120 I=1,JMAX                                                   SET1800
      DYSUM=DYSUM+DY(I)                                                 SET1810
      IF (PRYBOT.LT.DYSUM+.5*DY(I+1).AND.PRYBOT.GE.DYSUM-.5*DY(I)) GO TOSET1820
     1 130                                                              SET1830
120   CONTINUE                                                          SET1840
      GO TO 200                                                         SET1850
130   MB=MIN0(I+1,JMAX)                                                 SET1860
      M=1                                                               SET1870
C           *** SEARCH FOR J-VALUE OF TOP OF PROJECTILE (MC).           SET1875
      DO 140 I=MB,JMAX                                                  SET1880
      DYSUM=DYSUM+DY(I)                                                 SET1890
      IF (PRYTOP.GE.DYSUM-.5*DY(I).AND.PRYTOP.LT.DYSUM+.5*DY(I+1)) GO TOSET1900
     1 150                                                              SET1910
140   CONTINUE                                                          SET1920
      MC=JMAX                                                           SET1930
      GO TO 160                                                         SET1940
150   MC=I                                                              SET1950
C           *** CALCULATION OF I-VALUE OF RIGHT SIDE OF PROJECTILE (MR)SET1955
160   IF (IVARDX.GT.0) GO TO 170                                        SET1960
C           *** CALCULATION FOR CONSTANT DX.                            SET1965
      MR=INT(PRXRT/DXF+.5)                                              SET1970
      IF (MR.GT.IMAX) MR=IMAX                                           SET1980
      GO TO 210                                                         SET1990
C           *** CALCULATION FOR VARIABLE DX.                            SET1995
170   DXSUM=0.                                                          SET2000
      DO 180 I=1,IMAX                                                   SET2010
      DXSUM=DXSUM+DX(I)                                                 SET2020
      IF (PRXRT.GE.DXSUM-.5*DX(I).AND.PRXRT.LT.DXSUM+.5*DX(I+1)) GO TO 1SET2030
     190                                                                SET2040
180   CONTINUE                                                          SET2050
      MR=IMAX                                                           SET2060
      GO TO 210                                                         SET2070
190   MR=I                                                              SET2080
      GO TO 210                                                         SET2090
C           *** M=0 MEANS THE PROJECTILE PACKAGE IS NOT BEING USED.     SET2100
200   M=0                                                               SET2110
C           *** TAYBOT=-1. MEANS THE TARGET PACKAGE IS NOT BEING USED.  SET2115
210   IF (TAYBOT.LT.0.) GO TO 310                                       SET2120
C           *** DEFINE CELL BOUNDARIES(MZ,N,ME) OF TARGET PACKAGE.      SET2130
      IF (IVARDY.GT.0) GO TO 220                                        SET2140
```

```
C             *** CALCULATION FOR CONSTANT DY.                          SET2145
      MZ=INT(TAYBOT/DYF+.5)+1                                           SET2150
      IF (MZ.GT.JMAX) GO TO 310                                         SET2160
      MD=1                                                              SET2170
      N=INT(TAYTOP/DYF+.5)                                              SET2180
      IF (N.GT.JMAX) N=JMAX                                             SET2190
      GO TO 270                                                         SET2200
C             *** CALCULATION FOR VARIABLE DY.                          SET2205
220   DYSUM=0.                                                          SET2210
      I=0                                                               SET2220
C             *** SEARCH FOR J-VALUE OF BOTTOM OF TARGET (MZ).          SET2225
      IF (TAYBOT.EQ.0.) GO TO 240                                       SET2230
      DO 230 I=1,JMAX                                                   SET2240
      DYSUM=DYSUM+DY(I)                                                 SET2250
      IF (TAYBOT.GE.DYSUM-.5*DY(I).AND.TAYBOT.LT.DYSUM+.5*DY(I+1)) GO TOSET2260
     1 240                                                              SET2270
230   CONTINUE                                                          SET2280
      GO TO 310                                                         SET2290
240   MZ=MIN0(I+1,JMAX)                                                 SET2300
      MD=1                                                              SET2310
C             *** SEARCH FOR J-VALUE OF TOP OF TARGET (N).              SET2315
      DO 250 I=MZ,JMAX                                                  SET2320
      DYSUM=DYSUM+DY(I)                                                 SET2330
      IF (TAYTOP.GE.DYSUM-.5*DY(I).AND.TAYTOP.LT.DYSUM+.5*DY(I+1)) GO TOSET2340
     1 260                                                              SET2350
250   CONTINUE                                                          SET2360
      N=JMAX                                                            SET2370
      GO TO 270                                                         SET2380
260   N=I                                                               SET2390
C             *** CALCULATION OF I-VALUE OF RIGHT SIDE OF TARGET(ME).   SET2395
270   IF (IVARDX.GT.0) GO TO 280                                        SET2400
C             *** CALCULATION FOR CONSTANT DX.                          SET2405
      ME=INT(TAXRT/DXF+.5)                                              SET2410
      IF (ME.GT.IMAX) ME=IMAX                                           SET2420
      GO TO 320                                                         SET2430
C             *** CALCULATION FOR VARIABLE DX.                          SET2435
280   DXSUM=0.                                                          SET2440
      DO 290 I=1,IMAX                                                   SET2450
      DXSUM=DXSUM+DX(I)                                                 SET2460
      IF (TAXRT.GE.DXSUM-.5*DX(I).AND.TAXRT.LT.DXSUM+.5*DX(I+1)) GO TO 3SET2470
     100                                                                SET2480
290   CONTINUE                                                          SET2490
      ME=IMAX                                                           SET2500
      GO TO 320                                                         SET2510
300   ME=I                                                              SET2520
      GO TO 320                                                         SET2530
C             *** MD = 0 MEANS THE TARGET PACKAGE IS NOT BEING USED.    SET2540
310   MD=0                                                              SET2550
320   KMAX=IMAX*JMAX+1                                                  SET2560
      KMAXA=KMAX+1                                                      SET2570
      JMAXA=JMAX+1                                                      SET2580
      IMAXA=IMAX+1                                                      SET2590
C             *** INITIALIZE PROPERTY ARRAYS.                           SET2595
      DO 330 K=1,KMAX                                                   SET2600
      U(K)=0.0                                                          SET2610
      V(K)=0.0                                                          SET2620
      P(K)=0.0                                                          SET2630
```

```
      AMX(K)=0.0                                                        SET2640
      AIX(K)=0.0                                                        SET2650
330   CONTINUE                                                          SET2660
      PIDY=3.1415927                                                    SET2670
      WS=X(1)**2                                                        SET2680
C            *** CALCULATE CELL-FACE AREA, THE AREA GENERATED BY SEGMENTSET2682
C                X(I),X(I+1) ROTATED ABOUT THE Z-AXIS.                  SET2684
      TAU(1)=PIDY*WS                                                    SET2690
      DO 340 I=2,IMAX                                                   SET2700
      WSA=X(I)**2                                                       SET2710
      TAU(I)=PIDY*(WSA-WS)                                              SET2720
      WS=WSA                                                            SET2730
340   CONTINUE                                                          SET2740
      ETH=0.0                                                           SET2750
C            *** RADIUS.GT.0. MEANS SPHERE IS TO BE USED.               SET2760
C                SEE SPECIAL EQUIVALENCES FOR SETUP FOR LOCATION        SET2770
C                OF PARAMETERS DEFINING DIMENSIONS AND PROPERTIES OF    SET2780
C                SPHERE.                                                SET2790
      IF (RADIUS.LE.0.) GO TO 540                                       SET2850
C            *** COMPUTE ISPHMX, THE I-INDEX OF THE RIGHT-MOST COLUMN   SET2860
C                CONTAINING A PART OF THE SPHERE.                       SET2870
      DO 350 I=1,IMAX                                                   SET2900
      IF (X(I).GE.RADIUS-.000001*DX(I)) GO TO 360                       SET2910
350   CONTINUE                                                          SET2920
360   ISPHMX=I                                                          SET2930
      TOTSPH=0.                                                         SET2940
C            *** COMPUTE JCENTR=J-INDEX OF SPHERE-CENTER                SET2950
      DO 370 J=0,JMAX                                                   SET2960
      IF ((Y(J)+.5*DY(J+1)).GT.YCENTR) GO TO 380                        SET2970
370   CONTINUE                                                          SET2980
C                *** YCENTR SHOULD FALL ON CELL BOUNDARY,               SET2990
C                PRINT OUT INPUT VALUE AND ADJUSTED VALUE.              SET3000
380   WRITE (6,790) YCENTR,Y(J)                                         SET3010
      YCENTR=Y(J)                                                       SET3020
      JCENTR=J                                                          SET3030
C            COMPUTE JRADA AND JRADB.                                   SET3035
C           *** JRADB = THE NUMBER OF CELLS CONTAINING A PART OF THE    SET3040
C                       SPHERE FROM THE CENTER TO BOTTOM EDGE           SET3050
C           *** JRADA = THE NUMBER OF CELLS CONTAINING A PART OF THE    SET3060
C                       SPHERE FROM THE CENTER TO TOP EDGE              SET3070
      JRADB=0                                                           SET3080
      JRADA=0                                                           SET3090
      JB=JCENTR                                                         SET3100
      JA=JCENTR+1                                                       SET3110
      SUM1=0.                                                           SET3120
      SUM2=0.                                                           SET3130
      IF (JCENTR.EQ.0) GO TO 400                                        SET3140
390   SUM1=SUM1+DY(JB)                                                  SET3150
      JB=JB-1                                                           SET3160
      JRADB=JRADB+1                                                     SET3170
      IF (SUM1.LT.(RADIUS-.000001*DY(JB))) GO TO 390                    SET3180
400   SUM2=SUM2+DY(JA)                                                  SET3190
      JA=JA+1                                                           SET3200
      JRADA=JRADA+1                                                     SET3210
      IF (SUM2.LT.(RADIUS-.000001*DY(JA))) GO TO 400                    SET3220
C            *** COMPUTE(1)JSPHTP=J-INDEX OF UPPER-MOST ROW             SET3230
C            ***              WHICH CONTAINS PART OF THE SPHERE         SET3240
```

```
C             ***          (2)JSPHBT=J-INDEX OF LOWEST ROW                  SET3250
C             ***               WHICH CONTAINS PART OF THE SPHERE           SET3260
      JSPHBT=MAX0(1,JCENTR-JRADB+1)                                          SET3270
      JSPHTP=MIN0(JMAX,JCENTR+JRADA)                                         SET3280
C                                                                            SET3290
      YC2=YCENTR**2                                                          SET3300
      RSQRD=RADIUS**2                                                        SET3310
C             *** FOR EACH CELL IN RECTANGLE FROM X=0.                       SET3320
C             ***    TO X=(ISPHMX-1)*DXF AND FROM                            SET3330
C             ***    Y=(JSPHBT-1)*DYF TO Y=(JSPHTP)*DYF                      SET3340
C             ***    FIND VOLSPH=VOLUME OF SPHERE IN CELL K                  SET3350
C             ***    AND SET MASS AND SPEC. INT. ENERGY.                     SET3360
      DO 530 I=1,ISPHMX                                                      SET3370
      K=(JSPHBT-1)*IMAX+I+1                                                  SET3380
C             *** X(I-1)=VALUE OF X AT LEFT OF COLUMN                        SET3390
C             *** X(I)=VALUE OF X AT RIGHT OF COLUMN                         SET3400
      XL2=(X(I-1))**2                                                        SET3410
      XR2=(X(I))**2                                                          SET3420
C             *** YLINTA=Y-LEFT-INTERCEPT-ABOVE-CENTER                       SET3430
C             *** YLINTB=Y-LEFT-INTERCEPT-BELOW-CENTER                       SET3440
      WS=SQRT(RSQRD-XL2)                                                     SET3450
      YLINTA=YCENTR+WS                                                       SET3460
      YLINTB=YCENTR-WS                                                       SET3470
C             *** DOES CURVE INTERSECT X=X(I)                                SET3480
      IF (RSQRD.LE.XR2) GO TO 410                                            SET3490
C             *** YES                                                        SET3500
      WS=SQRT(RSQRD-XR2)                                                     SET3510
      YRINTA=YCENTR+WS                                                       SET3520
      YRINTB=YCENTR-WS                                                       SET3530
      GO TO 420                                                              SET3540
410   YRINTA=YCENTR                                                          SET3550
      YRINTB=YCENTR                                                          SET3560
420   CONTINUE                                                               SET3570
      DO 520 J=JSPHBT,JSPHTP                                                 SET3610
C             *** SKIP IF SPECIAL CELL                                       SET3620
      IF (AMX(K).NE.0.) GO TO 520                                            SET3630
      YTOP=Y(J)                                                              SET3640
      YBOTTM=Y(J-1)                                                          SET3650
      YDIFFT=(YTOP-YCENTR)**2                                                SET3660
      YDIFFB=(YBOTTM-YCENTR)**2                                              SET3670
      YDIFFO=AMAX1(YDIFFT,YDIFFB)                                            SET3680
      YDIFFI=AMIN1(YDIFFT,YDIFFB)                                            SET3690
C             *** IS ALL OF CELL WITHIN SPHERE BOUNDARY.                     SET3695
      IF ((YDIFFO+XR2).GT.RSQRD) GO TO 430                                   SET3700
C             *** YES. DEFINE VOLUME OF CELL.                                SET3705
      VOLSPH=TAU(I)*DY(J)                                                    SET3710
      GO TO 470                                                              SET3720
C             *** NO. IS ALL OF CELL OUTSIDE SPHERE BOUNDARY.                SET3725
430   IF ((YDIFFI+XL2).LT.RSQRD) GO TO 440                                   SET3730
C             *** YES.                                                       SET3735
      VOLSPH=0.                                                              SET3740
      GO TO 510                                                              SET3750
C             *** NO. PART OF CELL IS WITHIN SPHERE. COMPUTE VOLUME          SET3752
C                     OF PART OF CELL INSIDE THE SPHERE AND STORE            SET3754
C                     IN VOLSPH.                                             SET3756
440   IF (J.GT.JCENTR) GO TO 450                                             SET3760
      YLOWER=AMAX1(YBOTTM,YLINTB)                                            SET3770
```

```
      YUPPER=AMIN1(YTOP,YRINTB)                                         SET3790
      ADDVL=(YTOP-YUPPER)*TAU(I)                                        SET3790
      GO TO 460                                                         SET3800
 450  YLOWER=AMAX1(YBOTTM,YRINTA)                                       SET3810
      YUPPER=AMIN1(YTOP,YLINTA)                                         SET3820
      ADDVL=(YLOWER-YBOTTM)*TAU(I)                                      SET3830
 460  VOLSPH=ADDVL+PIDY*((RSQRD-YC2-XL2)*(YUPPER-YLOWER)-(YUPPER**3-YLOWSET3840
     1ER**3)/3.+YCENTR*(YUPPER**2-YLOWER**2))                           SET3850
 470  WS=VOLSPH*RHOSPH                                                  SET3860
      AMX(K)=WS                                                         SET3870
C            *** CHECK WHETHER THE CELL IS FULL                         SET3880
      WSA=TAU(I)*DY(J)                                                  SET3890
      WSB=WSA-VOLSPH                                                    SET3900
      IF (ABS(WSB/WSA).LT.ROEPS) GO TO 490                              SET3910
C            *** ADD RHOOUT MATERIAL TO CELL                            SET3920
      WSB=WSB*RHOOUT                                                    SET3930
      AMX(K)=WS+WSB                                                     SET3940
C                *** CHECK WHETHER MASS IS TOO SMALL TO KEEP           SET3945
      IF(AMX(K)/WSA.LT.EVAP*RHINI) GO TO 510                            SET3950
C            *** USE A WEIGHTED AVERAGE OF THE PROPERTIES OF THE SPHERE SET3960
C                AND THE PROJECTILE FOR CELLS PARTIALLY IN THE SPHERE. SET3965
      AIX(K)=(WS*SIESPH+WSB*PROJI)/AMX(K)                               SET3970
      V(K)=(WS*VINSPH+WSB*VINI)/AMX(K)                                  SET3980
      GO TO 500                                                         SET3990
C            *** ESSENTIALLY ALL OF CELL IS IN SPHERE                   SET4030
 490  AIX(K)=SIESPH                                                     SET4040
      V(K)=VINSPH                                                       SET4050
C                *** SUM SPHERE VOLUME                                 SET4060
 500  TOTSPH=TOTSPH+VOLSPH                                              SET4070
      GO TO 520                                                         SET4080
 510  AMX(K)=0.                                                         SET4090
C                *** END OF J-LOOP                                     SET4100
 520  K=K+IMAX                                                          SET4110
C                *** END OF I-LOOP                                     SET4120
 530  CONTINUE                                                          SET4130
 540  WRITE (6,800) RHOSPH,RHINI,RHINIT,RHOFIL,SIESPH,PROJI,TARGI,VINSPHSET4140
     1,VINI,TARGV,PROJU,RADIUS,PRYTOP,TAYTOP,YCENTR,PRYBOT,TAYBOT,PRXRT,SET4150
     2TAXRT                                                             SET4160
C            *** RESET BORROWED Z-STORAGE TO ZERO.                      SET4162
      EVAPM = 0.                                                        SET4164
      EVAPEN = 0.                                                       SET4166
      EVAPMU = 0.                                                       SET4168
      EVAPMV = 0.                                                       SET4169
C            *** M=0 MEANS THE PROJECTILE PACKAGE IS NOT BEING USED.    SET4170
      IF (M.EQ.0) GO TO 610                                             SET4220
      DO 600 I=M,MR                                                     SET4230
      K=(MB-1)*IMAX+I+1                                                 SET4240
C            *** ASSIGN PROPERTIES TO CELLS IN PROJECTILE.              SET4250
      DO 590 J=MB,MC                                                    SET4260
      IF (AMX(K).NE.0.) GO TO 550                                       SET4270
      AMX(K)=RHINI*DY(J)*TAU(I)                                         SET4280
      IF (V(K).NE.0.) GO TO 560                                         SET4290
      V(K)=VINI                                                         SET4300
 560  IF (U(K).NE.0.) GO TO 570                                         SET4310
      U(K)=PROJU                                                        SET4320
 570  IF (AIX(K).NE.0.) GO TO 580                                       SET4330
      AIX(K)=PROJI                                                      SET4340
```

```
580   CONTINUE                                                          SET4350
590   K=K+IMAX                                                          SET4360
600   CONTINUE                                                          SET4370
C           *** MD=0 MEANS THE TARGET PACKAGE IS NOT BEING USED.        SET4380
610   IF (MD.EQ.0) GO TO 650                                            SET4390
C           *** ASSIGN PROPERTIES TO CELLS IN TARGET.                   SET4395
      DO 640 I=MD,ME                                                    SET4400
      K=(MZ-1)*IMAX+I+1                                                 SET4410
      DO 630 J=MZ,N                                                     SET4420
      IF (V(K).NE.0.) GO TO 620                                         SET4430
      V(K)=TARGV                                                        SET4440
      IF (AMX(K).NE.0.) GO TO 630                                       SET4450
      AMX(K)=RHINIT*DY(J)*TAU(I)                                        SET4460
620   AIX(K)=TARGI                                                      SET4470
630   K=K+IMAX                                                          SET4480
640   CONTINUE                                                          SET4490
650   CONTINUE                                                          SET4500
C                                                                       SET4510
      CYCLE=0.0                                                         SET4520
      DT=0.0                                                            SET4530
      NREZ=NUMREZ                                                       SET4540
      NZ=1                                                              SET4550
C           *** RHOFIL=0. WHEN THERE IS NO FILLER MATERIAL BETWEEN      SET4560
C               PROJECTILE AND TARGET.                                  SET4565
      IF (RHOFIL.EQ.0.) GO TO 680                                       SET4570
C           *** FILL BETWEEN PACKAGES WITH MATERIAL OF DENSITY=RHOFIL.  SET4580
      MC=MC+1                                                           SET4590
      MZ=MZ-1                                                           SET4600
      DO 670 I=1,IMAX                                                   SET4610
      K=(MC-1)*IMAX+I+1                                                 SET4620
      DO 660 J=MC,MZ                                                    SET4630
      AMX(K)=RHOFIL*DY(J)*TAU(I)                                        SET4640
660   K=K+IMAX                                                          SET4650
670   CONTINUE                                                          SET4660
680   N3=0                                                              SET4670
C           *** PK(14).GT.0. MEANS SOME CELLS WILL BE DEFINED           SET4672
C               AFTER PACKAGES ARE SET UP.                              SET4674
      IF (PK(14).GT.0.) CALL CARDS                                      SET4680
C           *** CALCULATE INITIAL VALUE OF TOTAL ENERGY TO BE ADJUSTED  SET4682
C               WHEN MATERIAL IS EVAPORATED OR CROSSES A TRANSMITTIVE   SET4684
C               BOUNDARY AND TO BE USED IN EDIT TO CHECK ERROR IN       SET4686
C               ENERGY SUM.                                             SET4688
      DO 690 K=2,KMAX                                                   SET4690
      ETH=ETH+AMX(K)*(.5*(U(K)**2+V(K)**2)+AIX(K))                      SET4700
690   CONTINUE                                                          SET4710
      XMAX=X(IMAX)                                                      SET4720
      TXMAX=XMAX*2.0                                                    SET4730
      YMAX=Y(JMAX)                                                      SET4740
      TYMAX=YMAX*2.0                                                    SET4750
700   IF (Y2.GT.(-1.)) GO TO 750                                        SET4760
C           *** PUT TRACER POINT IN CENTER OF EVERY OTHER NONEMPTY CELLSET4770
C               IN EVERY OTHER ROW. THE TRACER POINT COORDINATES OF     SET4772
C               EMPTY CELLS ARE (0,0).                                  SET4774
      II=IMAX/2                                                         SET4780
      JJ=JMAX/2                                                         SET4790
      DO 720 J=1,JJ                                                     SET4800
      DO 720 I=1,II                                                     SET4810
```

```
      K=2*((J-1)*IMAX+I)                                                SET4820
      IF (AMX(K).EQ.0.) GO TO 710                                       SET4830
      XP(I,J)=FLOAT(2*I-1)-.5                                           SET4840
      YP(I,J)=FLOAT(2*J-1)-.5                                           SET4850
 710  NMP=NMP+1                                                         SET4860
C            *** NMPMAX IS THE MAXIMUM NUMBER OF TRACER POINTS TO       SET4862
C                BE USED AND IS DEFINED IN THE INPUT DECK.              SET4864
      IF (NMP.GE.NMPMAX) GO TO 730                                      SET4870
 720  CONTINUE                                                          SET4880
      GO TO 740                                                         SET4890
 730  JJ=J                                                              SET4900
C            *** N3=1 MEANS TRACER POINTS ARE BEING USED, ADDING        SET4902
C                ONE MORE RECORD TO EACH TAPE DUMP.                     SET4904
 740  N3=1                                                              SET4910
 750  REWIND 7                                                          SET4920
      WS=555.0                                                          SET4930
C     WRITE OUTPUT FOR OIL ON TAPE.                                     SET4940
      WRITE (7) WS,CYCLE,N3                                             SET4950
      WRITE (7) (Z(I),I=1,MZT)                                          SET4960
      WRITE (7) (U(I),V(I),AMX(I),AIX(I),P(I),I=1,KMAXA)                SET4970
      WRITE (7) X(0),(X(I),TAU(I),JPM(I),I=1,IMAX)                      SET4980
      WRITE (7) (Y(I),I=0,JMAX)                                         SET4990
      IF (Y2.GT.(-1.)) GO TO 760                                        SET5000
      WRITE (7) ((XP(I,J),YP(I,J),I=1,II),J=1,JJ)                       SET5010
 760  WRITE (7) (DX(I),I=1,IMAX)                                        SET5020
      WRITE (7) (DY(J),J=1,JMAX)                                        SET5030
      WS=666.0                                                          SET5040
      WRITE (7) WS,WS,WS                                                SET5050
      GO TO 780                                                         SET5060
C            *** DX AND/OR DY ARRAY NOT PROPERLY DEFINED.               SET5062
C                CHECK VALUE OF DXF AND DYF IF ZONES ARE CONSTANT.      SET5064
C                IF VARIABLE, CHECK LOCATION NUMBERS USED FOR           SET5065
C                READING IN  X AND/OR Y ARRAY ESPECIALLY IF ANY         SET5066
C                VARIABLE DIMENSIONS WERE CHANGED.                      SET5068
 770  NK=95                                                             SET5070
      NR=2                                                              SET5080
      CALL ERROR                                                        SET5090
 780  RETURN                                                            SET5100
C                                                                       SET5110
 790  FORMAT (/5X,15HINPUT YCENTR = ,1PE12.6,6X,18HADJUSTED YCENTR = ,1PSET5120
     1E12.6)                                                            SET5130
 800  FORMAT (////17X,18HINITIAL CONDITIONS//11X,6HSPHERE,13X,9HPACKAGE SET5140
     11,21H   PACKAGE 2   FILLER//8H DENSITY,1P1E12.4,6X,1P3E12.4/8H  S.SET5150
     2I.E.,1P1E12.4,6X,1P2E12.4/8H       V,1P1E12.4,6X,1P2E12.4/8H     SET5160
     3 U,13X,1P1E12.4/8H  RADIUS,1P1E12.4,5X,3HTOP,1P1E10.4,1P1E12.4/8H SET5170
     4YCENTER,1P1E12.4,2X,6HBOTTOM,1P1E10.4,1P1E12.4/23X,5HRIGHT,1P1E10.SET5180
     54,1P1E12.4)                                                       SET5190
      END                                                               SET5200-
```

```
      SUBROUTINE CDT
C     ..................................................................
C
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),
     3          GAMC(102),
     4          PK(15),   Z(150)   ,
     5          XP(26,51),YP(26,51),
     6          PL(204)  ,UL(204)  ,PR(204)  ,
     7           RSN(52),   RST(52),
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)
C            *** DIMENSIONED ARRAYS
C            *** Z-BLOCK IS SAVED ON TAPE.
      COMMON     Z
      COMMON    PK
      COMMON  YY,      XX
      COMMON  DDX,     DDY
      COMMON  AMX,     AIX,     U,     V,     P
      COMMON  TAU,     JPM
      COMMON  UL ,     PL
      COMMON  XP ,     YP,      CMXP, CMYP
C            *** NON-DIMENSIONED VARIABLES
      COMMON         AID    ,AMMV   ,AMMY   ,AMPY   ,AMUR   ,AMUT   ,AMVR   ,
     1AMVT   ,DELEB ,DELER ,DELET ,DELM   ,DTODX ,DXYMIN,EAMMP ,EAMPY ,
     2E      ,ERDUMP,I     ,I3    ,IWS    ,J      ,K      ,KA     ,KB     ,
     3LL     ,MD    ,ME    ,MZT   ,NERR   ,NK     ,NPRINT,
     4NR     ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR    ,SNT    ,STR    ,SOLID ,
     5SUM    ,TESTRH,TWOPI ,URR   ,WS     ,WSA    ,WSB    ,WSC    ,WFLAGF,
     6WFLAGL,WFLAGP
C
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE
C                X(0), Y(0), DX(0), DY(0)
C
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))
C
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY
C
      EQUIVALENCE          (UL,FLEFT),          (UL(103),YAMC),
     1                     (PL,GAMC,PR),        (PL(103),SIGC)
C
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY
C
      EQUIVALENCE          (UL,RSN),
     1                     (PL,RST),            (P,UK),
     2                     (P(157),VK),         (P(313),SNB),
     3                     (P(365),STB),        (P(417),RHO)
C
C            *** SPECIAL EQUIVALENCES FOR EDIT
C
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)
C
C            *** Z-STORAGE EQUIVALENCES
C
      EQUIVALENCE                      (Z(  1),PROB  ),(Z(  2),CYCLE ),
```

```
     1(Z(  3),DT     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  CDT 590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ), (Z(  9),TOPMU ),(Z( 10),RTMU  ), CDT 600
     3(Z( 11),STK1  ),(Z( 12),NUMREZ), (Z( 13),ETH   ),(Z( 14),UN14  ), CDT 610
     4(Z( 15),RHINIT),(Z( 16),PROJI ), (Z( 17),UN17  ),(Z( 18),XMAX  ), CDT 620
     5(Z( 19),NZ    ),(Z( 20),NREZ  ), (Z( 21),AMDM  ),(Z( 22),UVMAX ), CDT 630
     6(Z( 23),UN23  ),(Z( 24),DMIN  ), (Z( 25),JSTR  ),(Z( 26),DTNA  ), CDT 640
     7(Z( 27),CVIS  ),(Z( 28),STK2  ), (Z( 29),STEZ  ),(Z( 30),NC    ), CDT 650
     8(Z( 31),UN31  ),(Z( 32),NRC   ), (Z( 33),IMAX  ),(Z( 34),IMAXA ), CDT 660
     9(Z( 35),JMAX  ),(Z( 36),JMAXA ), (Z( 37),KMAX  ),(Z( 38),KMAXA ) CDT 670
      EQUIVALENCE                                                      CDT 680
     1(Z( 39),BOTM  ),(Z( 40),BOTMV ), (Z( 41),NUMSPT),(Z( 42),CZERO ), CDT 690
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ), (Z( 45),PRDELT),(Z( 46),PRFACT) CDT 700
      EQUIVALENCE                                                      CDT 710
     1(Z( 47),I1    ),(Z( 48),I2    ), (Z( 49),IPCYCL),(Z( 50),TSTOP ), CDT 720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ), (Z( 53),N3    ),(Z( 54),IVARDY), CDT 730
     3(Z( 55),VT    ),(Z( 56),N6    ), (Z( 57),RTM   ),(Z( 58),RTMV  ), CDT 740
     4(Z( 59),UN59  ),(Z( 60),N10   ), (Z( 61),N11   ),(Z( 62),GAMMA ), CDT 750
     5(Z( 63),TOPM  ),(Z( 64),BOTMU ), (Z( 65),SN    ),(Z( 66),TOPMV ), CDT 760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP), (Z( 69),PRXRT ),(Z( 70),CYCPH3), CDT 770
     7(Z( 71),REZFCT),(Z( 72),TARGI ), (Z( 73),PROJU ),(Z( 74),BBOUND), CDT 780
     8(Z( 75),EVAP  ),(Z( 76),ECK   ), (Z( 77),NECYCL),(Z( 78),II    ), CDT 790
     9(Z( 79),JJ    ),(Z( 80),NMP   ), (Z( 81),Y2    ),(Z( 82),EZPH1 ) CDT 800
      EQUIVALENCE                                                      CDT 810
     1(Z( 83),IVARDX),(Z( 84),T     ), (Z( 85),NMPMAX),(Z( 86),PMIN  ), CDT 820
     2(Z( 87),INTER ),(Z( 88),TAYBOT), (Z( 89),TAYTOP),(Z( 90),IEMAP ), CDT 830
     3(Z( 91),MC    ),(Z( 92),MR    ), (Z( 93),MZ    ),(Z( 94),MB    ) CDT 840
      EQUIVALENCE                                                      CDT 850
     1(Z( 95),REZ   ),(Z( 96),NODUMP), (Z( 97),UN97  ),(Z( 98),UN98  ), CDT 860
     2(Z( 99),UN99  ),(Z(100),EVAPM ), (Z(101),EVAPEN),(Z(102),EVAPMU), CDT 870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ), (Z(105),SNL   ),(Z(106),STL   ), CDT 880
     4(Z(107),TAXRT ),(Z(108),IDNMAP), (Z(109),IPRMAP),(Z(110),ROEPS ), CDT 890
     5(Z(111),RHINI ),(Z(112),VINI  ), (Z(113),FINAL ),(Z(114),IVMAP ), CDT 900
     6(Z(115),RHOZ  ),(Z(116),ESA   ), (Z(117),ESEZ  ),(Z(118),ESB   ), CDT 910
     7(Z(119),ESCAPA),(Z(120),ESESP ), (Z(121),ESESQ ),(Z(122),ESES  ), CDT 920
     8(Z(123),ESALPH),(Z(124),ESBETA), (Z(125),ESCAPB),(Z(126),IUMAP ), CDT 930
     9(Z(127),SS1   ),(Z(128),SS2   ), (Z(129),UMIN  ),(Z(130),SS4   ) CDT 940
      EQUIVALENCE                                                      CDT 950
     1(Z(131),PRTIME),(Z(132),EOR   ), (Z(133),EOT   ),(Z(134),EOB   ), CDT 960
     2(Z(135),EMOR  ),(Z(136),DXF   ), (Z(137),DYF   ),(Z(138),RHOMIN), CDT 970
     3(Z(139),STAB), (Z(140),XIENRG),  (Z(141),XKENRG), (Z(142),XTENRG),CDT 980
     4(Z(143),STT   ),(Z(144),DTMIN ), (Z(145),TRNSFC),(Z(146),EMOT  ), CDT 990
     5(Z(147),JPROJ ),(Z(148),CNAUT ), (Z(149),BBAR  ),(Z(150),EMOB  ) CDT1000
C                                                                      CDT1010
C     ..................................................................CDT1020
C                                                                      CDT1030
C     END OF COMMON                                                    CDT1040
C                                                                      CDT1050
C     ..................................................................CDT1060
C                                                                      CDT1070
C            *** SPECIAL EQUIV FOR ES AND CDT                          CDT1080
      EQUIVALENCE (RHOW,NULLE)                                         CDT1090
C                                                                      CDT1100
C                  ***CHECK COURANT CONDITION AND PARTICLE VELOCITY.   CDT1110
C                  ***RECORD I AND J OF ZONE WHERE DT IS CONTROLLED.   CDT1120
C                  ***FIRST CALCULATE PRESSURES FROM EQ. OF ST.        CDT1130
C                                                                      CDT1140
C                                                                      CDT1220
```

```
10    TRIAL=0.                                                          CDT1230
      SRATIO=10.**10                                                    CDT1240
C                **WSC WILL BE MAXIMUM U OR V                           CDT1250
      WSC=0.                                                            CDT1260
      DO 60 I=1,I1                                                      CDT1270
      K=I+1                                                             CDT1280
      DO 60 J=1,I2                                                      CDT1290
      RHOW=AMX(K)/(TAU(I)*DY(J))                                        CDT1330
      CALL ES                                                           CDT1340
C            *** IF DENSITY OF CELL K IS LESS THAN RHOMIN, IT'S         CDT1342
C                VELOCITY OR SOUND SPEED IS NOT USED IN DETERMINING DT. CDT1344
      IF (RHOW.LT.RHOMIN) GO TO 50                                      CDT1350
      IF (ABS(P(K ).LT.PMIN) P(K)=0.                                    CDT1360
      IF (CNAUT.GT.0.) GO TO 20                                         CDT1370
C                                                                       CDT1380
C                   ***CALCULATE SOUND SPEED FOR POLYTOPIC GAS WITH     CDT1390
C                   ***GAMMA EQUAL TO ESA+1.                            CDT1400
      WS=SQRT(GAMMA*ABS(P(K))/RHOW)                                     CDT1410
      GO TO 40                                                          CDT1420
C                                                                       CDT1430
C                  ***CHECK FOR NEGATIVE PRESSURE.                      CDT1440
20    IF (P(K).GT.0.) GO TO 30                                          CDT1450
C            *** NEGATIVE PRESSURES NOT ALLOWED ALONG GRID BOUNDARY     CDT1452
C                AND NOT ALLOWED ANYWHERE UNTIL ACTIVE GRID REACHES     CDT1454
C                JSTR(INPUT PARAMETER FOR TURNING ON STRENGTH           CDT1456
C                CALCULATIONS).                                         CDT1458
      IF ((IMAX.NE.1.AND.I.EQ.IMAX).OR.J.EQ.JMAX.OR.I2.LT.JSTR) P(K)=0. CDT1460
C                                                                       CDT1470
C                  ***PRESSURE IS NEGATIVE OR ZERO                      CDT1480
      WS=CNAUT                                                          CDT1490
      GO TO 40                                                          CDT1500
C                                                                       CDT1510
C                       ***PRESSURE IS POSITIVE.                        CDT1520
30    WS=CNAUT+BBAR*SQRT(P(K))                                          CDT1530
      WSA=SQRT(GAMMA*P(K)/RHOW)                                         CDT1540
      WS=AMAX1(WS,WSA)                                                  CDT1550
C            *** WS IS SOUND SPEED OF CELL K.                           CDT1552
C            *** WSB IS MAXIMUM OF RADIAL AND AXIAL VELOCITY OF CELL K. CDT1554
C            *** WSC STORES MAXIMUM VELOCITY OF CELLS USED TO DETERMINE CDT1556
C                DT. PRINTED AS MAXUV.                                  CDT1558
40    WSB=AMAX1(ABS(U(K)),ABS(V(K)))                                    CDT1560
      WSC=AMAX1(WSC,WSB)                                                CDT1570
      WS=AMAX1(WS,WSB)                                                  CDT1580
C            *** TRIAL STORES MAXIMUM OF VELOCTY AND SOUND SPEED USED   CDT1582
C                TO DETERMINE DT. PRINTED AS MAXCUV.                    CDT1584
      IF (WS.LE.TRIAL) GO TO 50                                         CDT1590
      TRIAL=WS                                                          CDT1600
50    IF (WS.LE.0.) GO TO 60                                            CDT1610
      DXYMIN=AMIN1(DX(I),DY(J))                                         CDT1620
      RATIO=DXYMIN/WS                                                   CDT1630
      IF (RATIO.GT.SRATIO) GO TO 60                                     CDT1640
C            *** I AND J OF CELL CONTROLLING DT STORED IN N10 AND N11   CDT1642
C                FOR PRINTOUT.                                          CDT1644
      N10=I                                                             CDT1650
      N11=J                                                             CDT1660
C            *** SRATIO IS SMALLEST VALUE CALCULATED FOR RATIO.         CDT1665
      SRATIO=RATIO                                                      CDT1670
```

```
C                                                                       CDT1680
C                    ***END OF I, J LOOP                                CDT1690
60    K=K+IMAX                                                          CDT1700
C            *** IF TRIAL.LE.0. THERE IS PROBABLY AN ERROR IN THE INPUT CDT1702
C                PARAMETERS FOR THE INITIAL VELOCITY, ENERGY OR DENSITY CDT1704
C                OF THE PACKAGES.                                       CDT1706
65    IF (TRIAL.LE.0.) GO TO 170                                        CDT1710
C            *** IF FINAL.EQ.0.USE STAB FOR VALUE OF STABILITY FRACTION CDT1720
C                IF FINAL.GT.0.USE A GEOMETRIC PROGRESSION WITH STAB    CDT1730
C                AS THE INITIAL VALUE AND FINAL AS THE FINAL VALUE.     CDT1740
      IF (FINAL.EQ.0.) GO TO 70                                         CDT1750
      STAB=2.*STAB                                                      CDT1760
      STAB=AMIN1(STAB,FINAL)                                            CDT1770
70    DT=STAB*SRATIO                                                    CDT1780
      IF (STAB.LT.FINAL) GO TO 80                                       CDT1790
C            *** AFTER STAB.GE.FINAL CHECK ON SIZE OF DT. DTMIN IS AN   CDT1792
C                INPUT PARAMETER AND CAN BE SET TO 0.                   CDT1794
75    IF (DT.LE.DTMIN) GO TO 150                                        CDT1800
80    CONTINUE                                                          CDT1810
C                                                                       CDT1820
C            *** IS CONTROL-CELL ISOLATED                               CDT1830
      K=(N11-1)*IMAX+N10+1                                              CDT1840
      WS=0.                                                             CDT1850
      IF (N10.GT.1) WS=AMX(K-1)                                         CDT1860
      IF (N10.LT.IMAX) WS=AMX(K+1)+WS                                   CDT1870
      IF (N11.GT.1) WS=AMX(K-IMAX)+WS                                   CDT1880
      IF (N11.LT.JMAX) WS=AMX(K+IMAX)+WS                                CDT1890
      IF (WS.GT.0.) GO TO 90                                            CDT1900
C           *** ISOLATED, SO DESTROY IT.                                CDT1910
      WS=(AIX(K)+(U(K)**2+V(K)**2)*.5)*AMX(K)                           CDT1920
      EVAPM=EVAPM+AMX(K)                                                CDT1930
      EVAPEN=EVAPEN+WS                                                  CDT1940
      ETH=ETH-WS                                                        CDT1950
      EVAPMU=EVAPMU+AMX(K)*U(K)                                         CDT1960
      EVAPMV=EVAPMV+AMX(K)*V(K)                                         CDT1970
      WRITE (6,290) N10,N11,T,DT,TRIAL,WSC,UMIN,PMIN                    CDT1980
      AMX(K)=0.                                                         CDT1990
      AIX(K)=0.                                                         CDT2000
      P(K)=0.                                                           CDT2010
      U(K)=0.                                                           CDT2020
      V(K)=0.                                                           CDT2030
C            *** RECALCULATE DT.                                        CDT2035
      GO TO 10                                                          CDT2040
C            *** INCREMENT TIME AND CYCLE.                              CDT2050
90    T=T+DTNA                                                          CDT2060
95    IF (T.LT.0.) GO TO 160                                            CDT2070
      NC=NC+1                                                           CDT2080
      CYCLE=NC                                                          CDT2090

C            *** RESET NPRINT. NPRINT=1 ON PRINT CYCLES.                CDT2100
      NPRINT=0                                                          CDT2110
C            *** DEFINE VELOCITY AND ENERGY CUTOFFS USED IN MAP AND PH2.CDT2115
      UMIN=TRIAL*ROEPS                                                  CDT2120
      SIEMIN=UMIN**2                                                    CDT2130
      PMIN=RHOZ*CNAUT*UMIN                                              CDT2140
      IF (PMIN.LT.ROEPS) PMIN=UMIN*RHOZ*TRIAL                           CDT2150
      WRITE (6,290) N10,N11,T,DT,TRIAL,WSC,UMIN,PMIN                    CDT2160
```

```
      DTNA=DT                                                           CDT2170
C            *** TESTRH = .2*RHOZ                                       CDT2171
C                THE PRESSURE OF COLD,FREE SURFACE CELLS IS REDUCED BY ACDT2172
C                FACTOR,F, WHICH ACCOUNTS FOR THE EFFECT OF FREE SURFACECDT2173
C                LOCATION ON THE PRESSURE GRADIENT. F IS THE DENSITY OF CDT2174
C                THE LOWEST DENSITY ADJACENT CELL DIVIDED BY THE NORMAL CDT2175
C                DENSITY,OR F IS TESTRH - WHICHEVER IS SMALLEST         CDT2176
      WT=TESTRH                                                         CDT2180
      DO 140 I=1,I1                                                     CDT2190
      K=I+1                                                             CDT2200
      DO 140 J=1,I2                                                     CDT2210
      RHOW=AMX(K)/(DY(J)*TAU(I))                                        CDT2220
      WTB=WT                                                            CDT2230
      IF (AIX(K).GE.ESESQ) GO TO 140                                    CDT2240
      IF (RHOW.LT.SOLID) GO TO 140                                      CDT2250
      IF (I.EQ.IMAX) GO TO 100                                          CDT2260
      WTA=AMX(K+1)/(DY(J)*TAU(I+1))                                     CDT2270
      IF (WTA.LT.WT) WTB=WTA                                            CDT2280
100   IF (I.EQ.1) GO TO 110                                             CDT2290
      WTA=AMX(K-1)/(DY(J)*TAU(I-1))                                     CDT2300
      IF (WTA.LT.WTB) WTB=WTA                                           CDT2310
110   IF (J.EQ.JMAX) GO TO 120                                          CDT2320
      KA=K+IMAX                                                         CDT2330
      WTA=AMX(KA)/(DY(J+1)*TAU(I))                                      CDT2340
      IF (WTA.LT.WTB) WTB=WTA                                           CDT2350
120   IF (J.EQ.1) GO TO 130                                             CDT2360
      KB=K-IMAX                                                         CDT2370
      WTA=AMX(KB)/(DY(J-1)*TAU(I))                                      CDT2380
      IF (WTA.LT.WTB) WTB=WTA                                           CDT2390
130   IF (WTB.LT.WT) P(K)=P(K)*WTB/RHOZ                                 CDT2400
140   K=K+IMAX                                                          CDT2410
      GO TO 190                                                         CDT2420
C                                                                       CDT2430
C                    *** DT TOO SMALL                                   CDT2440
150   NK=75                                                             CDT2450
      GO TO 180                                                         CDT2460
C                    *** T IS NEGATIVE                                  CDT2470
160   NK=95                                                             CDT2480
      GO TO 180                                                         CDT2490
C            *** DT WILL BE NEGATIVE OR ZERO.                           CDT2500
170   NK=65                                                             CDT2510
      GO TO 180                                                         CDT2520
180   NR=3                                                              CDT2530
      CALL ERROR                                                        CDT2540
C                                                                       CDT2550
C                      ***FIND THE MAXIMUM PRESSURE ON EACH COLUMN AND  CDT2560
C                      ***STORE ITS CELL NUMBER AS JPM. THIS WILL BE USED CDT2570
C                      ***IN DETERMINING THE REGION IN WHICH PHASE 3 IS CDT2580
C                      ***USED. WSA WILL BE A RUNNING MAXIMUM OF THE    CDT2590
C                      ***PRESSURE IN THE GRID.                         CDT2600
190   WSA=-1.E30                                                        CDT2610
      DO 260 I=1,I1                                                     CDT2620
C            *** WS WILL BE LOCAL MAXIMUM OF COLUMN I.                  CDT2630
      WS=-1.E30                                                         CDT2630
      K=(I2-1)*IMAX+I+1                                                 CDT2640
      JP=I2                                                             CDT2650
      JINTL=1                                                           CDT2660
```

```
C            *** START AT TOP OF COLUMN AND LOOK FOR PRESSURE PEAK.
200   DO 210 J=JINTL,I2
      IF (P(K).LT.WS) GO TO 220
      WS=P(K)
C            *** JP IS J-INDEX OF CELL WITH PEAK PRESSURE.
      JP=JP-1
210   K=K-IMAX
C            *** IF YOU FALL THROUGH,THEN THERE WAS NO MAXIMUM IN THIS
C                COLUMN
      GO TO 250
C
C                 *** COME HERE IF PRESSURE HAS PASSED A LOCAL MAXIMUM
C            *** PTEMP IS PEAK PRESSURE OF COLUMN I.
220   PTEMP=P(K+IMAX)
      IF (PTEMP.LT.WSA) GO TO 230
C            *** WSA WILL BE PEAK PRESSURE IN ACTIVE GRID (ABSOLUTE
C                MAXIMUM).
      WSA=PTEMP
      GO TO 240
C
C                 *** PTEMP IS LOCAL MAXIMUM BUT IS LESS THAN ABSOLUTE
C                MAXIMUM
230   IF (PTEMP.GT.0.3*WSA) GO TO 240
C
C            *** THIS LOCAL MAXIMUM IS NOT BIG ENOUGH TO USE FOR JPM
C
      JINTL=J+1
C
C
      JP=JP-1
C            ***  WE MAY HAVE REACHED BOTTOM OF COLUMN
      IF (JINTL.GE.I2) GO TO 250
C            *** CONTINUE DOWN COLUMN SEARCHING FOR SUFFICIENTLY LARGE
C                LOCAL MAXIMUM.
      WS=P(K)
      K=K-IMAX
      GO TO 200
C            *** IF POSITION OF PEAK PRESSURE IN COLUMN I DOES NOT
C                ADVANCE FROM ONE CYCLE TO THE NEXT, DO NOT CHANGE
C                VALUE OF JPM.
240   JP=JP+1
      IF (JP.LE.JPM(I)) GO TO 260
      JPM(I)=JP
C
C                    ***IF JPM IS ZERO THE SHOCK HAS NEVER REACHED THIS
C                    ***LOCATION. IF IT IS NONZERO THE SHOCK HAS PASSED
C                    ***AND WE MUST    CONTINUE TO INCREASE I UNTIL THE
C                    ***RIGHT BOUNDARY OF THE SHOCK IS REACHED.
250   IF (JPM(I).LE.0) GO TO 270
C            *** END OF I LOOP.
260   CONTINUE
C            *** IF PEAK PRESSURE OF COLUMN I HAS GONE BELOW A THIRD
C                THE GRID MAXIMUM, AND IF JPM(I)=0. FROM THE PREVIOUS
C                CYCLE, WE HAVE REACHED THE RIGHT EDGE OF THE SHOCK.
270   CONTINUE
C             *** JPM(I) MUST BE MONOTONIC DECREASING
      K=I1-1
```

```
      DO 280 IWS=1,K                                                    CDT3160
      I=I1-IWS                                                          CDT3170
280   IF (JPM(I).LT.JPM(I+1)) JPM(I)=JPM(I+1)                           CDT3180
      RETURN                                                            CDT3190
C                                                                       CDT3200
290   FORMAT (/4H CDT,I3,I4,4H  T=,1PE13.7,5H  DT=,1PE13.7,9H  MAXCUV=,1CDT3210
     1PE13.7,8H  MAXUV=,1PE13.7,7H  UMIN=,1PE13.7,7H  PMIN=,1PE13.7)    CDT3220
      END                                                               CDT3230-
```

```
      SUBROUTINE ES                                                     ES   10
C     ..................................................................ES   20
C                                                                       ES   30
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,      ES   40
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,                ES   50
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),   ES   60
     3          GAMC(102).                                             ES   70
     4          PK(15),   Z(150)   ,                                   ES   80
     5          XP(26,51),YP(26,51),                                   ES   90
     6          PL(204)  ,UL(204)  ,PR(204)  ,                         ES  100
     7           RSN(52),   RST(52),                                   ES  110
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,               ES  120
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,               ES  130
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)      ES  140
C            *** DIMENSIONED ARRAYS                                     ES  150
C            *** Z-BLOCK IS SAVED ON TAPE.                              ES  160
      COMMON      Z                                                     ES  170
      COMMON     PK                                                     ES  180
      COMMON  YY,      XX                                               ES  190
      COMMON  DDX,     DDY                                              ES  200
      COMMON  AMX,     AIX,     U,     V,     P                        ES  210
      COMMON  TAU,     JPM                                              ES  220
      COMMON  UL ,     PL                                               ES  230
      COMMON  XP ,     YP,      CMXP, CMYP                              ES  240
C            *** NON-DIMENSIONED VARIABLES                              ES  250
      COMMON          AID    ,AMMV  ,AMMY  ,AMPY  ,AMUR  ,AMUT  ,AMVR  ,ES  260
     1AMVT  ,DELEB ,DELER ,DELET ,DELM  ,DTODX  ,DXYMIN,EAMMP ,EAMPY ,  ES  270
     2E     ,ERDUMP,I     ,I3    ,IWS   ,J     ,K     ,KA    ,KB    ,   ES  280
     3LL    ,MD    ,ME    ,MZT   ,NERR  ,NK    ,NPRINT,                ES  290
     4NR    ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR   ,SNT   ,STR   ,SOLID ,  ES  300
     5SUM   ,TESTRH,TWOPI ,URR   ,WS    ,WSA   ,WSB   ,WSC   ,WFLAGF,   ES  310
     6WFLAGL,WFLAGP                                                     ES  320
C                                                                       ES  330
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE              ES  340
C                X(0), Y(0), DX(0), DY(0)                               ES  350
C                                                                       ES  360
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))                         ES  370
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))                     ES  380
C                                                                       ES  390
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY                      ES  400
C                                                                       ES  410
      EQUIVALENCE          (UL,FLEFT),          (UL(103),YAMC),         ES  420
     1                     (PL,GAMC,PR),        (PL(103),SIGC)          ES  430
C                                                                       ES  440
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY                      ES  450
C                                                                       ES  460
      EQUIVALENCE          (UL,RSN),                                    ES  470
     1                     (PL,RST),            (P,UK),                 ES  480
     2                     (P(157),VK),         (P(313),SNB),           ES  490
     3                     (P(365),STB),        (P(417),RHO)            ES  500
C                                                                       ES  510
C            *** SPECIAL EQUIVALENCES FOR EDIT                          ES  520
C                                                                       ES  530
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)                             ES  540
C                                                                       ES  550
C            *** Z-STORAGE EQUIVALENCES                                 ES  560
C                                                                       ES  570
      EQUIVALENCE                        (Z(  1),PROB  ),(Z(  2),CYCLE ), ES  580
```

```
     1(Z(  3),DT     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  ES  590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ), (Z(  9),TOPMU ),(Z( 10),RTMU  ), ES  600
     3(Z( 11),STK1  ),(Z( 12),NUMREZ), (Z( 13),ETH   ),(Z( 14),UN14  ), ES  610
     4(Z( 15),RHINIT),(Z( 16),PROJI ), (Z( 17),UN17  ),(Z( 18),XMAX  ), ES  620
     5(Z( 19),NZ    ),(Z( 20),NREZ  ), (Z( 21),AMDM  ),(Z( 22),UVMAX ), ES  630
     6(Z( 23),UN23  ),(Z( 24),DMIN  ), (Z( 25),JSTR  ),(Z( 26),DTNA  ), ES  640
     7(Z( 27),CVIS  ),(Z( 28),STK2  ), (Z( 29),STEZ  ),(Z( 30),NC    ), ES  650
     8(Z( 31),UN31  ),(Z( 32),NRC   ), (Z( 33),IMAX  ),(Z( 34),IMAXA ), ES  660
     9(Z( 35),JMAX  ),(Z( 36),JMAXA ), (Z( 37),KMAX  ),(Z( 38),KMAXA )  ES  670
      EQUIVALENCE                                                       ES  680
     1(Z( 39),BOTM  ),(Z( 40),BOTMV ), (Z( 41),NUMSPT),(Z( 42),CZERO ), ES  690
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ), (Z( 45),PRDELT),(Z( 46),PRFACT)  ES  700
      EQUIVALENCE                                                       ES  710
     1(Z( 47),I1    ),(Z( 48),I2    ), (Z( 49),IPCYCL),(Z( 50),TSTOP ), ES  720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ), (Z( 53),N3    ),(Z( 54),IVARDY), ES  730
     3(Z( 55),VT    ),(Z( 56),N6    ), (Z( 57),RTM   ),(Z( 58),RTMV  ), ES  740
     4(Z( 59),UN59  ),(Z( 60),N10   ), (Z( 61),N11   ),(Z( 62),GAMMA ), ES  750
     5(Z( 63),TOPM  ),(Z( 64),BOTMU ), (Z( 65),SN    ),(Z( 66),TOPMV ), ES  760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP), (Z( 69),PRXRT ),(Z( 70),CYCPH3), ES  770
     7(Z( 71),REZFCT),(Z( 72),TARGI ), (Z( 73),PROJU ),(Z( 74),BBOUND), ES  780
     8(Z( 75) EVAP  ),(Z( 76),ECK   ), (Z( 77),NECYCL),(Z( 78),II    ), ES  790
     9(Z( 79),JJ    ),(Z( 80),NMP   ), (Z( 81),Y2    ),(Z( 82),EZPH1 )  ES  800
      EQUIVALENCE                                                       ES  810
     1(Z( 83),IVARDX),(Z( 84),T     ), (Z( 85),NMPMAX),(Z( 86),PMIN  ), ES  820
     2(Z( 87),INTER ),(Z( 88),TAYBOT), (Z( 89),TAYTOP),(Z( 90),IEMAP ), ES  830
     3(Z( 91),MC    ),(Z( 92),MR    ), (Z( 93),MZ    ),(Z( 94),MB    )  ES  840
      EQUIVALENCE                                                       ES  850
     1(Z( 95),REZ   ),(Z( 96),NODUMP), (Z( 97),UN97  ),(Z( 98),UN98  ), ES  860
     2(Z( 99),UN99  ),(Z(100),EVAPM ), (Z(101),EVAPEN),(Z(102),EVAPMU), ES  870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ), (Z(105),SNL   ),(Z(106),STL   ), ES  880
     4(Z(107),TAXRT ),(Z(108),IDNMAP), (Z(109),IPRMAP),(Z(110),ROEPS ), ES  890
     5(Z(111),RHINI ),(Z(112),VINI  ), (Z(113),FINAL ),(Z(114),IVMAP ), ES  900
     6(Z(115),RHOZ  ),(Z(116),ESA   ), (Z(117),ESEZ  ),(Z(118),ESB   ), ES  910
     7(Z(119),ESCAPA),(Z(120),ESESP ), (Z(121),ESESQ ),(Z(122),ESES  ), ES  920
     8(Z(123),ESALPH),(Z(124),ESBETA), (Z(125),ESCAPB),(Z(126),IUMAP ), ES  930
     9(Z(127),SS1   ),(Z(128),SS2   ), (Z(129),UMIN  ),(Z(130),SS4   )  ES  940
      EQUIVALENCE                                                       ES  950
     1(Z(131),PRTIME),(Z(132),EOR   ), (Z(133),EOT   ),(Z(134),EOB   ), ES  960
     2(Z(135),EMOR  ),(Z(136),DXF   ), (Z(137),DYF   ),(Z(138),RHOMIN), ES  970
     3(Z(139),STAB), (Z(140),XIENRG),  (Z(141),XKENRG), (Z(142),XTENRG),ES  980
     4(Z(143),STT   ),(Z(144),DTMIN ), (Z(145),TRNSFC),(Z(146),EMOT  ), ES  990
     5(Z(147),JPROJ ),(Z(148),CNAUT ), (Z(149),BBAR  ),(Z(150),EMOB  )  ES 1000
C                                                                       ES 1010
C            *** SPECIAL EQUIV FOR ES AND CDT                           ES 1012
      EQUIVALENCE (RHOW,NULLE)                                          ES 1014
C     ..................................................................ES 1020
C                                                                       ES 1030
C     END OF COMMON                                                     ES 1040
C                                                                       ES 1050
C     ..................................................................ES 1060
C            *** P(K) CALCULATED FROM RHOW AND AIX(K).                  ES 1065
C                RHOW IS CALCULATED IN CDT                              ES 1070
C                                                                       ES 1075
      IF (ESCAPA.LE.0.) GO TO 30                                        ES 1110
      ETA=RHOW/RHOZ                                                     ES 1120
      VOW=1./ETA                                                        ES 1130
      IF (AIX(K).LE.0.) GO TO 20                                        ES 1140
```

```
C           *** P1 AND P4 ARE THERMAL PRESSURE TERMS.                    ES 1145
      P1=AIX(K)*RHOW*ESA                                                 ES 1150
      P4=ESB/(AIX(K)/(ESEZ*ETA**2)+1.)*AIX(K)*RHOW                       ES 1160
C           *** P5 IS MECHANICAL PRESSURE TERM                           ES 1165
10    P5=ESCAPA*(ETA-1.)                                                 ES 1170
      P2=-1.                                                             ES 1180
C           *** IF MATERIAL IS UNDER-DENSE AND ITS ENERGY IS BETWEEN     ES 1182
C               ESES AND ESESP, A COMBINATION OF THE EXPANDED AND        ES 1184
C               CONDENSED EQUATIONS OF STATE IS USED.                    ES 1186
      IF (ETA.GE.1.) GO TO 50                                            ES 1190
C           *** ESESP = ENERGY TO VAPORIZE MATERIAL. MUST EXCEED ESES.   ES 1195
      IF (AIX(K).GT.ESESP) GO TO 40                                      ES 1200
C           *** ESES = ENERGY TO BRING MATERIAL TO VAPOR TEMPERATURE.    ES 1205
      IF (AIX(K).GT.ESES) P2=1.                                          ES 1210
C           *** P2=1 MEANS BOTH THE EXPANDED AND COMPRESSED              ES 1212
C                   FORMULATIONS WILL BE USED.  OTHERWISE, P2=-1.        ES 1214
      GO TO 50                                                           ES 1220
C           *** WHEN SPECIFIC INTERNAL ENERGY OF CELL IS NEGATIVE,       ES 1222
C               THERMAL PRESSURES ARE SET TO ZERO.                       ES 1224
20    P1=0.                                                              ES 1230
      P4=0.                                                              ES 1240
C           *** WHEN SPECIFIC INTERNAL ENERGY IS NEGATIVE OR ZERO AND    ES 1242
C               DENSITY IS LESS THAN SOLID, SET PRESSURE TO ZERO.        ES 1244
      IF (ETA.LT.AMDM) GO TO 80                                          ES 1250
      GO TO 10                                                           ES 1260
C           *** IDEAL GAS                                                ES 1265
30    P(K)=ESA*RHOW*AIX(K)                                               ES 1270
      GO TO 90                                                           ES 1280
C           *** EXPANDED STATE                                           ES 1290
40    P8=(1.-VOW)                                                        ES 1300
      P9=EXP(ESALPH*P8)                                                  ES 1310
      P12=EXP(-ESBETA*P8**2)                                             ES 1320
      P(K)=P1+(P4+P5*P9)*P12                                             ES 1330
      IF (P2.LT.0.) GO TO 70                                             ES 1340
      P1=SS1*(AIX(K)-ESES)                                               ES 1350
      P(K)=P1*P(K)+(1.-P1)*P3                                            ES 1360
      GO TO 70                                                           ES 1370
C           *** CONDENSED STATE                                          ES 1380
C               P6 IS MECHANICAL PRESSURE TERM.                          ES 1385
50    P6=ESCAPB*((ETA-1.)**2)                                            ES 1390
      P(K)=P1+P4+P5+P6                                                   ES 1400
      IF (P2.LT.0.) GO TO 60                                             ES 1410
C           *** USING COMBINATION OF CONDENSED AND EXPANDED EQUATIONS    ES 1412
C               OF STATE.                                                ES 1414
      IF (P(K).LT.0.) P(K)=0.                                            ES 1420
      P3=P(K)                                                            ES 1430
      GO TO 40                                                           ES 1440
C           *** USING CONDENSED EQUATION OF STATE                        ES 1445
60    IF (P(K).GE.0.) GO TO 90                                           ES 1450
C           *** IF MATERIAL IS EXPANDED  OR J-INDEX OF CELL IS LESS      ES 1452
C               THAN N6, SET NEGATIVE PRESSURE TO ZERO. (N6 IS INPUT     ES 1454
C               PARAMETER)                                               ES 1456
      IF (J.LE.N6.OR.ETA.LE.AMDM) GO TO 80                               ES 1460
      GO TO 90                                                           ES 1470
C           *** SET NEGATIVE PRESSURES TO ZERO  WHEN USING COMBINED      ES 1472
C                               OR EXPANDED EQUATIONS OF STATE.          ES 1474
70    IF (P(K).GE.0.) GO TO 90                                           ES 1480
```

```
80      P(K)=0.                                          ES 1490
90      RETURN                                           ES 1500
        END                                              ES 1510-
```

```
      SUBROUTINE EDIT
C     ..................................................................
C
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  , CRAD(52),
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),
     3          GAMC(102),
     4          PK(15),   Z(150)   ,
     5          XP(26,51),YP(26,51),
     6          PL(204)  ,UL(204)  ,PR(204)  ,
     7           RSN(52),   RST(52),
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)
C            *** DIMENSIONED ARRAYS
C            *** Z-BLOCK IS SAVED ON TAPE.
      COMMON     Z
      COMMON    PK
      COMMON  YY,      XX
      COMMON  DDX,     DDY
      COMMON  AMX,     AIX,     U,     V,     P
      COMMON  TAU,     JPM
      COMMON  UL ,     PL
      COMMON  XP ,     YP,      CMXP, CMYP
C            *** NON-DIMENSIONED VARIABLES
      COMMON         AID    ,AMMV   ,AMMY   ,AMPY   ,AMUR   ,AMUT   ,AMVR   ,
     1AMVT   ,DELEB ,DELER ,DELET ,DELM   ,DTODX ,DXYMIN,EAMMP ,EAMPY ,
     2E      ,ERDUMP,I     ,I3    ,IWS    ,J      ,K      ,KA     ,KB     ,
     3LL     ,MD    ,ME    ,MZT   ,NERR   ,NK     ,NPRINT,
     4NR     ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR    ,SNT    ,STR    ,SOLID ,
     5SUM    ,TESTRH,TWOPI ,URR   ,WS     ,WSA    ,WSB    ,WSC    ,WFLAGF,
     6WFLAGL,WFLAGP
C
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE
C                X(0), Y(0), DX(0), DY(0)
C
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))
C
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY
C
      EQUIVALENCE          (UL,FLEFT),          (UL(103),YAMC),
     1                     (PL,GAMC,PR),        (PL(103),SIGC)
C
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY
C
      EQUIVALENCE          (UL,RSN),
     1                     (PL,RST),            (P,UK),
     2                     (P(157),VK),         (P(313),SNB),
     3                     (P(365),STB),        (P(417),RHO)
C
C            *** SPECIAL EQUIVALENCES FOR EDIT
C
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK),  (UL(103),CRAD)
C
C            *** Z-STORAGE EQUIVALENCES
C
      EQUIVALENCE                        (Z(  1),PROB  ),(Z(  2),CYCLE ),
```

```
     1(Z(  3),DT     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ), (Z(  9),TOPMU ),(Z( 10),RTMU  ),
     3(Z( 11),STK1  ),(Z( 12),NUMREZ), (Z( 13),ETH   ),(Z( 14),UN14  ),
     4(Z( 15),RHINIT),(Z( 16),PROJI ), (Z( 17),UN17  ),(Z( 18),XMAX  ),
     5(Z( 19),NZ    ),(Z( 20),NREZ  ), (Z( 21),AMDM  ),(Z( 22),UVMAX ),
     6(Z( 23),UN23  ),(Z( 24),DMIN  ), (Z( 25),JSTR  ),(Z( 26),DTNA  ),
     7(Z( 27),CVIS  ),(Z( 28),STK2  ), (Z( 29),STEZ  ),(Z( 30),NC    ),
     8(Z( 31),UN31  ),(Z( 32),NRC   ), (Z( 33),IMAX  ),(Z( 34),IMAXA ),
     9(Z( 35),JMAX  ),(Z( 36),JMAXA ), (Z( 37),KMAX  ),(Z( 38),KMAXA )
      EQUIVALENCE
     1(Z( 39),BOTM  ),(Z( 40),BOTMV ), (Z( 41),NUMSPT),(Z( 42),CZERO ),
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ), (Z( 45),PRDELT),(Z( 46),PRFACT)
      EQUIVALENCE
     1(Z( 47),I1    ),(Z( 48),I2    ), (Z( 49),IPCYCL),(Z( 50),TSTOP ),
     2(Z( 51),RHOFIL),(Z( 52),TARGV ), (Z( 53),N5    ),(Z( 54),IVARDY),
     3(Z( 55),VT    ),(Z( 56),N6    ), (Z( 57),RTM   ),(Z( 58),RTMV  ),
     4(Z( 59),UN59  ),(Z( 60),N10   ), (Z( 61),N11   ),(Z( 62),GAMMA ),
     5(Z( 63),TOPM  ),(Z( 64),BOTMU ), (Z( 65),SN    ),(Z( 66),TOPMV ),
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP), (Z( 69),PRXRT ),(Z( 70),CYCPH3),
     7(Z( 71),REZFCT),(Z( 72),TARGI ), (Z( 73),PROJU ),(Z( 74),BBOUND),
     8(Z( 75),EVAP  ),(Z( 76),ECK   ), (Z( 77),NECYCL),(Z( 78),II    ),
     9(Z( 79),JJ    ),(Z( 80),NMP   ), (Z( 81),Y2    ),(Z( 82),EZPH1 )
      EQUIVALENCE
     1(Z( 83),IVARDX),(Z( 84),T     ), (Z( 85),NMPMAX),(Z( 86),PMIN  ),
     2(Z( 87),INTER ),(Z( 88),TAYBOT), (Z( 89),TAYTOP),(Z( 90),IEMAP ),
     3(Z( 91),MC    ),(Z( 92),MR    ), (Z( 93),MZ    ),(Z( 94),MB    )
      EQUIVALENCE
     1(Z( 95),REZ   ),(Z( 96),NODUMP), (Z( 97),UN97  ),(Z( 98),UN98  ),
     2(Z( 99),UN99  ),(Z(100),EVAPM ), (Z(101),EVAPEN),(Z(102),EVAPMU),
     3(Z(103),EVAPMV),(Z(104),EZPH2 ), (Z(105),SNL   ),(Z(106),STL   ),
     4(Z(107),TAXRT ),(Z(108),IDNMAP), (Z(109),IPRMAP),(Z(110),ROEPS ),
     5(Z(111),RHINI ),(Z(112),VINI  ), (Z(113),FINAL ),(Z(114),IVMAP ),
     6(Z(115),RHOZ  ),(Z(116),ESA   ), (Z(117),ESEZ  ),(Z(118),ESB   ),
     7(Z(119),ESCAPA),(Z(120),ESESP ), (Z(121),ESESQ ),(Z(122),ESES  ),
     8(Z(123),ESALPH),(Z(124),ESBETA), (Z(125),ESCAPB),(Z(126),IUMAP ),
     9(Z(127),SS1   ),(Z(128),SS2   ), (Z(129),UMIN  ),(Z(130),SS4   )
      EQUIVALENCE
     1(Z(131),PRTIME),(Z(132),EOR   ), (Z(133),EOT   ),(Z(134),EOB   ),
     2(Z(135),EMOR  ),(Z(136),DXF   ), (Z(137),DYF   ),(Z(138),RHOMIN),
     3(Z(139),STAB), (Z(140),XIENRG),  (Z(141),XKENRG), (Z(142),XTENRG),
     4(Z(143),STT   ),(Z(144),DTMIN ), (Z(145),TRNSFC),(Z(146),EMOT  ),
     5(Z(147),JPROJ ),(Z(148),CNAUT ), (Z(149),BBAR  ),(Z(150),EMOB  )
C
C     ..............................................................
C
C     END OF COMMON
C
C     ..............................................................
C
C           *** SPECIAL EQUIV. FOR EDIT
      EQUIVALENCE (PR(1),TIETAR),(PR(2),TKETAR),(PR(3),TETAR ),
     1            (PR(4),TARMAS),(PR(5),TARMV ),(PR(6),TARMVP),
     2            (PR(7),RAMOMA),(PR(8),PRAMOA),(PR(9),TIEPRO),
     3            (PR(10),TKEPRO),(PR(11),TEPRO),(PR(12),PRMAS),
     4            (PR(13),PRMV  ),(PR(14),PRMVP),(PR(15),RAMOMB),
     5            (PR(16),PRAMOB)
      DIMENSION  PROPI(50)
```

```
C                                                                       1160
C             *** ERDUMP=1. WHEN ERROR CALLS EDIT FOR A TAPE DUMP ONLY  1162
      IF (ERDUMP.GT.0.) GO TO 150                                       1170
C             *** ENERGY SUM (ESUM) AND RELATIVE ERROR IN SUM (RELERR)  1172
C                 COMPUTED.  ECK IS LARGEST ERROR COMPUTED AND ON PRINT 1174
C                 CYCLES IS PRINTED AND COMPARED TO DMIN, MAXIMUM       1176
C                 ALLOWABLE  ERROR.                                     1178
      ESUM=0.                                                           1180
      DO 10 K=2,KMAX                                                    1190
10    ESUM=ESUM+AMX(K)*(.5*(U(K)**2+V(K)**2)+AIX(K))                    1200
      RELERR=(ESUM-ETH)/ETH                                             1210
      IF (ABS(RELERR).LT.ABS(ECK)) GO TO 20                             1220
      ECK=RELERR                                                        1230
      NECYCL=NC                                                         1240
20    CONTINUE                                                          1250
C             *** NPRINT = 1  WHEN EDIT IS CALLED TO DO AN INTERMEDIATE 1252
C                 PRINT. SKIP TESTS ON TIME TO STOP, PRINT, REZONE,ETC. 1254
C                 WHICH ALREADY HAVE BEEN DONE FOR THIS CYCLE.          1256
      IF (NPRINT.EQ.1) GO TO 190                                        1260
C             *** I3=1 SIGNALS A SHORT PRINT                            1270
      I3=1                                                              1280
C             *** IF THIS IS FIRST CYCLE OF RUN, WFLAGF=1.              1290
      IF (WFLAGF.GT.0.) GO TO 120                                       1300
C             *** IS THIS THE TIME OR CYCLE TO STOP EXECUTION           1305
      IF (ICSTOP.LE.NC.AND.ICSTOP.GT.0) GO TO 30                        1310
      IF (T*(1.+ROEPS).GE.TSTOP.AND.TSTOP.GT.0.) GO TO 30               1320
C             *** SHOULD THE GRID BE REZONED                            1325
      IF ((REZ.NE.0..AND.REZFCT.NE.0..AND.NUMREZ.GT.0).OR.SS4.NE.0.) GO 1330
     1TO 190                                                            1340
C                                                                       1350
      GO TO 40                                                          1360
C                 *** SET WFLAGL=1. TO SAY THIS IS LAST CYCLE OF RUN    1370
30    WFLAGL=1.                                                         1380
      I3=I1                                                             1390
      NPRINT=1                                                          1400
      NUMSPT=NDUMP7                                                     1410
      NUMSP=0                                                           1420
      GO TO 190                                                         1430
40    ASSIGN 140 TO LOCA                                                1440
      ASSIGN 110 TO LOCB                                                1450
C             *** ARE WE PRINTING ON TIME OR CYCLE INTERVALS            1455
      IF (PRDELT.NE.0.) GO TO 50                                        1460
45    IF (IPCYCL.NE.0) GO TO 100                                        1470
      GO TO 430                                                         1480
C             *** PRINTING ON TIME. IS IT TIME TO PRINT                 1485
50    IF (T*(1.+ROEPS).GE.PRTIME) GO TO 70                              1490
C             *** NO. BUT WILL NEXT CYCLE BYPASS THE PRINT TIME         1495
      IF (PRTIME.GE.T+DT) GO TO 60                                      1500
      DT=PRTIME-T                                                       1510
      DTNA=DT                                                           1520
60    GO TO LOCA, (140,130)                                             1530
C             *** YES, IT IS TIME TO PRINT. NPRINT=1 FLAGS THIS AS A    1532
C                 PRINT CYCLE.                                          1534
70    NPRINT=1                                                          1540
C             *** AVOID TRUNCATION                                      1550
      T=PRTIME                                                          1560
C             *** IS IT TIME TO RESCALE PRINT INTERVAL                  1565
```

```
      IF (T*(1.+ROEPS).LT.PRLIM.OR.NUMSCA.LE.0) GO TO 80
C           *** CHANGE PRINT INTERVAL AND THE TIME FOR THE NEXT
C               RESCALING.
      PRDELT=PRDELT*PRFACT
      PRLIM=PRLIM*PRFACT
      NUMSCA=NUMSCA-1
C           *** DEFINE TIME FOR NEXT PRINT.
 80   PRTIME=T+PRDELT
      IWS=(PRTIME+.5*PRDELT)/PRDELT
      WS=IWS
      PRTIME=WS*PRDELT
C           *** WILL WE BYPASS TIME TO PRINT
      IF (PRTIME.GE.T+DT) GO TO 90
C           *** YES, ADJUST DT
      DT=PRTIME-T
      DTNA=DT
 90   GO TO LOCB, (110,130)
C           *** PRINTING ON CYCLES. IS THIS A PRINT CYCLE
100   IF (MOD(NC,IPCYCL).NE.0) GO TO LOCA, (140,130)
C           *** YES. NPRINT = 1 FLAGS THIS AS A PRINT CYCLE.
      NPRINT=1
C           *** IS THIS THE CYCLE TO RESCALE PRINT INTERVAL
      IF (NC.LT.PRLIM.OR.NUMSCA.LE.0) GO TO LOCB, (110,130)
C
C           *** YES. MULTIPLY NUMBER OF CYCLES BETWEEN PRINTS BY PRFACT
C
      IPCYCL=INT(PRFACT)*IPCYCL
      PRLIM=PRFACT*PRLIM
      NUMSCA=NUMSCA-1
      GO TO LOCB, (110,130)
C               *** TEST FOR SHORT OR LONG PRINT
C           *** NUMSP COUNTS NUMBER OF SHORT PRINTS SINCE LAST LONG
C               PRINT. NUMSPT COUNTS NUMBER OF CYCLES SINCE LAST
C               TAPE DUMP.
110   NUMSP=NUMSP+1
      NUMSPT=NUMSPT+1
      IF (NUMSP.NE.NFRELP) GO TO 190
      NUMSP=0
C               *** I3=I1 SIGNALS A LONG PRINT
120   I3=I1
C           *** PRINT OF RESTART CYCLE WILL BE SHORT IF PK(3).LT.-1.
      IF (PK(3).LT.-1..AND.WFLAGF.GT.0.) I3=1
      GO TO 190
C               *** CHECK FOR ENERGY DISCREPANCY
130   IF (ABS(ECK).GT.DMIN) GO TO 440
C               *** IF LAST CYCLE, REWIND TAPE
140   IF (WFLAGL.EQ.0.) GO TO 470
      REWIND 7
      GO TO 470
150   NUMSPT=0
      IF (NODUMP.NE.0) GO TO 170
      BACKSPACE 7
      WS=555.0
      WRITE (7) WS,CYCLE,N3
      WRITE (7) (Z(L),L=1,MZT)
      WRITE (7) (U(K),V(K),AMX(K),AIX(K),P(K),K=1,KMAXA)
      WRITE (7) X(0),(X(K),TAU(K),JPM(K),K=1,IMAX)
```

```
      WRITE (7) (Y(K),K=0,JMAX)
C            *** ARE TRACER POINTS BEING GENERATED
      IF (Y2.GT.(-1.)) GO TO 160
C            *** YES. WRITE TRACER POINT COORDINATES (XP,YP) ON TAPE.
      WRITE (7) ((XP(I,J),YP(I,J),I=1,II),J=1,JJ)
160   WRITE (7) (DX(I),I=1,IMAX)
      WRITE (7) (DY(J),J=1,JMAX)
      WS=666.0
      WRITE (7) WS,WS,WS
      WRITE (6,550) NC
      IF (WFLAGL.EQ.0.) GO TO 170
      END FILE 7
170   CONTINUE
      IF (ERDUMP.GT.0.) CALL EXIT
      GO TO 280
180   N=2
      GO TO 220
C            *** INITIALIZE PR ARRAY, TEMPORARY STORAGE FOR ENERGY,MASS
C                AND MOMENTUM TOTALS PRINTED OUT.
190   DO 200 I=1,16
200   PR(I)=0.
C
C     RAMOMA=RADTAL MOMENTUM ABOVE JPROJ
C     RAMOMA=RADTAL MOMENTUM BELOW JPROJ
C     PRAMOA=POSITIVE RADIAL MOMENTUM ABOVE JPROJ
C     PRAMOB=POSITIVE RADIAL MOMENTUM BELOW JPROJ
C
      IF (JPROJ.EQ.0) GO TO 180
      N=IMAX*JPROJ+1
      DO 210 K=2,N
      WS=AMX(K)
      PRMAS=PRMAS+WS
      TIEPRO=TIEPRO+WS*AIX(K)
      TKEPRO=TKEPRO+.5*WS*(U(K)**2+V(K)**2)
      WSA=WS*V(K)
      PRMV=PRMV+WSA
      IF (WSA.GT.0.) PRMVP=PRMVP+WSA
      RAMOMB=RAMOMB+AMX(K)*U(K)
      IF (U(K).GT.0.) PRAMOB=PRAMOB+AMX(K)*U(K)
210   CONTINUE
      N=N+1
220   DO 230 K=N,KMAX
      WS=AMX(K)
      TARMAS=TARMAS+WS
      TIETAR=TIETAR+WS*AIX(K)
      TKETAR=TKETAR+.5*WS*(U(K)**2+V(K)**2)
      WSA=WS*V(K)
      TARMV=TARMV+WSA
      IF (WSA.GT.0.) TARMVP=TARMVP+WSA
      RAMOMA=RAMOMA+AMX(K)*U(K)
      IF (U(K).GT.0.) PRAMOA=PRAMOA+AMX(K)*U(K)
230   CONTINUE
      TETAR=TIETAR+TKETAR
      TEPRO=TIEPRO+TKEPRO
      DO 240 J=1,8
      PR(J+16)=PR(J)+PR(J+8)
240   CONTINUE
```

```
      IF (IMAX.GT.1) GO TO 260
C
C           *** IF DOING A 1-D PROBLEM DIVIDE TOTALS BY NZ WHERE
C               NZ=4**(NUMBER OF TIMES THE GRID HAS BEEN REZONED.)
C
      PROPI(1)=ETH/NZ
      PROPI(2)=ECK/NZ
      PROPI(4)=EZPH1/NZ
      PROPI(5)=EZPH2/NZ
      PROPI(6)=BBOUND/NZ
      DO 250 J=1,24
250   PROPI(J+6)=PR(J)/NZ
      PROPI(31)=BOTM/NZ
      PROPI(32)=RTM/NZ
      PROPI(33)=TOPM/NZ
      PROPI(34)=EVAPM/NZ
      PROPI(35)=EMOB/NZ
      PROPI(36)=EMOR/NZ
      PROPI(37)=EMOT/NZ
      PROPI(38)=EVAPEN/NZ
      PROPI(39)=BOTMU/NZ
      PROPI(40)=RTMU/NZ
      PROPI(41)=TOPMU/NZ
      PROPI(42)=EVAPMU/NZ
      PROPI(43)=BOTMV/NZ
      PROPI(44)=RTMV/NZ
      PROPI(45)=TOPMV/NZ
      PROPI(46)=EVAPMV/NZ
      PROPI(47)=EOB/NZ
      PROPI(48)=EOR/NZ
      PROPI(49)=EOT/NZ
      WRITE (6,530) PROB,T,NC,PROPI(1),PROPI(2),NECYCL,(PROPI(J),J=4:6)
      WRITE (6,540) (PROPI(J),J=7,49)
      GO TO 270
260   WRITE (6,530) PROB,T,NC,ETH,ECK,NECYCL,EZPH1,EZPH2,BBOUND
      WRITE (6,540) ((PR(J),J=1,24),BOTM,RTM,TOPM,EVAPM,EMOB,EMOR,EMOT,E
     1VAPEN,BOTMU,RTMU,TOPMU,EVAPMU,BOTMV,RTMV,TOPMV,EVAPMV,EOB,EOR,EOT)
270   WRITE (6,580) (JPM(I),I=1,I1)
C           *** ENERGY TOTALS STORED FOR LATER USE IN TRACER POINT
C               PLOTS.
      XIENRG=PR(17)
      XKENRG=PR(18)
      XTENRG=PR(19)
C           *** IS THIS A TAPE DUMP OR REZONE CYCLE
      IF (NUMSPT.EQ.NDUMP7.OR.(REZ.NE.0..AND.REZFCT.NE.0..AND.NUMREZ.GT.
     10)) GO TO 150
C           *** ARE TRACER POINTS BEING GENERATED
280   IF (Y2.GT.(-1.)) GO TO 305
C           *** YES, PRINT TRACER POINT COORDINATES IN CM.
      WRITE (6,590)
      N=0
      DO 300 J=1,JJ
      DO 300 I=1,II
      IF (XP(I,J).LE.0..AND.YP(I,J).LE.0.) GO TO 300
      IP=INT(XP(I,J))
      JP=INT(YP(I,J))
      KK=JP*IMAX+IP+2
```

```
      IF (AMX(KK).GT.0.) GO TO 290
      XP(I,J)=0.
      YP(I,J)=0.
      GO TO 300
290   N=N+1
      CMXP(N)=X(IP)+DX(IP+1)*(XP(I,J)-INT(XP(I,J)))
      CMYP(N)=Y(JP)+DY(JP+1)*(YP(I,J)-INT(YP(I,J)))
C
C            *** IJ, JK  = THE I AND J OF THE CELL THE TRACER POINT
C                  ORIGINATED IN .  (TRACER POINTS CHANGE POSITION IN
C                  XP AND YP ARRAYS WHEN THEY ARE WEEDED OUT
C                  DURING REZONE.)
C
      IJ(N)=2**(NRZ+1)*(I-1)+1
      JK(N)=2**(NRZ+1)*(J-1)+1
      IF (N.LT.5) GO TO 300
      WRITE (6,510) (IJ(M),JK(M),CMXP(M),CMYP(M),M=1,N)
      N=0
300   CONTINUE
      IF (N.EQ.0) GO TO 305
      WRITE (6,510) (IJ(M),JK(M),CMXP(M),CMYP(M),M=1,N)
305   IF (IMAX.EQ.1) GO TO 370
C            *** PRINT SYMBOLIC CONTOUR MAPS OF COMPRESSION, PRESSURE,
C                VELOCITY, AND INTERNAL ENERGY UNLESS DOING A 1-D
C                PROBLEM.
      CALL MAP
C            *** COMPUTE CRATER DEPTH AND VOLUME. AID SUMS DEPTH.
      AID = 0.
      WRITE(6,490)
C            *** START AT AXIS
      DO 330 I =1,I1
      CRAD(I) = .5*DX(I)+X(I-1)
      PL(I) = 0.
      UL(I) = 0.
      DO 320 J =1,I2
      K=(J-1)*IMAX + I + 1
C            *** WS IS COMPRESSION
      WS = AMX(K)/(TAU(I)*DY(J)*RHOZ)
      IF(WS.LT.(.99)) GO TO 310
      GO TO 325
  310 AID = AID + 1.-WS
C            *** NOT AT BOTTOM OF CRATER YET
  320 CONTINUE
  325 IAID = INT(AID)
C            *** UL(I) IS CM. DEPTH OF CRATER IN COLUMN I
C            *** PL(I) IS CELL DEPTH OF CRATER IN COLUMN I
      UL(I) = Y(IAID) + DY(IAID+1)*(AID-FLOAT(IAID)) - Y(JPROJ)
      IF(UL(I).GT.0..OR.UL(I).LT.0.) PL(I) = AID
      AID = 0.
  330 CONTINUE
C            *** PRINT CRATER DEPTHS
      DO 340 I=1,I1
      IF(UL(I).LT.0..OR.UL(I).GT.0.) GO TO 335
      GO TO 340
  335 WRITE(6,495) I, PL(I), CRAD(I), UL(I)
  340 CONTINUE
C            *** COMPUTE  CRATER VOLUME AND VOLUME OF HEMISPHERE WITH
```

```
C                 RADIUS=UL(1). PRINT VOLUMES WHEN THEY ARE POSITIVE.
      WSB=0.
      DO 345 I=1,I1
      IF(UL(I).LT.0.) GO TO 350
C             *** WSB GIVES CRATER VOLUME
      WSB = UL(I)*TAU(I)+WSB
  345 CONTINUE
  350 CONTINUE
C             *** PRINT CRATER VOLUME ONLY WHEN GREATER THAN ZERO
      IF(WSB.GT.0.) GO TO 355
      GO TO 360
C             *** WSC GIVES VOLUME OF HEMISPHERE
355   WSC=2.0944*(UL(1))**3
      WRITE(6,500) WSB, WSC
  360 CONTINUE
C             *** SHORT PRINT MEAN  I3=1 AND PROPERTIES ARE PRINTED ONLY
C                 FOR CELLS IN FIRST COLUMN. LONG PRINT MEANS I3=I1 AND
C                 PROPERTIES ARE PRINTED FOR ALL CELLS IN ACTIVE GRID.
370   DO 420 I=1,I3
      KSPACE=0
      WFLAGP=1.
      J=I2+1
      K=I2*IMAX+I+1
      DO 410 L=1,I2
      J=J-1
      K=K-IMAX
375   IF (AMX(K)) 450,400,380
380   IF (WFLAGP.EQ.0.) GO TO 390
      WRITE (6,560) I,X(I),DX(I)
      WFLAGP=0.
390   WS=AMX(K)/(TAU(I)*DY(J))
      WSA=WS/RHOZ
      WSC=P(K)
      WRITE (6,520) J,U(K),V(K),WSC,AMX(K),WS,AIX(K),WSA,Y(J)
      KSPACE=0
      GO TO 410
400   KSPACE=KSPACE+1
      IF (KSPACE.GT.1) GO TO 410
      WRITE (6,570)
410   CONTINUE
420   CONTINUE
      IF (NPRINT.EQ.1) GO TO 130
      ASSIGN 130 TO LOCA
      ASSIGN 130 TO LOCB
      IF (PRDELT.NE.0.) GO TO 50
      GO TO 100
C                 *** PRINT DELTA NOT SPECIFIED IN INPUT
430   NK=45
      GO TO 460
C             *** ENERGY CHECK
440   NK=130
      GO TO 460
C                 *** NEGATIVE MASS
450   NK=375
460   NR=5
      CALL ERROR
470   WFLAGP=0.
```

```
      WFLAGF=0.
C            *** SHOULD GRID BE REZONED ON THIS CYCLE
      IF ((REZ.NE.0..AND.REZFCT.NE.0..AND.NUMREZ.GT.0).OR.SS4.NE.0.) GO
     1TO 480
      RETURN
480   CALL REZONE
C            *** MUST CALL CDT TO RECALCULATE PRESSURES
      TNOW=T
      DTNOW=DT
      REZ=0.
      SS4=0.
      CALL CDT
      T=TNOW
      DT=DTNOW
      DTNA=DT
      NUMREZ=NUMREZ-1
C
C           *** NREZ = NUMBER OF REZONES ALLOWED (INPUT VALUE OF NUMREZ)
C               NUMREZ = NUMBER OF REZONES ALLOWED MINUS THE NUMBER
C                        OF REZONES PERFORMED  SINCE T=0.
C
      NRZ=NREZ-NUMREZ
C            *** NZ USED IN PRINTOUT OF TOTALS FOR 1-D PROBLEMS
      NZ=4.**NRZ
C
      NUMSPT=NDUMP7
      GO TO 120
C
C                FORMATS
C
490   FORMAT (1H0,17X,35HDEPTH OF CRATER MEASURED FROM JPROJ//12X,1HI,5X
     1,18HJ OF CRATER BOTTOM,12X,1HR,11X,17HDEPTH IN CM. D(I)//)
495   FORMAT (I13,9X,0PF6.1,13X,1PE10.4,9X,1PE10.4)
500   FORMAT (//6X,13HCRATER VOLUME,11X,43HCRATER VOLUME BASED ON (2/3)
     1* PI * D(1)**3/7X,1PE10.4,26X,1PE10.4)
510   FORMAT (5(I4,I4,1P2E9.2))
520   FORMAT (I4,1X,1P2E14.6,3E15.6,E14.6,E15.6,E14.6)
530   FORMAT(8H1PROBLEM,6X,4HTIME,8X,5HCYCLE,3X,13HTOT.EN.THEOR.3X,
     1       19HMAX.REL.ERROR-CYCLE,3X,18HIE SET TO ZERO-PH1,3X,
     2        18HIE SET TO ZERO-PH2,3X,12HPLASTIC-WORK/1F8.4,2X,1PE13.7,
     3        3X,I4,4X,1PE13.7,3X,1PE13.7,1X,I4,6X,1PE13.7,8X,1PE13.7,6X,
     4        1PE13.7/)
540   FORMAT (18X,2HIE,14X,2HKE,7X,13HTOT.EN. (SUM),7X,4HMASS,12X,2HMV,8
     1X,12HMV(POSITIVE),8X,2HMU,8X,12HMU(POSITIVE)/11H J.GT.JPROJ,1P8E15
     2.7/11H J.LE.JPROJ,1P8E15.7/14X,12H------------,3X,12H------------,
     33X,12H------------,3X,12H------------,3X,12H------------,3X,12H---
     4---------,3X,12H------------,3X,12H------------,3X/7H TOTALS,4X,1P
     58E15.7///9H BOUNDARY,9X,6HBOTTOM,9X,5HRIGHT,10X,3HTOP,8X,12H$EVAPO
     6RATED$//9H MASS OUT,2X,1P4E15.7/11H ENERGY OUT,1P4E15.7/7H MU OUT,
     74X,1P4E15.7/7H MV OUT,4X,1P4E15.7//11H WORK DONE ,1P3E15.7//)
550   FORMAT (1H0//21H TAPE 7 DUMP ON CYCLEI5////)
560   FORMAT (1H ///4H I =I3,6X,6HR(I) =F12.3,6X,7HDR(I) =E14.7//3H  J8X
     1,1HU13X,1HV13X,3H P 12X,3HAMX12X,3HRHO11X,3HAIX12X,4HCOMP11X,2H Z/
     2)
570   FORMAT (1H0)
580   FORMAT (//22H J OF PRESSURE-MAXIMUM/(25I5))
590   FORMAT(//103H TRACER POINTS - INITIAL LOCATION IN CELL COORDINATES
```

```
1 (I,J) - CURRENT LOCATION IN CM. COORDINATES (X,Y)// 5(4H    I,3X,     4612
21HJ,5X,1HX,8X,1HY,3X))                                                 4614
 END                                                                    4620-
```

```
      SUBROUTINE MAP                                                    MAP  10
C     ..................................................................MAP  20
C                                                                       MAP  30
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,      MAP  40
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,                MAP  50
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),   MAP  60
     3          GAMC(102),                                             MAP  70
     4          PK(15),   Z(150)   ,                                   MAP  80
     5          XP(26,51),YP(26,51),                                   MAP  90
     6          PL(204)  ,UL(204)  ,PR(204)  ,                         MAP 100
     7           RSN(52),   RST(52),                                   MAP 110
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,               MAP 120
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,               MAP 130
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)      MAP 140
C            *** DIMENSIONED ARRAYS                                     MAP 150
C            *** Z-BLOCK IS SAVED ON TAPE.                              MAP 160
      COMMON      Z                                                     MAP 170
      COMMON     PK                                                     MAP 180
      COMMON   YY,      XX                                              MAP 190
      COMMON   DDX,     DDY                                             MAP 200
      COMMON   AMX,     AIX,     U,     V,     P                       MAP 210
      COMMON   TAU,     JPM                                             MAP 220
      COMMON   UL ,     PL                                              MAP 230
      COMMON   XP ,     YP,      CMXP, CMYP                             MAP 240
C            *** NON-DIMENSIONED VARIABLES                              MAP 250
      COMMON          AID    ,AMMV   ,AMMY   ,AMPY   ,AMUR   ,AMUT   ,AMVR   ,  MAP 260
     1AMVT   ,DELEB ,DELER ,DELET ,DELM   ,DTODX ,DXYMIN,EAMMP ,EAMPY ,  MAP 270
     2E      ,ERDUMP,I     ,I3    ,IWS    ,J      ,K      ,KA     ,KB     ,  MAP 280
     3LL     ,MD    ,ME    ,MZT   ,NERR   ,NK     ,NPRINT,                MAP 290
     4NR     ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR    ,SNT    ,STR    ,SOLID ,  MAP 300
     5SUM    ,TESTRH,TWOPI ,URR   ,WS     ,WSA    ,WSB    ,WSC    ,WFLAGF,  MAP 310
     6WFLAGL,WFLAGP                                                     MAP 320
C                                                                       MAP 330
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE              MAP 340
C                X(0), Y(0), DX(0), DY(0)                               MAP 350
C                                                                       MAP 360
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))                         MAP 370
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))                     MAP 380
C                                                                       MAP 390
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY                      MAP 400
C                                                                       MAP 410
      EQUIVALENCE          (UL,FLEFT),          (UL(103),YAMC),         MAP 420
     1                     (PL,GAMC,PR),        (PL(103),SIGC)          MAP 430
C                                                                       MAP 440
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY                      MAP 450
C                                                                       MAP 460
      EQUIVALENCE          (UL,RSN),                                   MAP 470
     1                     (PL,RST),            (P,UK),                MAP 480
     2                     (P(157),VK),         (P(313),SNB),          MAP 490
     3                     (P(365),STB),        (P(417),RHO)           MAP 500
C                                                                       MAP 510
C            *** SPECIAL EQUIVALENCES FOR EDIT                          MAP 520
C                                                                       MAP 530
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)                             MAP 540
C                                                                       MAP 550
C            *** Z-STORAGE EQUIVALENCES                                 MAP 560
C                                                                       MAP 570
      EQUIVALENCE                          (Z(  1),PROB  ),(Z(  2),CYCLE ), MAP 580
```

```
     1(Z(  3),DT     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  MAP 590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ), (Z(  9),TOPMU ),(Z( 10),RTMU  ), MAP 600
     3(Z( 11),STK1  ),(Z( 12),NUMREZ), (Z( 13),ETH   ),(Z( 14),UN14  ), MAP 610
     4(Z( 15),RHINIT),(Z( 16),PROJI ), (Z( 17),UN17  ),(Z( 18),XMAX  ), MAP 620
     5(Z( 19),NZ    ),(Z( 20),NREZ  ), (Z( 21),AMDM  ),(Z( 22),UVMAX ), MAP 630
     6(Z( 23),UN23  ),(Z( 24),DMIN  ), (Z( 25),JSTR  ),(Z( 26),DTNA  ), MAP 640
     7(Z( 27),CVIS  ),(Z( 28),STK2  ), (Z( 29),STEZ  ),(Z( 30),NC    ), MAP 650
     8(Z( 31),UN31  ),(Z( 32),NRC   ), (Z( 33),IMAX  ),(Z( 34),IMAXA ), MAP 660
     9(Z( 35),JMAX  ),(Z( 36),JMAXA ), (Z( 37),KMAX  ),(Z( 38),KMAXA )  MAP 670
      EQUIVALENCE                                                       MAP 680
     1(Z( 39),BOTM  ),(Z( 40),BOTMV ), (Z( 41),NUMSPT),(Z( 42),CZERO ), MAP 690
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ), (Z( 45),PRDELT),(Z( 46),PRFACT)  MAP 700
      EQUIVALENCE                                                       MAP 710
     1(Z( 47),I1    ),(Z( 48),I2    ), (Z( 49),IFCYCL),(Z( 50),TSTOP ), MAP 720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ), (Z( 53),N3    ),(Z( 54),IVARDY), MAP 730
     3(Z( 55),VT    ),(Z( 56),N6    ), (Z( 57),RTM   ),(Z( 58),RTMV  ), MAP 740
     4(Z( 59),UN59  ),(Z( 60),N10   ), (Z( 61),N11   ),(Z( 62),GAMMA ), MAP 750
     5(Z( 63),TOPM  ),(Z( 64),BOTMU ), (Z( 65),SN    ),(Z( 66),TOPMV ), MAP 760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP), (Z( 69),PRXRT ),(Z( 70),CYCPH3), MAP 770
     7(Z( 71),REZFCT),(Z( 72),TARGI ), (Z( 73),PROJU ),(Z( 74),BBOUND), MAP 780
     8(Z( 75),EVAP  ),(Z( 76),ECK   ), (Z( 77),NECYCL),(Z( 78),II    ), MAP 790
     9(Z( 79),JJ    ),(Z( 80),NMP   ), (Z( 81),Y2    ),(Z( 82),EZPH1 )  MAP 800
      EQUIVALENCE                                                       MAP 810
     1(Z( 83   VARDX),(Z( 84),T     ), (Z( 85),NMPMAX),(Z( 86),PMIN  ), MAP 820
     2(Z( 87  NTER ),(Z( 88),TAYBOT), (Z( 89),TAYTOP),(Z( 90),IEMAP ), MAP 830
     3(Z( 91),MC    ),(Z( 92),MR    ), (Z( 93),MZ    ),(Z( 94),MB    )  MAP 840
      EQUIVALENCE                                                       MAP 850
     1(Z( 95),REZ   ),(Z( 96),NODUMP), (Z( 97),UN97  ),(Z( 98),UN98  ), MAP 860
     2(Z( 99),UN99  ),(Z(100),EVAPM ), (Z(101),EVAPEN),(Z(102),EVAPMU), MAP 870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ), (Z(105),SNL   ),(Z(106),STL   ), MAP 880
     4(Z(107),TAXRT ),(Z(108),IDNMAP), (Z(109),IPRMAP),(Z(110),ROEPS ), MAP 890
     5(Z(111),RHINI ),(Z(112),VINI  ), (Z(113),FINAL ),(Z(114),IVMAP ), MAP 900
     6(Z(115),RHOZ  ),(Z(116),ESA   ), (Z(117),ESEZ  ),(Z(118),ESB   ), MAP 910
     7(Z(119),ESCAPA),(Z(120),ESESP ), (Z(121),ESESQ ),(Z(122),ESES  ), MAP 920
     8(Z(123),ESALPH),(Z(124),ESBETA), (Z(125),ESCAPB),(Z(126),IUMAP ), MAP 930
     9(Z(127),SS1   ),(Z(128),SS2   ), (Z(129),UMIN  ),(Z(130),SS4   )  MAP 940
      EQUIVALENCE                                                       MAP 950
     1(Z(131),PRTIME),(Z(132),EOR   ), (Z(133),EOT   ),(Z(134),EOB   ), MAP 960
     2(Z(135),EMOR  ),(Z(136),DXF   ), (Z(137),DYF   ),(Z(138),RHOMIN), MAP 970
     3(Z(139),STAB), (Z(140),XIENRG),  (Z(141),XKENRG), (Z(142),XTENRG),MAP 980
     4(Z(143),STT   ),(Z(144),DTMIN ), (Z(145),TRNSFC),(Z(146),EMOT  ), MAP 990
     5(Z(147),JPROJ ),(Z(148),CNAUT ), (Z(149),BBAR  ),(Z(150),EMOB  )  MAP1000
C                                                                       MAP1010
C     ..................................................................MAP1020
C                                                                       MAP1030
C     END OF COMMON                                                     MAP1040
C                                                                       MAP1050
C     ..................................................................MAP1060
C                                                                       MAP1070
      DIMENSION VALUE(40)                                               MAP1080
      DIMENSION ALE(41)                                                 MAP1090
      DATA ALE/2H  ,2H .,2H -,2H A,2H B,2H C,2H D,2H E,2H F,            MAP1100
     1         2H G,2H H,2H I,2H J,2H K,2H L,2H M,2H N,2H O,            MAP1110
     2         2H P,2H Q,2H R,2H S,2H T,2H U,2H V,2H W,2H X,            MAP1120
     3         2H Y,2H Z,2H +,2H *,2H 1,2H 2,2H 3,2H 4,2H 5,            MAP1130
     4         2H 6,2H 7,2H 8,2H 9,2H 0/                                MAP1140
      DIMENSION XUM(41)                                                 MAP1150
```

```
      DATA XUM/2H  ,2H-.,2H--,2H-A,2H-B,2H-C,2H-D,2H-E,2H-F,
     1         2H-G,2H-H,2H-I,2H-J,2H-K,2H-L,2H-M,2H-N,2H-O,
     2         2H-P,2H-Q,2H-R,2H-S,2H-T,2H-U,2H-V,2H-W,2H-X,
     3         2H-Y,2H-Z,2H-+,2H-*,2H-1,2H-2,2H-3,2H-4,2H-5,
     4         2H-6,2H-7,2H-8,2H-9,2H 0/
C
C            *** SEARCH FOR MINIMUM AND MAXIMUM COMPRESSIONS
C                TO SCALE COMPRESSION MAP
C
      IDL=I1
      JDL=I2
      IF (NC.NE.0) GO TO 10
      IDL=MIN0(IMAX,55)
      JDL=JMAX
10    WSMAX=0.
      WSMIN=10.
C
      DO 20 J=1,JDL
      K=(J-1)*IMAX+1
      DO 20 I=1,IDL
      K=K+1
      WSA=AMX(K)/(TAU(I)*DY(J)*RHOZ)
      IF (WSA.EQ.0.) GO TO 20
      WSMAX=AMAX1(WSMAX,WSA)
      WSMIN=AMIN1(WSMIN,WSA)
20    CONTINUE
      IF (WSMIN.LT.WSMAX) GO TO 30
      WSMIN=0.
C
C            *** DEFINE LINEAR SCALE FACTOR  AND  PRINT KEY TO
C                COMPRESSION MAP.
30    DSCALE=(WSMAX-WSMIN)/FLOAT(IDNMAP)
      IF ((AINT(DSCALE*100.)).LT.(DSCALE*100.)) GO TO 50
      DSCALE=AINT(DSCALE*100.)/100.
      GO TO 60
50    DSCALE=AINT(DSCALE*100.+1.)/100.
60    CONTINUE
      DO 70 I=1,IDNMAP
70    VALUE(I)=WSMIN+FLOAT(I)*DSCALE
      WRITE (6,860)
      ILIM1=1
      ILIM2=20
80    IF (IDNMAP.LT.ILIM2) ILIM2=IDNMAP
      WRITE (6,870) (ALE(I+1),I=ILIM1,ILIM2)
      WRITE (6,960) (VALUE(I),I=ILIM1,ILIM2)
      IF (IDNMAP.EQ.ILIM2) GO TO 90
      ILIM1=ILIM2+1
      ILIM2=ILIM2+20
      GO TO 80
90    CONTINUE

      WRITE (6,980)
C
      J=JDL
100   K=(J-1)*IMAX+1
C
      DO 150 I=1,IDL
```

```
      K=K+1                                                             MAP1760
      IF (AMX(K).GT.0.) GO TO 110                                       MAP1770
      GO TO 130                                                         MAP1780
110   WSA=AMX(K)/(TAU(I)*DY(J)  HOZ)                                    MAP1790
      IF (WSA.GT.WSMIN) GO TO 120                                       MAP1800
C            *** PRINT A DOT TO REPRESENT SMALLEST COMPRESSION.         MAP1805
      MA=2                                                              MAP1810
      GO TO 140                                                         MAP1820
120   TMA=(WSA-WSMIN)/DSCALE+1.                                         MAP1830
      MA=TMA                                                            MAP1840
      IF (WSA.GT.FLOAT(MA-1)*DSCALE+WSMIN) MA=MA+1                      MAP1850
      GO TO 140                                                         MAP1860
C            *** PRINT A BLANK FOR EMPTY CELLS.                         MAP1865
130   MA=1                                                              MAP1870
140   PR(I)=ALE(MA)                                                     MAP1880
150   CONTINUE                                                          MAP1890
C            *** PRINT J-VALUE ALONG Y-AXIS WHEN IT IS A MULTIPLE OF 5. MAP1895
      IF (MOD(J,5).NE.0) GO TO 160                                      MAP1900
      WRITE (6,880) J,(PR(I),I=1,IDL)                                   MAP1910
      GO TO 170                                                         MAP1920
160   WRITE (6,890) (PR(I),I=1,IDL)                                     MAP1930
170   J=J-1                                                             MAP1940
      IF (J.GT.0) GO TO 100                                             MAP1950
C            *** PRINT AND LABEL X-AXIS OF MAP.                         MAP1960
      PR(1)=ALE(30)                                                     MAP1970
      WRITE (6,880) J,(PR(1),I=1,IDL)                                   MAP1980
      WRITE (6,900) (I,I=0,IDL,5)                                       MAP1990
C                                                                       MAP2000
C            *** SEARCH FOR MINIMUM AND MAXIMUM PRESSURES               MAP2010
C                TO SCALE PRESSURE MAP                                  MAP2020
C                                                                       MAP2030
      WSMAX=0.                                                          MAP2040
C                                                                       MAP2050
      DO 180 J=1,JDL                                                    MAP2060
      DO 180 I=1,IDL                                                    MAP2070
      K=(J-1)*IMAX+I+1                                                  MAP2080
      WSA=ABS(P(K))                                                     MAP2090
      IF (WSA.EQ.0.) GO TO 180                                          MAP2100
      WSMAX=AMAX1(WSMAX,WSA)                                            MAP2110
180   CONTINUE                                                          MAP2120
      WSMIN=10.*PMIN                                                    MAP2130
      WRITE (6,910)                                                     MAP2140
C            *** PRINT KEY TO MAP ONLY IF THERE ARE NON-ZERO PRESSURES. MAP2145
      IF (WSMAX.NE.0.) GO TO 190                                        MAP2150
      J=JDL                                                             MAP2160
      GO TO 260                                                         MAP2170
C                                                                       MAP2180
C            *** DEFINE LOGARITHMIC SCALE FACTOR AND PRINT KEY TO       MAP2220
C                PRESSURE MAP.                                          MAP2230
190   MAXEXP=INT(ALOG10(WSMAX))                                         MAP2240
      MINEXP=INT(ALOG10(WSMIN))                                         MAP2250
      PSCLE=FLOAT(MAXEXP-MINEXP+1)/FLOAT(IPRMAP)                        MAP2260
      IF ((AINT(PSCLE*1000.)).LT.(PSCLE*1000.)) GO TO 210               MAP2270
      PSCLE=AINT(PSCLE*1000.)/1000.                                     MAP2280
      GO TO 220                                                         MAP2290
210   PSCLE=AINT(PSCLE*1000.+1.)/1000.                                  MAP2300
220   CONTINUE                                                          MAP2310
```

```
      DO 230 I=1,IPRMAP                                                 MAP2320
230   VALUE(I)=10.**(MINEXP+FLOAT(I)*PSCLE)                             MAP2330
      ILIM1=1                                                           MAP2340
      ILIM2=10                                                          MAP2350
240   IF (IPRMAP.LT.ILIM2) ILIM2=IPRMAP                                 MAP2360
      WRITE (6,920) (ALE(I+3),I=ILIM1,ILIM2)                            MAP2370
      WRITE (6,970) (VALUE(I),I=ILIM1,ILIM2)                            MAP2380
      IF (IPRMAP.EQ.ILIM2) GO TO 250                                    MAP2390
      ILIM1=ILIM2+1                                                     MAP2400
      ILIM2=ILIM2+10                                                    MAP2410
      GO TO 240                                                         MAP2420
250   CONTINUE                                                          MAP2430
      WRITE (6,980)                                                     MAP2440
C                                                                       MAP2450
      J=JDL                                                             MAP2460
260   K=(J-1)*IMAX+1                                                    MAP2470
C                                                                       MAP2480
      DO 320 I=1,IDL                                                    MAP2490
      K=K+1                                                             MAP2500
      IF (AMX(K).GT.0.) GO TO 270                                       MAP2510
C            *** PRINT A BLANK FOR EMPTY CELLS.                         MAP2515
      MA=1                                                              MAP2520
      GO TO 310                                                         MAP2530
270   IF (P(K).NE.0.) GO TO 280                                         MAP2540
C            *** PRINT A ZERO FOR NONEMPTY CELLS WITH ZERO PRESSURE.    MAP2545
      MA=41                                                             MAP2550
      GO TO 310                                                         MAP2560
280   FLOTMA=(ALOG10(ABS(P(K)))-FLOAT(MINEXP))/PSCLE+3.                 MAP2570
      INTMA=INT(FLOTMA)                                                 MAP2580
      IF (FLOTMA.GT.FLOAT(INTMA)) GO TO 290                             MAP2590
      MA=INTMA                                                          MAP2600
      GO TO 300                                                         MAP2610
290   MA=INTMA+1                                                        MAP2620
C            *** DO NOT USE DOT AND DASH IN PRESSURE MAP.               MAP2625
300   IF (MA.LE.3) MA=4                                                 MAP2630
310   CONTINUE                                                          MAP2640
      PR(I)=ALE(MA)                                                     MAP2650
C            *** USE XUM ARRAY OF SYMBOLS FOR NEGATIVE PRESSURES.       MAP2655
320   IF (P(K).LT.0.) PR(I)=XUM(MA)                                     MAP2660
C                                                                       MAP2670
      IF (MOD(J,5).NE.0) GO TO 330                                      MAP2680
      WRITE (6,880) J,(PR(I),I=1,IDL)                                   MAP2690
      GO TO 340                                                         MAP2700
330   WRITE (6,890) (PR(I),I=1,IDL)                                     MAP2710
340   J=J-1                                                             MAP2720
      IF (J.GT.0) GO TO 260                                             MAP2730
C            *** PRINT AND LABEL X-AXIS OF MAP.                         MAP2740
      PR(1)=ALE(30)                                                     MAP2750
      WRITE (6,880) J,(PR(1),I=1,IDL)                                   MAP2760
      WRITE (6,900) (I,I=0,IDL,5)                                       MAP2770
C                                                                       MAP2780
C            *** SEARCH FOR MINIMUM AND MAXIMUM RADIAL                  MAP2790
C                VELOCITIES TO DEFINE SCALE FACTOR OF                   MAP2800
C                RADIAL VELOCITY MAP                                    MAP2810
C                                                                       MAP2820
      WSMAX=0.                                                          MAP2830
C                                                                       MAP2840
```

```
      DO 350 J=1,JDL                                                    MAP2850
      DO 350 I=1,IDL                                                    MAP2860
      K=(J-1)*IMAX+I+1                                                  MAP2870
      WSA=ABS(U(K))                                                     MAP2880
      IF (WSA.EQ.0.) GO TO 350                                          MAP2890
      WSMAX=AMAX1(WSMAX,WSA)                                            MAP2900
350   CONTINUE                                                          MAP2910
      WSMIN=10.*UMIN                                                    MAP2920
      WRITE (6,930)                                                     MAP2930
C            *** PRINT KEY TO MAP ONLY IF THERE ARE NON-ZERO VALUES.    MAP2935
      IF (WSMAX.NE.0.) GO TO 360                                        MAP2940
      J=JDL                                                             MAP2950
      GO TO 430                                                         MAP2960
C                                                                       MAP2970
C                                                                       MAP3020
C            *** USCLE IS LOGARITHMIC SCALE FACTOR OF RADIAL VELOCITY   MAP3030
C                MAP.                                                   MAP3040
360   MAXEXP=INT(ALOG10(WSMAX))                                         MAP3050
      MINEXP=INT(ALOG10(WSMIN))                                         MAP3060
      USCLE=FLOAT(MAXEXP-MINEXP+1)/FLOAT(IUMAP)                         MAP3070
      IF ((AINT(USCLE*1000.)).LT.(USCLE*1000.)) GO TO 380               MAP3080
      USCLE=AINT(USCLE*1000.)/1000.                                     MAP3090
      GO TO 390                                                         MAP3100
380   USCLE=AINT(USCLE*1000.+1.)/1000.                                  MAP3110
390   CONTINUE                                                          MAP3120
C            *** PRINT KEY TO MAP.                                      MAP3125
      DO 400 I=1,IUMAP                                                  MAP3130
400   VALUE(I)=10.**(MINEXP+FLOAT(I)*USCLE)                             MAP3140
      ILIM1=1                                                           MAP3150
      ILIM2=10                                                          MAP3160
410   IF (IUMAP.LT.ILIM2) ILIM2=IUMAP                                   MAP3170
      WRITE (6,920) (ALE(I+3),I=ILIM1,ILIM2)                            MAP3180
      WRITE (6,970) (VALUE(I),I=ILIM1,ILIM2)                            MAP3190
      IF (IUMAP.EQ.ILIM2) GO TO 420                                     MAP3200
      ILIM1=ILIM2+1                                                     MAP3210
      ILIM2=ILIM2+10                                                    MAP3220
      GO TO 410                                                         MAP3230
420   CONTINUE                                                          MAP3240
      WRITE (6,980)                                                     MAP3250
C                                                                       MAP3260
      J=JDL                                                             MAP3270
430   K=(J-1)*IMAX+1                                                    MAP3280
C                                                                       MAP3290
      DO 490 I=1,IDL                                                    MAP3300
      K=K+1                                                             MAP3310
      IF (AMX(K).GT.0.) GO TO 440                                       MAP3320
C            *** EMPTY CELL.                                            MAP3325
      MA=1                                                              MAP3330
      GO TO 480                                                         MAP3340
440   IF (U(K).NE.0.) GO TO 450                                         MAP3350
C            *** ZERO RADIAL VELOCITY.                                  MAP3355
      MA=41                                                             MAP3360
      GO TO 480                                                         MAP3370
450   FLOTMA=(ALOG10(ABS(U(K)))-FLOAT(MINEXP))/USCLE+3.                 MAP3380
      INTMA=INT(FLOTMA)                                                 MAP3390
      IF (FLOTMA.GT.FLOAT(INTMA)) GO TO 460                             MAP3400
      MA=INTMA                                                          MAP3410
```

```
      GO TO 470                                                         MAP3420
400   MA=INTMA+1                                                        MAP3430
C           *** DO NOT USE DOT OR DASH IN RADIAL VELOCITY MAP.          MAP3435
470   IF (MA.LE.3) MA=4                                                 MAP3440
480   CONTINUE                                                          MAP3450
      PR(I)=ALE(MA)                                                     MAP3460
C           *** USE XUM ARRAY FOR NEGATIVE RADIAL VELOCITIES.           MAP3465
490   IF (U(K).LT.0.) PR(I)=XUM(MA)                                     MAP3470
C           *** PRINT J-VALUE ALONG Y-AXIS WHEN IT IS A MULTIPLE        MAP3480
C               OF 5.                                                   MAP3485
      IF (MOD(J,5).NE.0) GO TO 500                                      MAP3490
      WRITE (6,880) J,(PR(I),I=1,IDL)                                   MAP3500
      GO TO 510                                                         MAP3510
500   WRITE (6,890) (PR(I),I=1,IDL)                                     MAP3520
510   J=J-1                                                             MAP3530
      IF (J.GT.0) GO TO 430                                             MAP3540
C           *** PRINT AND LABEL X-AXIS OF MAP.                          MAP3550
      PR(1)=ALE(30)                                                     MAP3560
      WRITE (6,880) J,(PR(1),I=1,IDL)                                   MAP3570
      WRITE (6,900) (I,I=0,IDL,5)                                       MAP3580
C                                                                       MAP3590
C           *** SEARCH FOR MINIMUM AND MAXIMUM AXIAL                    MAP3600
C               VELOCITES TO DEFINE SCALE FACTOR OF                     MAP3610
C               AXIAL VELOCITY MAP                                      MAP3620
C                                                                       MAP3630
      WSMAX=0.                                                          MAP3640
C                                                                       MAP3650
      DO 520 J=1,JDL                                                    MAP3660
      DO 520 I=1,IDL                                                    MAP3670
      K=(J-1)*IMAX+I+1                                                  MAP3680
      WSA=ABS(V(K))                                                     MAP3690
      IF (WSA.EQ.0.) GO TO 520                                          MAP3700
      WSMAX=AMAX1(WSMAX,WSA)                                            MAP3710
520   CONTINUE                                                          MAP3720
      WSMIN=10.*UMIN                                                    MAP3730
      WRITE (6,940)                                                     MAP3740
C           *** PRINT KEY TO MAP ONLY IF THERE ARE NON-ZERO VALUES.     MAP3745
      IF (WSMAX.NE.0.) GO TO 530                                        MAP3750
      J=JDL                                                             MAP3760
      GO TO 600                                                         MAP3770
C                                                                       MAP3780
C                                                                       MAP3820
C           *** VSCLE IS LOGARITHMIC SCALE FACTOR FOR AXIAL VELOCITY    MAP3830
C               MAP.                                                    MAP3840
530   MAXEXP=INT(ALOG10(WSMAX))                                         MAP3850
      MINEXP=INT(ALOG10(WSMIN))                                         MAP3860
      VSCLE=FLOAT(MAXEXP-MINEXP+1)/FLOAT(IVMAP)                         MAP3870
      IF ((AINT(VSCLE*1000.)).LT.(VSCLE*1000.)) GO TO 550               MAP3880
      VSCLE=AINT(VSCLE*1000.)/1000.                                     MAP3890
      GO TO 560                                                         MAP3900
550   VSCLE=AINT(VSCLE*1000.+1.)/1000.                                  MAP3910
560   CONTINUE                                                          MAP3920
C           *** PRINT KEY TO AXIAL VELOCITY MAP.                        MAP3925
      DO 570 I=1,IVMAP                                                  MAP3930
570   VALUE(I)=10.**(MINEXP+FLOAT(I)*VSCLE)                             MAP3940
      ILIM1=1                                                           MAP3950
      ILIM2=10                                                          MAP3960
```

```
580   IF (IVMAP.LT.ILIM2) ILIM2=IVMAP                                  MAP3970
      WRITE (6,920) (ALE(I+3),I=ILIM1,ILIM2)                           MAP3980
      WRITE (6,970) (VALUE(I),I=ILIM1,ILIM2)                           MAP3990
      IF (IVMAP.EQ.ILIM2) GO TO 590                                    MAP4000
      ILIM1=ILIM2+1                                                    MAP4010
      ILIM2=ILIM2+10                                                   MAP4020
      GO TO 580                                                        MAP4030
590   CONTINUE                                                         MAP4040
      WRITE (6,980)                                                    MAP4050
C                                                                      MAP4060
      J=JDL                                                            MAP4070
600   K=(J-1)*IMAX+1                                                   MAP4080
C                                                                      MAP4090
      DO 660 I=1,IDL                                                   MAP4100
      K=K+1                                                            MAP4110
      IF (AMX(K).GT.0.) GO TO 610                                      MAP4120
C            *** EMPTY CELL.                                           MAP4125
      MA=1                                                             MAP4130
      GO TO 650                                                        MAP4140
610   IF (V(K).NE.0.) GO TO 620                                        MAP4150
C            *** ZERO AXIAL VELOCITY.                                  MAP4155
      MA=41                                                            MAP4160
      GO TO 650                                                        MAP4170
620   FLOTMA=(ALOG10(ABS(V(K)))-FLOAT(MINEXP))/VSCLE+3.                MAP4180
      INTMA=INT(FLOTMA)                                                MAP4190
      IF (FLOTMA.GT.FLOAT(INTMA)) GO TO 630                            MAP4200
      MA=INTMA                                                         MAP4210
      GO TO 640                                                        MAP4220
630   MA=INTMA+1                                                       MAP4230
C            *** DO NOT USE DOT OR DASH IN AXIAL VELOCITY MAP.         MAP4235
640   IF (MA.LE.3) MA=4                                                MAP4240
650   CONTINUE                                                         MAP4250
      PR(I)=ALE(MA)                                                    MAP4260
C            *** USE XUM ARRAY FOR NEGATIVE AXIAL VELOCITIES.          MAP4265
660   IF (V(K).LT.0.) PR(I)=XUM(MA)                                    MAP4270
C            *** PRINT J-VALUE ALONG Y-AXIS WHEN IT IS A MULTIPLE OF 5. MAP4280
      IF (MOD(J,5).NE.0) GO TO 670                                     MAP4290
      WRITE (6,880) J,(PR(I),I=1,IDL)                                  MAP4300
      GO TO 680                                                        MAP4310
670   WRITE (6,890) (PR(I),I=1,IDL)                                    MAP4320
680   J=J-1                                                            MAP4330
      IF (J.GT.0) GO TO 600                                            MAP4340
C            *** PRINT AND LABEL X-AXIS OF MAP.                        MAP4350
      PR(1)=ALE(30)                                                    MAP4360
      WRITE (6,880) J,(PR(1),I=1,IDL)                                  MAP4370
      WRITE (6,900) (I,I=0,IDL,5)                                      MAP4380
C                                                                      MAP4390
C            *** SEARCH FOR MINIMUM AND MAXIMUM  SPECIFIC INTERNAL     MAP4400
C                ENERGIES TO DEFINE SCALE FACTOR OF ENERGY MAP         MAP4410
C                                                                      MAP4420
      WSMAX=0.                                                         MAP4430
C                                                                      MAP4440
      DO 690 J=1,JDL                                                   MAP4450
      DO 690 I=1,IDL                                                   MAP4460
      K=(J-1)*IMAX+I+1                                                 MAP4470
      WSA=ABS(AIX(K))                                                  MAP4480
      IF (WSA.EQ.0.) GO TO 690                                         MAP4490
```

```
      WSMAX=AMAX1(WSMAX,WSA)                                            MAP4500
690   CONTINUE                                                          MAP4510
      WSMIN=10.*SIEMIN                                                  MAP4520
      WRITE (6,950)                                                     MAP4530
C            *** PRINT KEY TO MAP ONLY IF THERE ARE NON-ZERO VALUES.    MAP4535
      IF (WSMAX.NE.0.) GO TO 700                                        MAP4540
      J=JDL                                                             MAP4550
      GO TO 770                                                         MAP4560
C                                                                       MAP4570
C                                                                       MAP4610
C            *** ESCLE IS LOGARITHMIC SCALE FACTOR FOR INTERNAL ENERGY  MAP4620
C                MAP.                                                   MAP4630
700   MAXEXP=INT(ALOG10(WSMAX))                                         MAP4640
      MINEXP=INT(ALOG10(WSMIN))                                         MAP4650
      ESCLE=FLOAT(MAXEXP-MINEXP+1)/FLOAT(IEMAP)                         MAP4660
      IF ((AINT(ESCLE*1000.)).LT.(ESCLE*1000.)) GO TO 720               MAP4670
      ESCLE=AINT(ESCLE*1000.)/1000.                                     MAP4680
      GO TO 730                                                         MAP4690
720   ESCLE=AINT(ESCLE*1000.+1.)/1000.                                  MAP4700
C            *** PRINT KEY TO INTERNAL ENERGY MAP.                      MAP4705
730   CONTINUE                                                          MAP4710
      DO 740 I=1,IEMAP                                                  MAP4720
740   VALUE(I)=10.**(MINEXP+FLOAT(I)*ESCLE)                             MAP4730
      ILIM1=1                                                           MAP4740
      ILIM2=10                                                          MAP4750
750   IF (IEMAP.LT.ILIM2) ILIM2=IEMAP                                   MAP4760
      WRITE (6,920) (ALE(I+3),I=ILIM1,ILIM2)                            MAP4770
      WRITE (6,970) (VALUE(I),I=ILIM1,ILIM2)                            MAP4780
      IF (IEMAP.EQ.ILIM2) GO TO 760                                     MAP4790
      ILIM1=ILIM2+1                                                     MAP4800
      ILIM2=ILIM2+10                                                    MAP4810
      GO TO 750                                                         MAP4820
760   CONTINUE                                                          MAP4830
      WRITE (6,980)                                                     MAP4840
C                                                                       MAP4850
      J=JDL                                                             MAP4860
770   K=(J-1)*IMAX+1                                                    MAP4870
C                                                                       MAP4880
      DO 830 I=1,IDL                                                    MAP4890
      K=K+1                                                             MAP4900
      IF (AMX(K).GT.0.) GO TO 780                                       MAP4910
C            *** EMPTY CELL.                                            MAP4915
      MA=1                                                              MAP4920
      GO TO 820                                                         MAP4930
780   IF (AIX(K).NE.0.) GO TO 790                                       MAP4940
C            *** ZERO INTERNAL ENERGY.                                  MAP4945
      MA=41                                                             MAP4950
      GO TO 820                                                         MAP4960
790   FLOTMA=(ALOG10(ABS(AIX(K)))-FLOAT(MINEXP))/ESCLE+3.               MAP4970
      INTMA=INT(FLOTMA)                                                 MAP4980
      IF (FLOTMA.GT.FLOAT(INTMA)) GO TO 800                             MAP4990
      MA=INTMA                                                          MAP5000
      GO TO 810                                                         MAP5010
800   MA=INTMA+1                                                        MAP5020
C            *** DO NOT USE DOT AND DASH IN INTERNAL ENERGY MAP.        MAP5025
810   IF (MA.LE.3) MA=4                                                 MAP5030
820   CONTINUE                                                          MAP5040
```

```
      PR(I)=ALE(MA)                                                     MAP5050
830   IF (AIX(K).LT.0.) PR(I)=XUM(MA)                                   MAP5060
C           *** PRINT J-VALUE ALONG Y-AXIS WHEN IT IS A MULTIPLE OF 5. MAP5070
      IF (MOD(J,5).NE.0) GO TO 840                                      MAP5080
      WRITE (6,880) J,(PR(I),I=1,IDL)                                   MAP5090
      GO TO 850                                                         MAP5100
840   WRITE (6,890) (PR(I),I=1,IDL)                                     MAP5110
850   J=J-1                                                             MAP5120
      IF (J.GT.0) GO TO 770                                             MAP5130
C           *** PRINT AND LABEL X-AXIS OF MAP.                          MAP5140
      PR(1)=ALE(30)                                                     MAP5150
      WRITE (6,880) J,(PR(1),I=1,IDL)                                   MAP5160
      WRITE (6,900) (I,I=0,IDL,5)                                       MAP5170
C                                                                       MAP5180
C                                                                       MAP5190
      RETURN                                                            MAP5200
C                                                                       MAP5210
C           *** FORMATS                                                 MAP5215
860   FORMAT (1H1,4X,15HCOMPRESSION    //)                             MAP5220
870   FORMAT (16H    SYMBOL       ,20(3X,A2))                          MAP5230
880   FORMAT (I10,2H I,54A2)                                            MAP5240
890   FORMAT (10X,2H I,54A2)                                            MAP5250
900   FORMAT (I12,10I10////)                                            MAP5260
910   FORMAT (1H1,4X,15HPRESSURE       //)                             MAP5270
920   FORMAT (16H    SYMBOL       ,10(3X,A2,5X))                       MAP5280
930   FORMAT (1H1,4X,15HRADIAL VELOCITY//)                              MAP5290
940   FORMAT (1H1,4X,15HAXIAL VELOCITY //)                              MAP5300
950   FORMAT (1H1,4X,15HINTERNAL ENERGY//)                              MAP5310
960   FORMAT (16H  MAXIMUM VALUE ,20(F5.2))                             MAP5320
970   FORMAT (16H  MAXIMUM VALUE ,1P10E10.2)                            MAP5330
980   FORMAT (//)                                                       MAP5340
      END                                                               MAP5350
```

```
      SUBROUTINE PH1                                                    PH1  10
C     ..................................................................PH1  20
C                                                                       PH1  30
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,      PH1  40
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,                PH1  50
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),   PH1  60
     3          GAMC(102),                                             PH1  70
     4          PK(15),   Z(150)  ,                                    PH1  80
     5          XP(26,51),YP(26,51),                                   PH1  90
     6          PL(204)  ,UL(204)  ,PR(204)  ,                         PH1 100
     7           RSN(52),   RST(52),                                   PH1 110
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,               PH1 120
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,               PH1 130
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)      PH1 140
C            *** DIMENSIONED ARRAYS                                     PH1 150
C            *** Z-BLOCK IS SAVED ON TAPE.                              PH1 160
      COMMON     Z                                                     PH1 170
      COMMON    PK                                                     PH1 180
      COMMON  YY,     XX                                               PH1 190
      COMMON  DDX,    DDY                                              PH1 200
      COMMON  AMX,    AIX,    U,    V,    P                            PH1 210
      COMMON  TAU,    JPM                                              PH1 220
      COMMON  UL ,    PL                                               PH1 230
      COMMON  XP ,    YP,     CMXP, CMYP                               PH1 240
C            *** NON-DIMENSIONED VARIABLES                              PH1 250
      COMMON        AID   ,AMMV  ,AMMY  ,AMPY  ,AMUR  ,AMUT  ,AMVR  ,  PH1 260
     1AMVT  ,DELEB ,DELER ,DELET ,DELM  ,DTODX ,DXYMIN,EAMMP ,EAMPY ,   PH1 270
     2E     ,ERDUMP,I     ,13    ,IWS   ,J     ,K     ,KA    ,KB    ,   PH1 280
     3LL    ,MD    ,ME    ,MZT   ,NERR  ,NK    ,NPRINT,                PH1 290
     4NR    ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR   ,SNT   ,STR   ,SOLID ,  PH1 300
     5SUM   ,TESTRH,TWOPI ,URR   ,WS    ,WSA   ,WSB   ,WSC   ,WFLAGF,   PH1 310
     6WFLAGL,WFLAGP                                                    PH1 320
C                                                                       PH1 330
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE              PH1 340
C                X(0), Y(0), DX(0), DY(0)                               PH1 350
C                                                                       PH1 360
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))                         PH1 370
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))                     PH1 380
C                                                                       PH1 390
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY                      PH1 400
C                                                                       PH1 410
      EQUIVALENCE          (UL,FLEFT),        (UL(103),YAMC),          PH1 420
     1                     (PL,GAMC,PR),      (PL(103),SIGC)           PH1 430
C                                                                       PH1 440
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY                      PH1 450
C                                                                       PH1 460
      EQUIVALENCE          (UL,RSN),                                   PH1 470
     1                     (PL,RST),          (P,UK),                  PH1 480
     2                     (P(157),VK),       (P(313),SNB),            PH1 490
     3                     (P(365),STB),      (P(417),RHO)             PH1 500
C                                                                       PH1 510
C            *** SPECIAL EQUIVALENCES FOR EDIT                          PH1 520
C                                                                       PH1 530
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)                             PH1 540
C                                                                       PH1 550
C            *** Z-STORAGE EQUIVALENCES                                 PH1 560
C                                                                       PH1 570
      EQUIVALENCE                        (Z(  1),PROB  ),(Z(  2),CYCLE ), PH1 580
```

```
     1(Z(  3),NOT    ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  PH1 590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ), (Z(  9),TOPMU ),(Z( 10),RTMU  ),  PH1 600
     3(Z( 11),STK1  ),(Z( 12),NUMREZ), (Z( 13),ETH   ),(Z( 14),UN14  ),  PH1 610
     4(Z( 15),RHINIT),(Z( 16),PROJI ), (Z( 17),UN17  ),(Z( 18),XMAX  ),  PH1 620
     5(Z( 19),NZ    ),(Z( 20),NREZ  ), (Z( 21),AMDM  ),(Z( 22),UVMAX ),  PH1 630
     6(Z( 23),UN23  ),(Z( 24),DMIN  ), (Z( 25),JSTR  ),(Z( 26),DTNA  ),  PH1 640
     7(Z( 27),CVIS  ),(Z( 28),STK2  ), (Z( 29),STEZ  ),(Z( 30),NC    ),  PH1 650
     8(Z( 31),UN31  ),(Z( 32),NRC   ), (Z( 33),IMAX  ),(Z( 34),IMAXA ),  PH1 660
     9(Z( 35),JMAX  ),(Z( 36),JMAXA ), (Z( 37),KMAX  ),(Z( 38),KMAXA )  PH1 670
      EQUIVALENCE                                                       PH1 680
     1(Z( 39),BOTM  ),(Z( 40),BOTMV ), (Z( 41),NUMSPT),(Z( 42),CZERO ),  PH1 690
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ), (Z( 45),PRDELT),(Z( 46),PRFACT)  PH1 700
      EQUIVALENCE                                                       PH1 710
     1(Z( 47),I1    ),(Z( 48),I2    ), (Z( 49),IPCYCL),(Z( 50),TSTOP ),  PH1 720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ), (Z( 53),N3    ),(Z( 54),IVARDY),  PH1 730
     3(Z( 55),VT    ),(Z( 56),N6    ), (Z( 57),RTM   ),(Z( 58),RTMV  ),  PH1 740
     4(Z( 59),UN59  ),(Z( 60),N10   ), (Z( 61),N11   ),(Z( 62),GAMMA ),  PH1 750
     5(Z( 63),TOPM  ),(Z( 64),BOTMU ), (Z( 65),SN    ),(Z( 66),TOPMV ),  PH1 760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP), (Z( 69),PRXRT ),(Z( 70),CYCPH3),  PH1 770
     7(Z( 71),REZFCT),(Z( 72),TARGI ), (Z( 73),PROJU ),(Z( 74),BBOUND),  PH1 780
     8(Z( 75),EVAP  ),(Z( 76),ECK   ), (Z( 77),NECYCL),(Z( 78),II    ),  PH1 790
     9(Z( 79),JJ    ),(Z( 80),NMP   ), (Z( 81),Y2    ),(Z( 82),EZPH1 )  PH1 800
      EQUIVALENCE                                                       PH1 810
     1(Z( 83),IVARDX),(Z( 84),T     ), (Z( 85),NMPMAX),(Z( 86),PMIN  ),  PH1 820
     2(Z( 87),INTER ),(Z( 88),TAYBOT), (Z( 89),TAYTOP),(Z( 90),IEMAP ),  PH1 830
     3(Z( 91),MC    ),(Z( 92),MR    ), (Z( 93),MZ    ),(Z( 94),MB    )  PH1 840
      EQUIVALENCE                                                       PH1 850
     1(Z( 95),REZ   ),(Z( 96),NODUMP), (Z( 97),UN97  ),(Z( 98),UN98  ),  PH1 860
     2(Z( 99),UN99  ),(Z(100),EVAPM ), (Z(101),EVAPEN),(Z(102),EVAPMU),  PH1 870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ), (Z(105),SNL   ),(Z(106),STL   ),  PH1 880
     4(Z(107),TAXRT ),(Z(108),IDNMAP), (Z(109),IPRMAP),(Z(110),ROEPS ),  PH1 890
     5(Z(111),RHINI ),(Z(112),VINI  ), (Z(113),FINAL ),(Z(114),IVMAP ),  PH1 900
     6(Z(115),RHOZ  ),(Z(116),ESA   ), (Z(117),ESEZ  ),(Z(118),ESB   ),  PH1 910
     7(Z(119),ESCAPA),(Z(120),ESESP ), (Z(121),ESESQ ),(Z(122),ESES  ),  PH1 920
     8(Z(123),ESALPH),(Z(124),ESBETA), (Z(125),ESCAPB),(Z(126),IUMAP ),  PH1 930
     9(Z(127),SS1   ),(Z(128),SS2   ), (Z(129),UMIN  ),(Z(130),SS4   )  PH1 940
      EQUIVALENCE                                                       PH1 950
     1(Z(131),PRTIME),(Z(132),EOR   ), (Z(133),EOT   ),(Z(134),EOB   ),  PH1 960
     2(Z(135),EMOR  ),(Z(136),DXF   ), (Z(137),DYF   ),(Z(138),RHOMIN),  PH1 970
     3(Z(139),STAB), (Z(140),XIENRG),  (Z(141),XKENRG), (Z(142),XTENRG),PH1 980
     4(Z(143),STT   ),(Z(144),DTMIN ), (Z(145),TRNSFC),(Z(146),EMOT  ),  PH1 990
     5(Z(147),JPROJ ),(Z(148),CNAUT ), (Z(149),BBAR  ),(Z(150),EMOB  )  PH11000
C                                                                       PH11010
C     ..................................................................PH11020
C                                                                       PH11030
C     END OF COMMON                                                     PH11040
C                                                                       PH11050
C     ..................................................................PH11060
C
C            *** PH1 COMPUTES THE EFFECT OF THE PRESSURE GRADIENTS ON   PH11062
C                UPDATING THE VELOCITIES AND INTERNAL ENERGIES.         PH11064
C                                                                       PH11070
C            *** NRT AND NRC ARE USED TO ADVANCE THE ACTIVE GRID.       PH11075
      NRT=0                                                             PH11080
      NRC=0                                                             PH11090
C            *** VEL=1. FLAGS FIRST PASS. ON SECOND PASS, VEL = 0.      PH11095
      VEL=1.0                                                           PH11100
```

```
C            *** RC = DISTANCE FROM AXIS TO CENTER OF CELL K.
C                RR = DISTANCE FROM AXIS TO CENTER OF CELL K+1.
10    RC=DX(1)/2.0
      RR=RC+(DX(1)+DX(2))/2.0
      K=2
C            *** FOR ALL CELLS IN COLUMN NEXT TO AXIS, SET PRESSURE
C                AT LEFT SIDE OF CELL = PRESSURE IN CELL, AND SET
C                RADIAL VELOCITY AT LEFT SIDE OF CELL = 0.
      DO 20 J=1,JMAX
      PL(J)=P(K)
      UL(J)=0.0
20    K=K+IMAX
      DO 140 I=1,I1
      K=I+1
C            *** DEFINE PRESSURE AND AXIAL VELOCITY AT BOTTOM
C                BOUNDARY OF GRID.
      VBLO=V(K)
      PBLO=P(K)
C            *** IF BOTTOM BOUNDARY OF GRID IS REFLECTIVE, SET
C                AXIAL VELOCITY AT THAT BOUNDARY = 0.
      IF (CVIS.GT.(-.5)) VBLO=0.
      TAUDTS=TAU(I)*DT
      DO 130 J=1,I2
      N=K+IMAX
      PIDTS=1.0/(PIDY*DT*DY(J))
      IF (AMX(K).LE.0.) GO TO 30
      IF (I.LT.IMAX) GO TO 50
C            *** FOR ALL CELLS IN LAST COLUMN OF GRID, SET PRESSURE
C                AT RIGHT OF CELL = PRESSURE IN CELL.     COMPUTE
C                ENERGY LOST ACROSS RIGHT BOUNDARY AND SUBTRACT IT
C                FROM ETH, THEORETICAL ENERGY TOTAL.
      PRR=P(K)
      E=PRR*U(K)/PIDTS*X(I)
      ETH=ETH-E
      EOR=EOR-E
      GO TO 40
C            *** CELL K IS EMPTY
30    PL(J)=0.
      UL(J)=U(K+1)*RR
      PBLO=0.
      VBLO=V(N)
      GO TO 130
40    URR=RC*U(K)
      GO TO 70
C                  *** IF CELL ON RIGHT IS EMPTY SET SPECIAL P AND U
50    IF (AMX(K+1).GT.0.) GO TO 60
      PRR=0.
      URR=U(K)*RC
      GO TO 70
60    PRR=(P(K)+P(K+1))/2.
      URR=(U(K)*RC+U(K+1)*RR)/2.
70    IF (J.LT.JMAX) GO TO 80
C            *** FOR ALL CELLS IN TOP ROW OF GRID, SET PRESSURE AND
C                AXIAL VELOCITY AT TOP OF CELL = PRESSURE AND AXIAL
C                VELOCITY IN CELL. COMPUTE ENERGY LOST ACROSS TOP
C                BOUNDARY.
      PABOVE=P(K)
```

```
      E=PABOVE*V(K)/2.*TAUDTS                                           PH11500
      ETH=ETH-E                                                         PH11510
      EOT=EOT-E                                                         PH11520
      VABOVE=V(K)                                                       PH11530
      GO TO 110                                                         PH11540
C                 *** IF CELL ABOVE IS EMPTY SET SPECIAL P AND V        PH11550
80    IF (AMX(N).GT.0.) GO TO 90                                        PH11560
      PABOVE=0.                                                         PH11570
      VABOVE=V(K)                                                       PH11580
      GO TO 100                                                         PH11590
90    PABOVE=(P(K)+P(N))/2.                                             PH11600
      VABOVE=(V(K)+V(N))/2.                                             PH11610
100   IF (J.GT.1) GO TO 110                                             PH11620
C           *** IF BOTTOM BOUNDARY OF GRID IS REFLECTIVE, ADD TO ETH    PH11622
C               THE ENERGY GENERATED BY PRESSURE AT THAT BOUNDARY.      PH11624
      IF (CVIS.GT.-.5) GO TO 110                                        PH11630
      E=PBLO*V(K)/2.*TAUDTS                                             PH11640
      ETH=ETH+E                                                         PH11650
      EOB=EOB+E                                                         PH11660
110   IF (VEL.EQ.0.) GO TO 120                                          PH11670
C           *** COMPUTE UPDATED VELOCITIES ON FIRST PASS (VEL = 1.)     PH11675
      V(K)=V(K)+(PBLO-PABOVE)*TAUDTS/(AMX(K))                           PH11680
      U(K)=U(K)+(PL(J)-PRR)/(AMX(K))*RC/PIDTS*2.0                       PH11690
120   CONTINUE                                                          PH11700
C           *** AIX(X) CHANGED ON BASIS OF GRADIENTS COMPUTED IN FIRST  PH11702
C               PASS.  ON  SECOND PASS AIX(K) CHANGED AGAIN ON BASIS    PH11704
C               OF GRADIENTS CACULATED FROM THE UPDATED VELOCITIES.     PH11706
      WS=(VBLO-VABOVE)*TAUDTS/2.                                        PH11710
      WS=(UL(J)-URR)/PIDTS+WS                                           PH11720
      WSA=AIX(K)+WS*P(K)/AMX(K)                                         PH11730
      AIX(K)=WSA                                                        PH11740
      VBLO=VABOVE                                                       PH11750
      PL(J)=PRR                                                         PH11760
      UL(J)=URR                                                         PH11770
      PBLO=PABOVE                                                       PH11780
C           *** RC, N, RR REDEFINED FOR NEXT CELL IN ROW J.             PH11785
130   K=N                                                               PH11790
      RC=RR                                                             PH11800
140   RR=(X(I+1)+X(I+2))/2.0                                            PH11810
      IF (VEL.EQ.0.) GO TO 150                                          PH11820
      VEL=0.0                                                           PH11830
      GO TO 10                                                          PH11840
150   DO 190 I=1,I1                                                     PH11850
      K=I+1                                                             PH11860
      DO 180 J=1,I2                                                     PH11870
C           *** SN = 0 (INPUT PARAMETER) SETS NEGATIVE INTERNAL         PH11872
C               ENERGIES TO ZERO.                                       PH11874
      IF (AIX(K).GE.0..OR.SN.GT.0.) GO TO 170                           PH11880
      E=AIX(K)*AMX(K)                                                   PH11890
      ETH=ETH-E                                                         PH11900
      EZPH1=EZPH1-E                                                     PH11910
      IF (INTER.EQ.0) GO TO 160                                         PH11920
C           *** PRINT PROPERTIES OF CELLS WHOSE NEGATIVE ENERGY IS      PH11922
C               SET TO ZERO WHEN DOING INTERMEDIATE PRINTS (INTER.GT.0).PH11924
      WRITE (6,240) I,J,AMX(K),AIX(K),U(K),V(K)                         PH11930
160   AIX(K)=0.0                                                        PH11940
170   IF (I.NE.I1) GO TO 180                                            PH11950
```

```
C            *** ENLARGE ACTIVE GRID IN I-DIRECTION IF A CELL IN THE I1 PH11952
C                COLUMN HAS NONZERO VELOCITY OR ENERGY.                 PH11954
      IF (U(K).NE.0..OR.V(K).NE.0..OR.AIX(K).NE.0.) NRC=1                PH11960
180   K=K+IMAX                                                           PH11970
      LL=K-2*IMAX                                                        PH11980
C            *** ENLARGE ACTIVE GRID IN J-DIRECTION IF A CELL IN THE I2 PH11982
C                ROW HAS NONZERO VELOCITY OR ENERGY.                    PH11984
      IF (U(LL).NE.0..OR.V(LL).NE.0..OR.AIX(LL).NE.0.) NRT=1             PH11990
190   CONTINUE                                                           PH12000
      I1=I1+NRC                                                          PH12010
      I2=I2+NRT                                                          PH12020
C            *** DONT ALLOW ACTIVE GRID TO EXCEED IMAX BY JMAX GRID.    PH12025
      IF (I1-IMAX) 210,210,200                                           PH12030
200   I1=IMAX                                                            PH12040
210   IF (I2-JMAX) 230,230,220                                           PH12050
220   I2=JMAX                                                            PH12060
230   RETURN                                                             PH12070
C                                                                        PH12080
240   FORMAT (4H PH1,2I4,4H  M=,1PE15.8,6H  SIE=,1PE15.8,4H  U=,1PE15.8,PH12090
     14H  V=,1PE15.8,18H   SIE SET TO ZERO)                              PH12100
      END                                                                PH12110-
```

```
      SUBROUTINE PH3                                                    PH3  10
C     ..................................................................PH3  20
C                                                                       PH3  30
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,      PH3  40
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,                PH3  50
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),   PH3  60
     3          GAMC(102),                                             PH3  70
     4          PK(15),   Z(150)   ,                                   PH3  80
     5          XP(26,51),YP(26,51),                                   PH3  90
     6          PL(204)  ,UL(204)  ,PR(204)  ,                         PH3 100
     7           RSN(52),   RST(52),                                   PH3 110
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,               PH3 120
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,               PH3 130
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)      PH3 140
C            *** DIMENSIONED ARRAYS                                     PH3 150
C            *** Z-BLOCK IS SAVED ON TAPE.                              PH3 160
      COMMON     Z                                                     PH3 170
      COMMON    PK                                                     PH3 180
      COMMON  YY,      XX                                              PH3 190
      COMMON  DDX,     DDY                                             PH3 200
      COMMON  AMX,     AIX,     U,     V,     P                       PH3 210
      COMMON  TAU,     JPM                                             PH3 220
      COMMON  UL ,     PL                                              PH3 230
      COMMON  XP ,     YP,      CMXP, CMYP                             PH3 240
C            *** NON-DIMENSIONED VARIABLES                              PH3 250
      COMMON         AID    ,AMMV  ,AMMY  ,AMPY  ,AMUR  ,AMUT  ,AMVR  ,PH3 260
     1AMVT  ,DELEB ,DELER ,DELET ,DELM  ,DTODX ,DXYMIN,EAMMP ,EAMPY ,   PH3 270
     2E     ,ERDUMP,I     ,I3    ,IWS   ,J     ,K     ,KA    ,KB    ,   PH3 280
     3LL    ,MD    ,ME    ,MZT   ,NERR  ,NK    ,NPRINT,                PH3 290
     4NR    ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR   ,SNT   ,STR   ,SOLID ,  PH3 300
     5SUM   ,TESTRH,TWOPI ,URR   ,WS    ,WSA   ,WSB   ,WSC   ,WFLAGF,   PH3 310
     6WFLAGL,WFLAGP                                                     PH3 320
C                                                                       PH3 330
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE              PH3 340
C                X(0), Y(0), DX(0), DY(0)                               PH3 350
C                                                                       PH3 360
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))                         PH3 370
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))                     PH3 380
C                                                                       PH3 390
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY                      PH3 400
C                                                                       PH3 410
      EQUIVALENCE          (UL,FLEFT),         (UL(103),YAMC),          PH3 420
     1                     (PL,GAMC,PR),       (PL(103),SIGC)           PH3 430
C                                                                       PH3 440
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY                      PH3 450
C                                                                       PH3 460
      EQUIVALENCE          (UL,RSN),                                   PH3 470
     1                     (PL,RST),           (P,UK),                 PH3 480
     2                     (P(157),VK),        (P(313),SNB),           PH3 490
     3                     (P(365),STB),       (P(417),RHO)            PH3 500
C                                                                       PH3 510
C            *** SPECIAL EQUIVALENCES FOR EDIT                          PH3 520
C                                                                       PH3 530
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)                             PH3 540
C                                                                       PH3 550
C            *** Z-STORAGE EQUIVALENCES                                 PH3 560
C                                                                       PH3 570
      EQUIVALENCE                        (Z(  1),PROB  ),(Z(  2),CYCLE ), PH3 580
```

```
     1(Z(  3),DT     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  PH3 590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ),(Z(  9),TOPMU ),(Z( 10),RTMU  ),  PH3 600
     3(Z( 11),STK1   ),(Z( 12),NUMREZ),(Z( 13),ETH   ),(Z( 14),UN14  ),  PH3 610
     4(Z( 15),RHINIT),(Z( 16),PROJI ),(Z( 17),UN17  ),(Z( 18),XMAX  ),  PH3 620
     5(Z( 19),NZ     ),(Z( 20),NREZ  ),(Z( 21),AMDM  ),(Z( 22),UVMAX ),  PH3 630
     6(Z( 23),UN23   ),(Z( 24),DMIN  ),(Z( 25),JSTR  ),(Z( 26),DTNA  ),  PH3 640
     7(Z( 27),CVIS   ),(Z( 28),STK2  ),(Z( 29),STEZ  ),(Z( 30),NC    ),  PH3 650
     8(Z( 31),UN31   ),(Z( 32),NRC   ),(Z( 33),IMAX  ),(Z( 34),IMAXA ),  PH3 660
     9(Z( 35),JMAX   ),(Z( 36),JMAXA ),(Z( 37),KMAX  ),(Z( 38),KMAXA )  PH3 670
      EQUIVALENCE                                                        PH3 680
     1(Z( 39),BOTM   ),(Z( 40),BOTMV ),(Z( 41),NUMSPT),(Z( 42),CZERO ),  PH3 690
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ),(Z( 45),PRDELT),(Z( 46),PRFACT)   PH3 700
      EQUIVALENCE                                                        PH3 710
     1(Z( 47),I1     ),(Z( 48),I2    ),(Z( 49),IPCYCL),(Z( 50),TSTOP ),  PH3 720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ),(Z( 53),N3    ),(Z( 54),IVARDY),  PH3 730
     3(Z( 55),VT     ),(Z( 56),N6    ),(Z( 57),RTM   ),(Z( 58),RTMV  ),  PH3 740
     4(Z( 59),UN59   ),(Z( 60),N10   ),(Z( 61),N11   ),(Z( 62),GAMMA ),  PH3 750
     5(Z( 63),TOPM   ),(Z( 64),BOTMU ),(Z( 65),SN    ),(Z( 66),TOPMV ),  PH3 760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP),(Z( 69),PRXRT ),(Z( 70),CYCPH3),  PH3 770
     7(Z( 71),REZFCT),(Z( 72),TARGI ),(Z( 73),PROJU ),(Z( 74),BBOUND),  PH3 780
     8(Z( 75),EVAP   ),(Z( 76),ECK   ),(Z( 77),NECYCL),(Z( 78),II    ),  PH3 790
     9(Z( 79),JJ     ),(Z( 80),NMP   ),(Z( 81),Y2    ),(Z( 82),EZPH1 )  PH3 800
      EQUIVALENCE                                                        PH3 810
     1(Z( 83),IVARDX),(Z( 84),T     ),(Z( 85),NMPMAX),(Z( 86),PMIN  ),  PH3 820
     2(Z( 87),INTER ),(Z( 88),TAYBOT),(Z( 89),TAYTOP),(Z( 90),IEMAP ),  PH3 830
     3(Z( 91),MC     ),(Z( 92),MR    ),(Z( 93),MZ    ),(Z( 94),MB    )  PH3 840
      EQUIVALENCE                                                        PH3 850
     1(Z( 95),REZ    ),(Z( 96),NODUMP),(Z( 97),UN97  ),(Z( 98),UN98  ),  PH3 860
     2(Z( 99),UN99   ),(Z(100),EVAPM ),(Z(101),EVAPEN),(Z(102),EVAPMU),  PH3 870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ),(Z(105),SNL   ),(Z(106),STL   ),  PH3 880
     4(Z(107),TAXRT ),(Z(108),IDNMAP),(Z(109),IPRMAP),(Z(110),ROEPS ),  PH3 890
     5(Z(111),RHINI ),(Z(112),VINI  ),(Z(113),FINAL ),(Z(114),IVMAP ),  PH3 900
     6(Z(115),RHOZ  ),(Z(116),ESA   ),(Z(117),ESEZ  ),(Z(118),ESB   ),  PH3 910
     7(Z(119),ESCAPA),(Z(120),ESESP ),(Z(121),ESESQ ),(Z(122),ESES  ),  PH3 920
     8(Z(123),ESALPH),(Z(124),ESBETA),(Z(125),ESCAPB),(Z(126),IUMAP ),  PH3 930
     9(Z(127),SS1    ),(Z(128),SS2   ),(Z(129),UMIN  ),(Z(130),SS4   )  PH3 940
      EQUIVALENCE                                                        PH3 950
     1(Z(131),PRTIME),(Z(132),EOR   ),(Z(133),EOT   ),(Z(134),EOB   ),  PH3 960
     2(Z(135),EMOR   ),(Z(136),DXF   ),(Z(137),DYF   ),(Z(138),RHOMIN),  PH3 970
     3(Z(139),STAB), (Z(140),XIENRG), (Z(141),XKENRG), (Z(142),XTENRG),PH3 980
     4(Z(143),STT    ),(Z(144),DTMIN ),(Z(145),TRNSFC),(Z(146),EMOT  ),  PH3 990
     5(Z(147),JPROJ  ),(Z(148),CNAUT ),(Z(149),BBAR  ),(Z(150),EMOB  )  PH31000
C                                                                        PH31010
C     ..................................................................PH31020
C                                                                        PH31030
C     END OF COMMON                                                      PH31040
C                                                                        PH31050
C     ..................................................................PH31060
      DX(0)=-DX(1)                                                       PH31070
      DY(0)=-DY(1)                                                       PH31080
C                                                                        PH31090
C           *** TURN ON R-P TREATMENT WHEN ACTIVE-GRID REACHES JSTR      PH31100
      IF (I2.LT.JSTR) GO TO 400                                          PH31110
C           *** TURN OFF JSTR                                            PH31120
      JSTR=0                                                             PH31130
      PW=0.                                                              PH31140
C           *** USE P-STORAGE FOR U,V,SIE BEING CALC.                    PH31150
```

```
      DO 10 K=2,KMAX                                                    PH31160
10    P(K)=0.                                                           PH31170
C             *** CALCULATE SUBCYCLE DT                                 PH31180
      ICP3 = INT(CYCPHS)                                                PH31190
C             *** CALCULATE FACTOR FOR VARIABLE DT                      PH31200
      N=ICP3*(ICP3+1)                                                   PH31210
      DTFACT=2./FLOAT(N)                                                PH31220
C             *** LOOP THRU SUBCYCLES                                   PH31230
      DO 380 NN=1,ICP3                                                  PH31240
C             *** DECREASING DT                                         PH31250
      WS=ICP3-NN+1                                                      PH31260
      DTSTR=WS*DTFACT*DT                                                PH31270
C                                                                       PH31280
C             *** INITIALIZE UK,VK FOR ROW1,2 AND BORDER BELOW          PH31290
C             *** NOTE THAT THESE ARE STORED WITH AN EXTRA CELL TO      PH31300
C                 RIGHT AND LEFT OF MESH. SO K = 2 IS AXIS CELL.        PH31310
C                                                                       PH31320
      VFACT=-1.                                                         PH31330
C             ***     IF REFLECTIVE,PUT NEG. V IN BORDER CELLS          PH31340
      IF (CVIS.LT.0.) VFACT=1.                                          PH31350
C             *** BUT IF TRANS., USE V                                  PH31360
      NKB=1                                                             PH31370
      NK=2                                                              PH31380
      NKA=3                                                             PH31390
C             *** SET LIMITS USED IN PH3                                PH31400
      M=I1+1                                                            PH31410
      LL=I1-1                                                           PH31420
      DO 20 K=2,M                                                       PH31430
      L=K+IMAX                                                          PH31440
      I=K-1                                                             PH31450
C             *** SET VALUES ADJOINING BOTTOM ROW                       PH31460
      RHO(K,NK)=AMX(K)/(TAU(I)*DY(1))                                   PH31470
      RHO(K,NKB)=RHO(K,NK)                                              PH31480
      RHO(K,NKA)=AMX(L)/(TAU(I)*DY(2))                                  PH31490
      UK(K,NK)=U(K)                                                     PH31500
      UK(K,NKB)=U(K)                                                    PH31510
      UK(K,NKA)=U(L)                                                    PH31520
      VK(K,NK)=V(K)                                                     PH31530
      VK(K,NKB)=V(K)*VFACT                                              PH31540
20    VK(K,NKA)=V(L)                                                    PH31550
C             *** BORDER CELL TO LEFT                                   PH31560
      DO 30 N=1,3                                                       PH31570
      RHO(1,N)=RHO(2,N)                                                 PH31580
      UK(1,N)=UK(2,N)                                                   PH31590
30    VK(1,N)=VK(2,N)                                                   PH31600
C             *** SNB AND STB HAVE BEEN SET TO 0. BY SETTING ALL        PH31610
C                 P STORAGE TO 0.                                       PH31620
C             *** SET NORMAL STRESSES ON BOTTOM IF REFLECTIVE           PH31630
      IF (CVIS.LT.0.) GO TO 100                                         PH31640
      IF (IMAX.GT.1) GO TO 40                                           PH31650
C             ***IMAX=1                                                 PH31660
      WSA=1.-AIX(2)/STEZ                                                PH31670
      IF (WSA.LT.0) WSA=0.                                              PH31680
      WSB=AMX(2)/(TAU(1)*DY(1)*RHOZ)-1.                                 PH31690
      STRENG=(CZERO+WSB*(STK1+WSB*STK2))*WSA                            PH31700
      IF (STRENG.LT.0.) STRENG=0.                                       PH31710
      SNB(2)=STRENG*SQRT(2)*ABS(V(2))/V(2)                              PH31720
```

```
      GO TO 100                                                         PH31730
C                                                                       PH31740
C             ***PUT IWS = RIGHT BOUNDARY OF JPM ARRAY                  PH31750
40    IWS=1                                                             PH31760
      DO 50 I=1,IMAX                                                    PH31770
      IF (JPM(I).EQ.0) GO TO 60                                         PH31780
      IWS=IWS+1                                                         PH31790
50    CONTINUE                                                          PH31800
C                                                                       PH31810
60    DO 90 K=2,IWS                                                     PH31820
      WSA=.5*DX(K)+DX(K-1)+.5*DX(K-2)                                   PH31830
      DUODX=(U(K+1)-U(K-1))/WSA                                         PH31840
      DVODX=(V(K+1)-V(K-1))/WSA                                         PH31850
      DVODY=2*V(K)/DY(1)                                                PH31860
      UOX=2*U(K)/(X(K-1)+X(K-2))                                        PH31870
      WSA=DUODX+DVODY+UOX                                               PH31880
      TH03=WSA/3.                                                       PH31890
      WS=DUODX**2+DVODY**2+UOX**2+.5*(DVODX**2)-TH03*WSA                PH31900
      IF (WS.LE.0.) GO TO 70                                            PH31910
      WSA=1.-AIX(K)/STEZ                                                PH31920
      IF (WSA.LT.0.) WSA=0.                                             PH31930
      WSB=AMX(K)/(TAU(K-1)*DY(1)*RHOZ)-1.                               PH31940
      STRENG=(CZERO+WSB*(STK1+WSB*STK2))*WSA                            PH31950
      IF (STRENG.LT.0.) STRENG=0.                                       PH31960
      B=STRENG*SQRT(2./WS)                                              PH31970
      GO TO 80                                                          PH31980
70    B=0.                                                              PH31990
80    SNB(K)=B*(DVODY-TH03)                                             PH32000
90    CONTINUE                                                          PH32010
100   L=I2-1                                                            PH32020
C             *** DO ROWS                                               PH32030
      DO 370 J=1,L                                                      PH32040
      K=(J-1)*IMAX+2                                                    PH32050
C             *** STRESS AT AXIS =0.                                    PH32060
      SNL=0.                                                            PH32070
      STL=0.                                                            PH32080
C             *** LOOP ON CELLS ACROSS ROW                              PH32090
      DO 330 I=1,LL                                                     PH32100
      IK=I+1                                                            PH32110
C             *** IF NOT SOLID, SKIP STRESS CALCULATION                 PH32120
      IF (RHO(IK,NK).LT.SOLID) GO TO 170                                PH32130
C             *** IF ABOVE JPM(I)+1 WE ARE DONE WITH THIS ROW UNLESS ALSOPH32140
C                 BELOW JPM(I-1)                                        PH32150
      JFLAG=0                                                           PH32160
      IF (J.LE.JPM(I)+1) GO TO 110                                      PH32170
      IF (I.EQ.1) GO TO 340                                             PH32180
      IF (J.GT.JPM(I-1)) GO TO 340                                      PH32190
      JFLAG=1                                                           PH32200
C             *** CALCULATE STRENGTH                                    PH32210
110   WSA=1.-AIX(K)/STEZ                                                PH32220
      IF (WSA.LT.0.) GO TO 170                                          PH32230
      WSB=RHO(IK,NK)/RHOZ-1.                                            PH32240
      STRENG=(CZERO+WSB*(STK1+WSB*STK2))*WSA                            PH32250
      IF (STRENG.LT.0.) GO TO 170                                       PH32260
C                                                                       PH32270
C     ..................................................................PH32280
C                 *** HOOP STRESS***                                    PH32290
```

```
C     ..................................................................PH32300
      HOOP=0.                                                           PH32310
      IF (J.EQ.JPM(I)+1.OR.JFLAG.EQ.1) GO TO 170                        PH32320
C            *** SKIP HOOP AND RT CALC IF 1-D                           PH32330
      IF (IMAX.EQ.1) GO TO 140                                          PH32340
C            *** DIFFERENCES ARE CENTERED AT CELL-CENTER                PH32350
      UOX=UK(IK,NK)/(X(I)+X(I-1))*2.                                    PH32360
      WS=1./(.5*DX(I+1)+DX(I)+.5*DX(I-1))                               PH32370
      DUODX=(UK(IK+1,NK)-UK(IK-1,NK))*WS                                PH32380
      DVODX=(VK(IK+1,NK)-VK(IK-1,NK))*WS                                PH32390
      WS=1./(.5*DY(J+1)+DY(J)+.5*DY(J-1))                               PH32400
      DUODY=(UK(IK,NKA)-UK(IK,NKB))*WS                                  PH32410
      DVODY=(VK(IK,NKA)-VK(IK,NKB))*WS                                  PH32420
      ASSIGN 120 TO LOCA                                                PH32430
      GO TO 240                                                         PH32440
C           *** CALCULATED THO3,B                                       PH32450
120   HOOP=B*(UOX-THO3)                                                 PH32460
C     ..................................................................PH32470
C     ......*** END OF HOOP CALC.                                       PH32480
C     ..................................................................PH32490
C            *** IF THE CELL ON RIGHT IS NOT SOLID, STRESSES ARE ZERO   PH32500
      IF (RHO(IK+1,NK).LT.SOLID.OR.IK.EQ.LL) GO TO 140                  PH32510
C            *** DIFFERENCES ARE CENTERED AT RIGHT EDGE OF CELL         PH32520
      WS=2./(DX(I)+DX(I+1))                                             PH32530
      DUODX=(UK(IK+1,NK)-UK(IK,NK))*WS                                  PH32540
      DVODX=(VK(IK+1,NK)-VK(IK,NK))*WS                                  PH32550
      WS=1./(DY(J+1)+2.*DY(J)+DY(J-1))                                  PH32560
      DUODY=(UK(IK,NKA)+UK(IK+1,NKA)-UK(IK,NKB)-UK(IK+1,NKB))*WS        PH32570
      DVODY=(VK(IK,NKA)+VK(IK+1,NKA)-VK(IK,NKB)-VK(IK+1,NKB))*WS        PH32580
      UOX=(UK(IK+1,NK)+UK(IK,NK))/X(I)*.5                               PH32590
C                                                                       PH32600
      ASSIGN 130 TO LOCA                                                PH32610
C            *** CALC. THO3 AND B                                       PH32620
      GO TO 240                                                         PH32630
130   SNR=B*(DUODX-THO3)                                                PH32640
      STR=B*(DUODY+DVODX)*.5                                            PH32650
      GO TO 150                                                         PH32660
140   SNR=0.                                                            PH32670
      STR=0.                                                            PH32680
C                  *** IF THE CELL ABOVE IS NOT SOLID, STRESSES ABOVE AREPH32690
150   IF (RHO(IK,NKA).LT.SOLID.OR.J.EQ.L) GO TO 180                     PH32700
C                                                                       PH32710
C            *** DIFFERENCES ARE CENTERED AT TOP EDGE OF CELL           PH32720
      WS=2./(DY(J+1)+DY(J))                                             PH32730
      DUODY=(UK(IK,NKA)-UK(IK,NK))*WS                                   PH32740
      DVODY=(VK(IK,NKA)-VK(IK,NK))*WS                                   PH32750
      WS=1./(DX(I+1)+2.*DX(I)+DX(I-1))                                  PH32760
      DUODX=(UK(IK+1,NK)+UK(IK+1,NKA)-UK(IK-1,NK)-UK(IK-1,NKA))*WS      PH32770
      DVODX=(VK(IK+1,NK)+VK(IK+1,NKA)-VK(IK-1,NK)-VK(IK-1,NKA))*WS      PH32780
      UOX=(UK(IK,NKA)+UK(IK,NK))/(X(I)+X(I-1))                          PH32790
      ASSIGN 160 TO LOCA                                                PH32800
      GO TO 240                                                         PH32810
160   SNT=B*(DVODY-THO3)                                                PH32820
      STT=B*(DUODY+DVODX)*.5                                            PH32830
      GO TO 190                                                         PH32840
170   SNR=0.                                                            PH32850
      STR=0.                                                            PH32860
```

```
180   SNT=0.                                                            PH32870
      STT=0.                                                            PH32880
      IF (AMX(K).GT.0.) GO TO 190                                       PH32890
      GO TO 230                                                         PH32900
190   IF (INTER.NE.99) GO TO 200                                        PH32910
      WRITE (6,410) I,J,IK,NKB,NK,NKA,HOOP,SNL,STL,SNR,STR,STRENG,SNB(IKPH32920
     1),STB(IK),SNT,STT                                                 PH32930
C                 *** ALL STRESSES ON CELL K HAVE BEEN CALCULATED. NOW TPH32940
C            ***    NEW VALUES OF U,V, AND S.I.E. CAN BE DETERMINED.   PH32950
200   SNLX=SNL*X(I-1)                                                   PH32960
      WS=TWOPI*DTSTR/AMX(K)                                             PH32970
      IF (IMAX.EQ.1) GO TO 210                                          PH32980
      DELU=WS*(DY(J)*(SNR*X(I)-SNLX)+TAU(I)/TWOPI*(STT-STB(IK))-HOOP*DX(PH32990
     1I)*DY(J))                                                         PH33000
210   STLX=STL*X(I-1)                                                   PH33010
      DELV=WS*((SNT-SNB(IK))*TAU(I)/TWOPI+DY(J)*(STR*X(I)-STLX))        PH33020
C                                                                       PH33030
C                 *** NOW CALC. CHANGE OF ENERGY                        PH33040
C            *** UKT AND VKT ARE TEMPORARY STORAGE                      PH33050
      UKT=UK(IK,NK)                                                     PH33060
      VKT=VK(IK,NK)                                                     PH33070
      WS=TAU(1)*.5*(((UKT+UK(IK,NKA))*STT+(VKT+VK(IK,NKA))*SNT)-((UKT+UKPH33080
     1(IK,NKB))*STB(IK)+(VKT+VK(IK,NKB))*SNB(IK)))                      PH33090
      WSA=PIDY*DY(J)*(X(I)*((UK(IK+1,NK)+UKT)*SNR+(VK(IK+1,NK)+VKT)*STR)PH33100
     1-(X(I-1)*((UKT+UK(IK-1,NK))*SNL+(VKT+VK(IK-1,NK))*STL)))          PH33110
      WSB=(WSA+WS)/AMX(K)*DTSTR                                         PH33120
      WSC=DELU*(UKT+DELU/2.)+DELV*(VKT+DELV/2.)                         PH33130
      DELI=WSB-WSC                                                      PH33140
      IF (INTER.NE.99) GO TO 220                                        PH33150
      WRITE (6,420) I,J,IK,DELU,DELV,DELI                               PH33160
220   U(K)=U(K)+DELU                                                    PH33170
      V(K)=V(K)+DELV                                                    PH33180
      AIX(K)=AIX(K)+DELI                                                PH33190

      BBOUND=BBOUND+DELI*AMX(K)                                         PH33200
C            *** CELL K IS DONE. MOVE TOP TO BOTTOM AND RIGHT TO LEFT   PH33210
230   SNB(IK)=SNT                                                       PH33220
      STB(IK)=STT                                                       PH33230
      SNL=SNR                                                           PH33240
      STL=STR                                                           PH33250
      GO TO 280                                                         PH33260
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - PH33270
C            *** CALCULATE THO3 AND B                                   PH33280
C                                                                       PH33290
240   WSA=DUODX+DVODY+UOX                                               PH33300
      THO3=WSA/3.                                                       PH33310
      WS=DUODX**2+DVODY**2+UOX**2+.5*(DUODY+DVODX)**2-THO3*WSA          PH33320
      IF (WS.LE.0.) GO TO 250                                           PH33330
      B=STRENG*SQRT(2./WS)                                              PH33340
      GO TO 260                                                         PH33350
250   B=0.                                                              PH33360
260   IF (INTER.NE.99) GO TO 270                                        PH33370
      WRITE (6,430) DUODX,DVODY,UOX,THO3,DUODY,DVODX,WS,B,LOCA          PH33380
C                                                                       PH33390
270   GO TO LOCA, (130,160,120)                                         PH33400
C                                                                       PH33410
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - PH33420
```

```
C                 END OF THO3 AND B CALCULATION                          PH33430
C     - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -PH33440
C                                                                        PH33450
280   IF (INTER.NE.99) GO TO 330                                         PH33460
      E=0.                                                               PH33470
      DO 290 LJD=2,KMAX                                                  PH33480
      E=E+AMX(LJD)*(.5*(U(LJD)**2+V(LJD)**2)+AIX(LJD))                   PH33490
290   CONTINUE                                                           PH33500
      WRITE (6,440) I,J,E                                                PH33510
      DO 300 LJD=2,IK                                                    PH33520
      UBAR=.5*(UK(LJD,NK)+UK(LJD,NKA))                                   PH33530
      VBAR=.5*(VK(LJD,NK)+VK(LJD,NKA))                                   PH33540
300   E=E-TAU(LJD-1)*(UBAR*STB(LJD)+VBAR*SNB(LJD))*DTSTR                 PH33550
      IKK=IK+1                                                           PH33560
      DO 310 LJD=IKK,I1                                                  PH33570
      IF (J.GT.JPM(LJD-2)) GO TO 320                                     PH33580
      UBAR=.5*(UK(LJD,NK)+UK(LJD,NKB))                                   PH33590
      VBAR=.5*(VK(LJD,NK)+VK(LJD,NKB))                                   PH33600
310   E=E-TAU(LJD-1)*(UBAR*STB(LJD)+VBAR*SNB(LJD))*DTSTR                 PH33610
320   UBAR=.5*(UK(IK+1,NK)+UK(IK,NK))                                    PH33620
      VBAR=.5*(VK(IK+1,NK)+VK(IK,NK))                                    PH33630
      E=E-TWOPI*DY(J)*X(I)*(UBAR*SNL+VBAR*STL)*DTSTR                     PH33640
      WRITE (6,440) I,J,E                                                PH33650
      PW=PW+DELI*AMX(K)                                                  PH33660
      WRITE (6,450) PW                                                   PH33670
330   K=K+1                                                              PH33680
C                 *** END OF LOOP ON I                                   PH33690
C           *** MOVE NK-POINTERS AND STORE A NEW ROW OF VELOCITIES       PH33700
340   IF (J.EQ.L) GO TO 370                                              PH33710
      NKA=NKA+1                                                          PH33720
      NK=NK+1                                                            PH33730
      NKB=NKB+1                                                          PH33740
      IF (NKA.GT.3) NKA=1                                                PH33750
      IF (NK.GT.3) NK=1                                                  PH33760
      IF (NKB.GT.3) NKB=1                                                PH33770
      K=(J+1)*IMAX+2                                                     PH33780
      DO 350 I=1,I1                                                      PH33790
      IK=I+1                                                             PH33800
      UK(IK,NKA)=U(K)                                                    PH33810
      VK(IK,NKA)=V(K)                                                    PH33820
      RHO(IK,NKA)=AMX(K)/(TAU(I)*DY(J+1))                                PH33830
350   K=K+1                                                              PH33840
C                 *** END LOOP                                           PH33850
      IF (IMAX.NE.1) GO TO 360                                           PH33860
      UK(3,NKA)=UK(2,NKA)                                                PH33870
      VK(3,NKA)=VK(2,NKA)                                                PH33880
      RHO(3,NKA)=RHO(2,NKA)                                              PH33890
C                                                                        PH33900
360   UK(1,NKA)=UK(2,NKA)                                                PH33910
      VK(1,NKA)=VK(2,NKA)                                                PH33920
      RHO(1,NKA)=RHO(2,NKA)                                              PH33930
C                 *** END OF J-LOOP                                      PH33940
370   CONTINUE                                                           PH33950
C                 *** END OF RIGID-PLASTIC CALCULATION FOR ONE DTSTR     PH33960
380   CONTINUE                                                           PH33970
      DO 390 K=2,KMAX                                                    PH33980
390   P(K)=0.                                                            PH33990
```

```
400   RETURN                                                            PH34000
C                                                                       PH34010
410   FORMAT (/6I5,/7H  HOOP=,1PE12.6,7H  SNL =,1PE12.6,7H  STL =,1PE12.PH34020
     1o,7H  SNR =,1PE12.6,7H  STR =,1PE12.6,/7HSTRENG=,1PE12.6,7H  SNB =PH34030
     2,1PE12.o,7H  STB =,1PE12.6,7H  SNT =,1PE12.6,7H  STT =,1PE12.6/)  PH34040
420   FORMAT (/3I5,/7H  DELU=,1PE12.6,7H  DELV=,1PE12.6,7H  DELI=,1PE12.PH34050
     16/)                                                               PH34060
430   FORMAT (7H  DUODX=,1PE12.o,7H DVODY=,1PE12.6,7H  UOX =,1PE12.6,7H TPH34070
     1H03 =,1PE12.o,/7H DUODY=,1PE12.6,7H DVODX=,1PE12.6,7H WS   =,1PE12PH34080
     2.o.7H  B   =,1PE12.6,I7)                                          PH34090
440   FORMAT (4X,2HI=I2,4X,2HJ=I2,4X,2HE=1PE13.7)                       PH34100
450   FORMAT (4X,3HPW=1PE12.6)                                          PH34110
      END                                                               PH34120-
```

```
      SUBROUTINE PH2                                                    PH2  10
C     ..................................................................PH2  20
C                                                                       PH2  30
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,      PH2  40
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,               PH2  50
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),  PH2  60
     3          GAMC(102),                                             PH2  70
     4          PK(15),   Z(150)   ,                                   PH2  80
     5          XP(26,51),YP(26,51),                                   PH2  90
     6          PL(204)  ,UL(204)  ,PR(204)  ,                         PH2 100
     7           RSN(52),   RST(52),                                   PH2 110
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,               PH2 120
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,               PH2 130
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)      PH2 140
C            *** DIMENSIONED ARRAYS                                     PH2 150
C            *** Z-BLOCK IS SAVED ON TAPE.                              PH2 160
      COMMON     Z                                                     PH2 170
      COMMON    PK                                                     PH2 180
      COMMON  YY,      XX                                              PH2 190
      COMMON  DDX,     DDY                                             PH2 200
      COMMON  AMX,     AIX,     U,     V,     P                       PH2 210
      COMMON  TAU,     JPM                                             PH2 220
      COMMON  UL ,     PL                                              PH2 230
      COMMON  XP ,     YP,      CMXP, CMYP                             PH2 240
C            *** NON-DIMENSIONED VARIABLES                              PH2 250
      COMMON         AID    ,AMMV   ,AMMY   ,AMPY   ,AMUR   ,AMUT   ,AMVR   ,  PH2 260
     1AMVT   ,DELEB ,DELER ,DELET ,DELM   ,DTODX ,DXYMIN,EAMMP ,EAMPY ,  PH2 270
     2E      ,ERDUMP,I     ,I3    ,IWS    ,J      ,K      ,KA     ,KB     ,  PH2 280
     3LL     ,MD    ,ME    ,MZT   ,NERR   ,NK     ,NPRINT,                 PH2 290
     4NR     ,NRZ   ,NUMLE ,PIDTS ,SIEMIN,SNR    ,SNT    ,STR    ,SOLID ,  PH2 300
     5SUM    ,TESTRH,TWOPI ,URR   ,WS     ,WSA    ,WSB    ,WSC    ,WFLAGF,  PH2 310
     6WFLAGL,WFLAGP                                                     PH2 320
C                                                                       PH2 330
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE              PH2 340
C                X(0), Y(0), DX(0), DY(0)                               PH2 350
C                                                                       PH2 360
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))                         PH2 370
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))                     PH2 380
C                                                                       PH2 390
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY                      PH2 400
C                                                                       PH2 410
      EQUIVALENCE          (UL,FLEFT),          (UL(103),YAMC),         PH2 420
     1                     (PL,GAMC,PR),        (PL(103),SIGC)          PH2 430
C                                                                       PH2 440
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY                      PH2 450
C                                                                       PH2 460
      EQUIVALENCE          (UL,RSN),                                   PH2 470
     1                     (PL,RST),            (P,UK),                PH2 480
     2                     (P(157),VK),         (P(313),SNB),          PH2 490
     3                     (P(365),STB),        (P(417),RHO)           PH2 500
C                                                                       PH2 510
C            *** SPECIAL EQUIVALENCES FOR EDIT                          PH2 520
C                                                                       PH2 530
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)                             PH2 540
C                                                                       PH2 550
C            *** Z-STORAGE EQUIVALENCES                                 PH2 560
C                                                                       PH2 570
      EQUIVALENCE                      (Z(  1),PROB  ),(Z(  2),CYCLE ), PH2 580
```

```
     1(Z(  3),DT     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  PH2 590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ),(Z(  9),TOPMU ),(Z( 10),RTMU  ),  PH2 600
     3(Z( 11),STK1  ),(Z( 12),NUMREZ),(Z( 13),ETH   ),(Z( 14),UN14  ),  PH2 610
     4(Z( 15),RHINIT),(Z( 16),PROJI ),(Z( 17),UN17  ),(Z( 18),XMAX  ),  PH2 620
     5(Z( 19),NZ    ),(Z( 20),NREZ  ),(Z( 21),AMDM  ),(Z( 22),UVMAX ),  PH2 630
     6(Z( 23),UN23  ),(Z( 24),DMIN  ),(Z( 25),JSTR  ),(Z( 26),DTNA  ),  PH2 640
     7(Z( 27),CVIS  ),(Z( 28),STK2  ),(Z( 29),STEZ  ),(Z( 30),NC    ),  PH2 650
     8(Z( 31),UN31  ),(Z( 32),NRC   ),(Z( 33),IMAX  ),(Z( 34),IMAXA ),  PH2 660
     9(Z( 35),JMAX  ),(Z( 36),JMAXA ),(Z( 37),KMAX  ),(Z( 38),KMAXA )  PH2 670
      EQUIVALENCE                                                      PH2 680
     1(Z( 39),BOTM  ),(Z( 40),BOTMV ),(Z( 41),NUMSPT),(Z( 42),CZERO ),  PH2 690
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ),(Z( 45),PRDELT),(Z( 46),PRFACT)  PH2 700
      EQUIVALENCE                                                      PH2 710
     1(Z( 47),I1    ),(Z( 48),I2    ),(Z( 49),IPCYCL),(Z( 50),TSTOP ),  PH2 720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ),(Z( 53),N3    ),(Z( 54),IVARDY),  PH2 730
     3(Z( 55),VT    ),(Z( 56),N6    ),(Z( 57),RTM   ),(Z( 58),RTMV  ),  PH2 740
     4(Z( 59),UN59  ),(Z( 60),N10   ),(Z( 61),N11   ),(Z( 62),GAMMA ),  PH2 750
     5(Z( 63),TOPM  ),(Z( 64),BOTMU ),(Z( 65),SN    ),(Z( 66),TOPMV ),  PH2 760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP),(Z( 69),PRXRT ),(Z( 70),CYCPH3),  PH2 770
     7(Z( 71),REZFCT),(Z( 72),TARGI ),(Z( 73),PROJU ),(Z( 74),BBOUND),  PH2 780
     8(Z( 75),EVAP  ),(Z( 76),ECK   ),(Z( 77),NECYCL),(Z( 78),II    ),  PH2 790
     9(Z( 79),JJ    ),(Z( 80),NMP   ),(Z( 81),Y2    ),(Z( 82),EZPH1 )  PH2 800
      EQUIVALENCE                                                      PH2 810
     1(Z( 83),IVARDX),(Z( 84),T     ),(Z( 85),NMPMAX),(Z( 86),PMIN  ),  PH2 820
     2(Z( 87),INTER ),(Z( 88),TAYBOT),(Z( 89),TAYTOP),(Z( 90),ICMAP ),  PH2 830
     3(Z( 91),MC    ),(Z( 92),MR    ),(Z( 93),MZ    ),(Z( 94),MB    )  PH2 840
      EQUIVALENCE                                                      PH2 850
     1(Z( 95),REZ   ),(Z( 96),NODUMP),(Z( 97),UN97  ),(Z( 98),UN98  ),  PH2 860
     2(Z( 99),UN99  ),(Z(100),EVAPM ),(Z(101),EVAPEN),(Z(102),EVAPMU),  PH2 870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ),(Z(105),SNL   ),(Z(106),STL   ),  PH2 880
     4(Z(107),TAXRT ),(Z(108),IDNMAP),(Z(109),IPRMAP),(Z(110),ROEPS ),  PH2 890
     5(Z(111),RHINI ),(Z(112),VINI  ),(Z(113),FINAL ),(Z(114),IVMAP ),  PH2 900
     6(Z(115),RHOZ  ),(Z(116),ESA   ),(Z(117),ESEZ  ),(Z(118),ESB   ),  PH2 910
     7(Z(119),ESCAPA),(Z(120),ESESP ),(Z(121),ESESQ ),(Z(122),ESES  ),  PH2 920
     8(Z(123),ESALPH),(Z(124),ESBETA),(Z(125),ESCAPB),(Z(126),IUMAP ),  PH2 930
     9(Z(127),SS1   ),(Z(128),SS2   ),(Z(129),UMIN  ),(Z(130),SS4   )  PH2 940
      EQUIVALENCE                                                      PH2 950
     1(Z(131),PRTIME),(Z(132),EOR   ),(Z(133),EOT   ),(Z(134),EOB   ),  PH2 960
     2(Z(135),EMOR  ),(Z(136),DXF   ),(Z(137),DYF   ),(Z(138),RHOMIN),  PH2 970
     3(Z(139),STAB), (Z(140  XIENRG), (Z(141),XKENRG), (Z(142),XTENRG),PH2 980
     4(Z(143),STT   ),(Z(144),DTMIN ),(Z(145),TRNSFC),(Z(146),EMOT  ),  PH2 990
     5(Z(147),JPROJ ),(Z(148),CNAUT ),(Z(149),BBAR  ),(Z(150),EMOB  )  PH21000
C                                                                       PH21010
C          *** SPECIAL EQUIVALENCE FOR PH2                              PH21012
      EQUIVALENCE (WSOUT,UOTK)                                          PH21014
C
C     ..................................................................PH21020
C                                                                       PH21030
C     END OF COMMON                                                     PH21040
C                                                                       PH21050
C     ..................................................................PH21060
C                                                                       PH21070
      SUME=0.                                                           PH21090
C          *** ARE TRACER POINTS BEING GENERATED                        PH21095
      IF (Y2.GT.(-1.)) GO TO 260                                        PH21100
C          *** YES. CALCULATE NEW POSITIONS OF POINTS IN ACTIVE GRID.   PH21105
      DO 250 J=1,JJ                                                     PH21110
```

```
      DO 250 I=1,II                                                     PH21120
      IF (XP(I,J).LE.0.) GO TO 250                                      PH21130
      IX=XP(I,J)                                                        PH21140
      IY=YP(I,J)                                                        PH21150
      IF (IX.GT.I1) GO TO 250                                           PH21160
      IF (IY.GT.I2) GO TO 250                                           PH21170
      K=IY*IMAX+IX+2                                                    PH21180
C            *** SKIP CALCULATION IF POINT IS IN EMPTY CELL             PH21190
C                (AHEAD OF THE MASS IT REPRESENTS).                     PH21195
      IF (AMX(K).EQ.0.) GO TO 250                                       PH21200
C            *** FRX AND FRY GIVE LOCATION OF TRACER POINT IN           PH21202
C                CELL(IX+1,IY+1).                                       PH21204
      FRX=XP(I,J)-AINT(XP(I,J))                                         PH21210
      FRY=YP(I,J)-AINT(YP(I,J))                                         PH21220
C                                                                       PH21230
C            *** IWS=1 FLAGS TRACER POINTS IN CELLS ON RIGHT OR         PH21230
C                      LEFT BOUNDARY OF GRID.                           PH21232
C                IWS=2 FLAGS TRACER POINTS IN CELLS ON TOP OR           PH21234
C                      BOTTOM BOUNDARY OF GRID.                         PH21236
C                IWS=0 FLAGS TRACER POINTS IN CELLS WHICH ARE NOT       PH21240
C                      ON A GRID BOUNDARY.                              PH21245
      IWS=0                                                             PH21250
C            *** SEE IF TRACER POINT IS IN A BOUNDARY CELL.             PH21255
      IF (IX.LT.1) GO TO 130                                            PH21260
      IF (IX.GT.IMAX-2) GO TO 110                                       PH21270
      IF (IY.LT.1) GO TO 160                                            PH21280
      IF (IY.GT.JMAX-2) GO TO 140                                       PH21290
C            *** NOT IN BOUNDARY CELL. IS POINT ON LEFT SIDE OF CELL    PH21295
10    IF (FRX.LT..5) GO TO 40                                           PH21300
C            *** POINT IS ON RIGHT SIDE OF CELL. IS CELL ON RIGHT EMPTY PH21305
20    IF (AMX(K+1).EQ.0.) GO TO 30                                      PH21310
C            *** RADIAL COMPONENT BASED ON AVERAGE OF RADIAL VELOCITIES PH21312
C                OF CELL K AND CELL ON RIGHT OR LEFT.                   PH21314
      UEFF=(FRX-.5)*U(K+1)+(1.5-FRX)*U(K)                               PH21320
      GO TO 50                                                          PH21330
C            *** CELL ON RIGHT OR LEFT EMPTY-USE RADIAL COMPONENT       PH21342
C                OF CELL K.                                             PH21344
30    UEFF=U(K)                                                         PH21340
      GO TO 50                                                          PH21350
C            *** POINT IS ON LEFT SIDE OF CELL. IS CELL ON LEFT EMPTY   PH21355
40    IF (AMX(K-1).EQ.0.) GO TO 30                                      PH21360
C            *** RADIAL COMPONENT BASED ON AVERAGE OF TWO CELLS.        PH21365
      UEFF=(.5-FRX)*U(K-1)+(.5+FRX)*U(K)                                PH21370
C            *** WHEN IWS = 2 AXIAL COMPONENT OF CELL HAS ALREADY BEEN  PH21372
C                CALCULATED.                                            PH21374
50    IF (IWS.GT.1) GO TO 100                                           PH21380
C            *** IS POINT IN BOTTOM HALF OF CELL                        PH21385
60    IF (FRY.LT..5) GO TO 90                                           PH21390
C            *** POINT IS IN TOP HALF. IS CELL ABOVE EMPTY              PH21395
70    KA=K+IMAX                                                         PH21400
      IF (AMX(KA).EQ.0.) GO TO 80                                       PH21410
C            *** AXIAL COMPONENT BASED ON AVERAGE OF AXIAL VELOCITIES   PH21412
C                OF CELL K AND CELL ABOVE OR BELOW.                     PH21414
      VEFF=(FRY-.5)*V(KA)+(1.5-FRY)*V(K)                                PH21420
      GO TO 100                                                         PH21430
C            *** CELL ABOVE OR BELOW IS EMPTY. USE AXIAL COMPONENT OF   PH21432
C                CELL K.                                                PH21434
```

```
60    VEFF=V(K)                                                        PH21440
      GO TO 100                                                        PH21450
90    KA=K-IMAX                                                        PH21460
      IF (AMX(KA).EQ.0.) GO TO 80                                      PH21470
      VEFF=(.5-FRY)*V(KA)+(.5+FRY)*V(K)                                PH21480
C            *** IX+1 IS I-INDEX AND IY+1 IS J-INDEX OF CELL TRACER    PH21482
C                POINT IS IN. COMPUTE NEW LOCATION OF TRACER POINT.    PH21484
100   DTODX=DT/DX(IX+1)                                                PH21490
      XP(I,J)=XP(I,J)+UEFF*DTODX                                       PH21500
      DTODY=DT/DY(IY+1)                                                PH21510
      YP(I,J)=YP(I,J)+VEFF*DTODY                                       PH21520
C            ***IWS.LT.1 MEANS TRACER POINT WAS NOT IN BOUNDARY CELL   PH21522
C                BEFORE BEING MOVED, AND ITS  NEW  POSITION NEED NOT   PH21524
C                BE CHECKED - GO TO END OF LOOP.                       PH21526
      IF (IWS.LT.1) GO TO 250                                          PH21530
      GO TO 200                                                        PH21540
C            *** POINT IN CELL ON RIGHT BOUNDARY.                      PH21542
110   IF (FRX.LT..5) GO TO 40                                          PH21550
      IWS=1                                                            PH21560
      UEFF=U(K)                                                        PH21570
C            *** IS POINT IN CELL ON BOTTOM BOUNDARY                   PH21575
120   IF (IY.LT.1) GO TO 170                                           PH21580
C            *** IS POINT IN CELL ON TOP BOUNDARY                      PH21585
      IF (IY.GT.JMAX-2) GO TO 190                                      PH21590
      GO TO 60                                                         PH21600
C            *** POINT IN CELL ON AXIS.                                PH21605
130   IF (FRX.GT..5) GO TO 20                                          PH21610
      IWS=1                                                            PH21620
      UEFF=2.*FRX*U(K)                                                 PH21630
      GO TO 120                                                        PH21640
C            *** POINT IN CELL ON TOP BOUNDARY.                        PH21645
140   IF (FRY.LT..5) GO TO 10                                          PH21650
      IWS=2                                                            PH21660
150   VEFF=V(K)                                                        PH21670
      GO TO 10                                                         PH21680
C            *** POINT IN CELL ON BOTTOM BOUNDARY. RADIAL COMPONENT    PH21682
C                HAS NOT BEEN COMPUTED.                                PH21684
160   IF (FRY.GT..5) GO TO 10                                          PH21690
      IWS=2                                                            PH21700
      IF (CVIS.LT.0.) GO TO 150                                        PH21710
      VEFF=2.*FRY*V(K)                                                 PH21720
      GO TO 10                                                         PH21730
C            *** POINT IN CELL ON BOTTOM BOUNDARY.    AXIAL COMPONENT  PH21732
C                HAS BEEN COMPUTED.                                    PH21734
170   IF (FRY.GT..5) GO TO 70                                          PH21740
C            *** COMPUTE AXIAL COMPONENT ON BASIS OF BOTTOM BOUNDARY   PH21742
C                CONDITION.                                            PH21744
      IF (CVIS.LT.0.) GO TO 180                                        PH21750
C            *** REFLECTIVE.                                           PH21755
      VEFF=2.*FRY*V(K)                                                 PH21760
      GO TO 100                                                        PH21770
C            *** TRANSMITTIVE.                                         PH21775
180   VEFF=V(K)                                                        PH21780
      GO TO 100                                                        PH21790
C            *** POINT IN CELL ON TOP BOUNDARY.                        PH21795
190   IF (FRY.LT..5) GO TO 90                                          PH21800
      GO TO 180                                                        PH21810
```

```
C             *** SPECIAL TESTS FOR TRACER POINTS WHICH WERE IN          PH21812
C                 BOUNDARY CELLS BEFORE BEING MOVED.                     PH21814
C             *** IF POINT MOVED BELOW GRID, TEST ON BOTTOM BOUNDARY     PH21816
C                 CONDITION, IF REFLECTIVE MOVE POINT BACK INTO GRID,    PH21818
C                 IF TRANSMITTIVE SET X-COORDINATE TO -1.                PH21819
200   IF (YP(I,J).LT.0.) GO TO 220                                       PH21820
C             *** IF POINT CROSSED TOP BOUNDARY SET ITS X-COORDINATE     PH21822
C                 TO -1.                                                 PH21824
      IF (INT(YP(I,J)).LT.JMAX) GO TO 230                                PH21830
210   XP(I,J)=-1.                                                        PH21840
      GO TO 250                                                          PH21850
220   IF (CVIS.LT.0.) GO TO 210                                          PH21860
C             *** REFLECTIVE.                                            PH21865
      YP(I,J)=-YP(I,J)                                                   PH21870
C             *** ADJUST X-COORDINATE IF ITS CALCULATED POSITION IS      PH21872
C                 NEGATIVE.                                              PH21874
230   IF (XP(I,J).LT.0.) GO TO 240                                       PH21880
C             *** IF POINT CROSSED RIGHT BOUNDARY SET ITS X-COORDINATE   PH21882
C                 TO -1.                                                 PH21884
      IF (INT(XP(I,J)).LT.IMAX) GO TO 250                                PH21890
      GO TO 210                                                          PH21900
240   XP(I,J)=-XP(I,J)                                                   PH21910
C             *** END OF LOOP FOR TRACER POINT MOVEMENT.                 PH21915
250   CONTINUE                                                           PH21920
C
C             *** SET TO ZERO ACTIVE GRID AND REZONE FLAGS.              PH21925
260   NRT=0                                                              PH21930
      NRC=0                                                              PH21940
      REZ=0.0                                                            PH21950
      PIDTS=1.0/(PIDY*DT)                                                PH21960
      TWOPDT=2./PIDTS                                                    PH21970
      K=2                                                                PH21980
C             *** CALCULATE FLUXES ON LEFT SIDE OF CELLS IN AXIS COLUMN. PH21985
      DO 310 J=1,JMAX                                                    PH21990
      IF (AMX(K).LE.0.) GO TO 270                                        PH22000
      IF (U(K).LT.0.) GO TO 280                                          PH22010
270   FLEFT(J)=0.                                                        PH22020
      GO TO 300                                                          PH22030
280   GAMC(J)=AMX(K)*U(K)*DT/DX(1)                                       PH22040
      IF ((GAMC(J)+AMX(K)).GE.0.) GO TO 290                              PH22050
      GAMC(J)=-AMX(K)                                                    PH22060
290   FLEFT(J)=2.*GAMC(J)*U(K)/SS2                                       PH22070
300   GAMC(J)=0.                                                         PH22080
      YAMC(J)=0.                                                         PH22090
      SIGC(J)=0.                                                         PH22100
310   K=K+IMAX                                                           PH22110
C             *** DO LOOP IN I-DIRECTION - MOVE UP COLUMNS - SPECIAL     PH22112
C                 TREATMENT FOR FLUXES AT BOTTOM BOUNDARY OCCURS BEFORE  PH22114
C                 J-LOOP BEGINS.                                         PH22116
      DO 1150 I=1,I1                                                     PH22120
      J=1                                                                PH22130
      K=I+1                                                              PH22140
315   IF(AMX(K)) 1220, 330, 320                                          PH22150
320   IF (-V(K).GT.UMIN) GO TO 340                                       PH22160
330   AMMV=0.0                                                           PH22170
      GO TO 390                                                          PH22180
340   AMMY=AMX(K)*V(K)*DT/DY(J)                                          PH22190
```

```
      IF (AMMY+AMX(K)) 350,360,360                                      PH22200
350   AMMY=-AMX(K)                                                      PH22210
360   IF (CVIS) 370,380,380                                             PH22220
C           *** BOTTOM BOUNDARY IS TRANSMITTIVE, MATERIAL IS MOVING     PH22230
C               OUT, REMOVE ITS ENERGY FROM ETH.                        PH22240
370   AMMU=AMMY*U(K)                                                    PH22250
      AMMV=AMMY*V(K)                                                    PH22260
      DELEB=AIX(K)+(U(K)**2+V(K)**2)/2.0                                PH22270
      DELEB=AMMY*DELEB                                                  PH22280
      EMOB=EMOB-DELEB                                                   PH22290
      ETH=ETH+DELEB                                                     PH22300
      BOTM=BOTM-AMMY                                                    PH22310
      BOTMV=BOTMV-AMMV                                                  PH22320
      BOTMU=BOTMU-AMMU                                                  PH22330
      GO TO 400                                                         PH22340
C           *** BOTTOM BOUNDARY IS REFLECTIVE, NET MOMENTUM CHANGE      PH22350
C               IN Z DIRECTION IS 2*MV/SS2.                             PH22360
380   IF (V(K).GE.0.) GO TO 330                                         PH22370
      AMMV=2.*AMMY*V(K)/SS2                                             PH22380
390   AMMY=0.0                                                          PH22390
      AMMU=0.                                                           PH22400
      DELEB=0.0                                                         PH22410
C           *** BEGIN DO LOOP IN J(Z) DIRECTION.                        PH22420
400   DO 1140 J=1,I2                                                    PH22430
      MSLAVE=0                                                          PH22440
      NSLAVE=0                                                          PH22450
      IF (J.EQ.JMAX) GO TO 420                                          PH22460
C                   NOT AT TOP OF MESH                                  PH22470
C                   L IS INDEX OF CELL ABOVE K                          PH22480
      L=K+IMAX                                                          PH22490
C                   IS CELL K EMPTY                                     PH22500
      IF (AMX(K).GT.0.) GO TO 540                                       PH22510
C                   IF CELL ABOVE IS ALSO EMPTY THEN FLUX=0 OR          PH22520
C                   IF FLUX WOULD BE OUT OF EMPTY CELL, THEN FLUX=0.    PH22530
      IF (AMX(L).EQ.0..OR.V(L).GE.0.) GO TO 430                         PH22540
C                   CELL ABOVE NOT EMPTY. MASS MOVING IN DIRECTION OF   PH22550
C                   CELL K WHICH IS EMPTY.                              PH22560
C                   IS CELL ABOVE COLD AND SOLID                        PH22570
      IF (AIX(L).GT.ESESQ.OR.AMX(L)/(TAU(I)*DY(J+1)).GE.RHOZ) GO TO 410 PH22580
C                   COLD, BUT NOT UP TO NORMAL DENSITY                  PH22590
C                   IS NEXT CELL ABOVE C OLD                            PH22600
      IF ((J+1).EQ.JMAX) GO TO 410                                      PH22610
      LA=L+IMAX                                                         PH22620
      IF (AIX(LA).LT.ESESQ.AND.AMX(LA)/(TAU(I)*DY(J+1)).GT.SOLID) GO TO PH22630
     1430                                                               PH22640
C                   CELL ABOVE IS HOT.  DO NOT HOLD BACK.               PH22650
410   M=L                                                               PH22660
      VABOVE=V(L)                                                       PH22670
      WDELF=DY(J+1)                                                     PH22680
      GO TO 460                                                         PH22690
C                   TOP OF MESH, IS MASS MOVING OUT.                    PH22700
420   IF (V(K).GT.0.) GO TO 440                                         PH22710
C                   SET FLUX TERMS TO ZERO.                             PH22720
430   AMPY=0.                                                           PH22730
      GO TO 590                                                         PH22740
C                   MASS MOVING OUT OF TOP BOUNDARY                     PH22750
440   VABOVE=V(K)                                                       PH22760
```

```
      WDYF=DY(J)                                                        PH22770
450   M=K                                                               PH22780
C                    CALCULATE MASS FLUX AT TOP OF CELL                 PH22790
460   IF (ABS(VABOVE).LE.UMIN) GO TO 430                                PH22800
      UVMAX=TRNSFC*DY(J)/DT                                             PH22810
      IF (ABS(VABOVE).LT.UVMAX) GO TO 480                               PH22820
      IF (VABOVE.GT.0.) GO TO 470                                       PH22830
      VABOVE=-UVMAX                                                     PH22840
      GO TO 480                                                         PH22850
470   VABOVE=UVMAX                                                      PH22860
480   AMPY=AMX(M)*VABOVE*DT/WDYF                                        PH22870
      IF (MSLAVE.NE.0) GO TO 500                                        PH22880
490   EAMPY=.5*(U(M)**2+V(M)**2)+AIX(M)                                 PH22890
      UAMPY=U(M)                                                        PH22900
      VAMPY=V(M)                                                        PH22910
      GO TO 590                                                         PH22920
500   IF (VABOVE.GT.0.) GO TO 510                                       PH22930
      M=L                                                               PH22940
      GO TO 490                                                         PH22950
510   M=K                                                               PH22960
      GO TO 490                                                         PH22970
520   WSA=.5*(V(K)+V(L))                                                PH22980
      WSB=1.0+(V(L)-V(K))*DT/((DY(J+1)+DY(J))/2.0)                      PH22990
      WDYF=(DY(J)+DY(J+1))/2.                                           PH23000
      VABOVE=WSA/WSB                                                    PH23010
      IF (MSLAVE.NE.0) GO TO 460                                        PH23020
      IF (VABOVE) 530,430,450                                           PH23030
530   M=L                                                               PH23040
      GO TO 460                                                         PH23050
C                    CELL K IS NOT EMPTY. HOW ABOUT CELL ABOVE K.       PH23060
540   IF (AMX(L).GT.0.) GO TO 550                                       PH23070
C                    CELL ABOVE IS EMPTY. IS FLUX INTO IT.              PH23080
      IF (V(K).LE.0.) GO TO 430                                         PH23090
C                    FLUX TOWARD EMPTY CELL                             PH23100
      IF (J.EQ.1) GO TO 440                                             PH23110
C                    SHOULD MASS BE HELD UP UNTIL CELL IS FULL          PH23120
      LB=K-IMAX                                                         PH23130
      IF (AIX(LB).GT.ESESQ.OR.AIX(K).GT.ESESQ.OR.AMX(LB)/(TAU(I)*DY(J-1)PH23140
     1).LT.SOLID.OR.AMX(K)/(TAU(I)*DY(J)).GT.SOLID) GO TO 440           PH23150
      GO TO 430                                                         PH23160
550   IF (V(K).GT.0..AND.V(L).LT.0.) GO TO 560                          PH23170
      IF ((J+1).EQ.JMAX) GO TO 580                                      PH23180
      LA=L+IMAX                                                         PH23190
      IF (AMX(LA).GT.0..OR.V(K).GE.0..OR.V(L).GE.0.) GO TO 570          PH23200
C                    K AND L NOT EMPTY BUT CELL ABOVE L IS EMPTY        PH23210
C                    TEST FOR SLAVING L TO K                            PH23220
      IF (AMX(L)/(TAU(I)*DY(J+1)).GE.RHOZ.OR.AIX(K).GT.ESESQ.OR.AIX(L).GPH23230
     1T.ESESQ) GO TO 520                                                PH23240
C                    YES, SLAVE L TO K                                  PH23250
      MSLAVE=L                                                          PH23260
      M=K                                                               PH23270
      GO TO 520                                                         PH23280
C                   *** $$REFLECTIVE$$ TREATMENT                        PH23290
560   VMK=V(K)*AMX(K)                                                   PH23300
      VML=V(L)*AMX(L)                                                   PH23310
      WSA=VMK+VML                                                       PH23320
      AMPY=WSA*DT/((DY(J)+DY(J+1))/2.0)                                 PH23330
```

```
      VAMPY=(VMK*V(K)+VML*V(L))/WSA                                     PH23340
      UAMPY=(VMK*U(K)+VML*U(L))/WSA                                     PH23350
      SAVEK=AIX(K)+.5*(U(K)**2+V(K)**2)                                 PH23360
      EAMPY=(VMK*SAVEK+VML*(AIX(L)+.5*(U(L)**2+V(L)**2)))/WSA           PH23370
      GO TO 590                                                         PH23380
570   IF (J.EQ.1) GO TO 520                                             PH23390
580   LB=K-IMAX                                                         PH23400
      IF (AMX(LB).NE.0..OR.V(L).LE.0..OR.V(K).LE.0.) GO TO 520          PH23410
C                   SHOULD K BE SLAVED TO L                             PH23420
      IF (AMX(K)/(TAU(I)*DY(J)).GE.RHOZ.OR.AIX(L).GE.ESESQ.OR.AIX(K).GT.PH23430
     1ESESQ) GO TO 520                                                  PH23440
C                   YES. SLAVE K TO L.                                  PH23450
      MSLAVE=K                                                          PH23460
      M=L                                                               PH23470
      GO TO 520                                                         PH23480
C                 *** CHECK FOR ONE-D                                   PH23490
590   IF (ABS(AMPY).LT.ROEPS*AMX(K).AND.ABS(AMPY).LT.ROEPS*AMX(K+IMAX)) PH23500
     1AMPY=0.                                                           PH23510
      IF (IMAX.EQ.1) GO TO 620                                          PH23520
      IF (I.EQ.IMAX) GO TO 610                                          PH23530
C                   NOT AT RIGHT BOUNDARY                               PH23540
C                   IS CELL K.EMPTY                                     PH23550

      IF (AMX(K).GT.0.) GO TO 730                                       PH23560
C                   SET FLUX=0 IF CELL ON RIGHT IS EMPTY                PH23570
C                   OR IF VELOCITY IS AWAY FROM EMPTY CELL K            PH23580
      IF (AMX(K+1).EQ.0..OR.U(K+1).GE.0.) GO TO 620                     PH23590
C                   CELL TO RIGHT IS NOT EMPTY. SHALL WE LET MASS MOVE  PH23600
C                   INTO CELL K WHICH IS EMPTY.                         PH23610
      IF (AIX(K+1).GT.ESESQ.OR.AMX(K+1)/(TAU(I+1)*DY(J)).GT.RHOZ) GO TO PH23620
     1600                                                               PH23630
C                   COLD AND NOT UP TO NORMAL DENSITY                   PH23640
C                   IS NEXT CELL TO RIGHT COLD                          PH23650
      IF ((I+1).EQ.IMAX) GO TO 600                                      PH23660
      IF (AIX(K+2).LT.ESESQ.AND.AMX(K+2)/(TAU(I+2)*DY(J)).GT.SOLID) GO TPH23670
     1O 620                                                             PH23680
C                 ***CELL ON RIGHT IS HOT. DO NOT HOLD BACK             PH23690
600   M=K+1                                                             PH23700
      URR=U(M)                                                          PH23710
      N=I+1                                                             PH23720
      GO TO 650                                                         PH23730
C                   RIGHT EDGE OF MESH                                  PH23740
610   IF (U(K).GT.0.) GO TO 630                                         PH23750
C                   NO MASS COMES IN FROM OUTSIDE                       PH23760
C                   SET FLUX TERMS TO ZERO                              PH23770
620   AMMP=0.                                                           PH23780
      GO TO 790                                                         PH23790
C                   MASS MOVING OUT OF RIGHT EDGE                       PH23800
630   URR=U(K)                                                          PH23810
640   N=I                                                               PH23820
      M=K                                                               PH23830
C                   CALCULATE MASS FLUX AT RIGHT OF CELL                PH23840
650   IF (ABS(URR).LE.UMIN) GO TO 620                                   PH23850
      UVMAX=TRNSFC*DX(I)/DT                                             PH23860
      IF (ABS(URR).LT.UVMAX) GO TO 670                                  PH23870
      IF (URR.GT.0.) GO TO 660                                          PH23880
      URR=-UVMAX                                                        PH23890
```

```
      GO TO 670                                                         PH23900
660   URR=UVMAX                                                         PH23910
670   AMMP=AMX(M)/TAU(N)*TWOPDT*X(I)*URR                                PH23920
C                   SET SPECIFIC ENERGY + MOMENTUM                      PH23930
      IF (NSLAVE.NE.0) GO TO 690                                        PH23940
680   EAMMP=.5*(U(M)**2+V(M)**2)+AIX(M)                                 PH23950
      UAMMP=U(M)                                                        PH23960
      VAMMP=V(M)                                                        PH23970
      GO TO 790                                                         PH23980
690   IF (URR.GT.0.) GO TO 700                                          PH23990
      M=K+1                                                             PH24000
      GO TO 680                                                         PH24010
700   M=K                                                               PH24020
      GO TO 680                                                         PH24030
710   WSA=.5*(U(K)+U(K+1))                                              PH24040
      WSB=1.+(U(K+1)-U(K))*DT/((DX(I+1)+DX(I))/2.0)                     PH24050
      URR=WSA/WSB                                                       PH24060
      IF (NSLAVE.NE.0) GO TO 650                                        PH24070
      IF (URR) 720,620,640                                              PH24080
720   M=K+1                                                             PH24090
      N=I+1                                                             PH24100
      GO TO 650                                                         PH24110
C                   CELL K IS NOT EMPTY                                 PH24120
730   IF (AMX(K+1).GT.0.) GO TO 750                                     PH24130
C                   CELL ON RIGHT OF K IS EMPTY                         PH24140
      IF (U(K).LE.0.) GO TO 620                                         PH24150
C                   SHOULD MASS GO INTO EMPTY CELL                      PH24160
      IF (I.EQ.1) GO TO 740                                             PH24170
      IF (AIX(K-1).GT.ESESQ.OR.AIX(K).GT.ESESQ.OR.AMX(K-1)/(TAU(I-1)*DY(PH24180
     1J)).LT.SOLID.OR.AMX(K)/(TAU(I)*DY(J)).GT.SOLID) GO TO 630         PH24190
      GO TO 620                                                         PH24200
740   IF (AIX(K).GE.ESESQ.OR.AMX(K)/(TAU(I)*DY(J)).GE.RHOZ) GO TO 630   PH24210
      GO TO 620                                                         PH24220
750   IF (U(K).GT.0..AND.U(K+1).LT.0.) GO TO 760                        PH24230
      IF ((I+1).EQ.IMAX) GO TO 780                                      PH24240
      IF (AMX(K+2).GT.0..OR.U(K).GE.0..OR.U(K+1).GE.0.) GO TO 770       PH24250
C                   K AND K+1 NOT EMPTY BUT CELL K+2                    PH24260
C                   IS EMPTY. TEST FOR SLAVING K+1 TO K                 PH24270
      IF (AIX(K).GE.ESESQ.OR.AIX(K+1).GE.ESESQ.OR.AMX(K+1)/(TAU(I)*DY(J)PH24280
     1).GE.RHOZ) GO TO 710                                              PH24290
C                   YES, SLAVE K+1 TO K                                 PH24300
      NSLAVE=K+1                                                        PH24310
      N=I                                                               PH24320
      M=K                                                               PH24330
      GO TO 710                                                         PH24340
C                 *** $$REFLECTIVE$$ TREATMENT                          PH24350
760   UMK=U(K)*AMX(K)                                                   PH24360
      UMKP=U(K+1)*AMX(K+1)                                              PH24370
      WSA=TWOPDT*X(I)                                                   PH24380
      UOTK=UMK/TAU(I)                                                   PH24390
      UOTKP=UMKP/TAU(I+1)                                               PH24400
      WSB=UOTK+UOTKP                                                    PH24410
      AMMP=WSB*WSA                                                      PH24420
      UAMMP=(UOTK*U(K)+UOTKP*U(K+1))/WSB                                PH24430
      VAMMP=(UOTK*V(K)+UOTKP*V(K+1))/WSB                                PH24440
      SAVEK=AIX(K)+.5*(U(K)**2+V(K)**2)                                 PH24450
      EAMMP=(UOTK*SAVEK+UOTKP*(AIX(K+1)+.5*(U(K+1)**2+V(K+1)**2)))/WSB  PH24460
```

```
      GO TO 790                                                         PH24470
770   IF (I.EQ.1) GO TO 710                                             PH24480
780   IF (AMX(K-1).NE.0..OR.U(K+1).LE.0..OR.U(K).LE.0.) GO TO 710       PH24490
C                  SHOULD K BE SLAVED TO K+1                            PH24500
      IF (AIX(K).GE.ESESQ.OR.AIX(K+1).GE.ESESQ.OR.AMX(K)/(TAU(I)*DY(J)).PH24510
     1GE.RHOZ) GO TO 710                                                PH24520
C                  YES, SLAVE K TO K+1                                  PH24530
      NSLAVE=K                                                          PH24540
      N=I+1                                                             PH24550
      M=K+1                                                             PH24560
      GO TO 710                                                         PH24570
C                  WILL K BECOME MORE THAN EMPTY                        PH24580
790   IF (ABS(AMMP).LT.AMX(K)*ROEPS.AND.ABS(AMMP).LT.ROEPS*AMX(K+1)) AMMPH24590
     1P=0.                                                              PH24600
      WSOUT=0.                                                          PH24610
      WSA=0.                                                            PH24620
      WSB=0.                                                            PH24630
      IF (AMMP.GT.0.) GO TO 830                                         PH24640
      WSA=-AMMP                                                         PH24650
800   IF (AMPY.GT.0.) GO TO 840                                         PH24660
      WSA=WSA-AMPY                                                      PH24670
810   IF (GAMC(J).LT.0.) GO TO 850                                      PH24680
      WSA=WSA+GAMC(J)                                                   PH24690
820   IF (AMMY.LT.0.) GO TO 860                                         PH24700
      WSA=WSA+AMMY                                                      PH24710
      GO TO 870                                                         PH24720
830   WSB=AMMP                                                          PH24730
      GO TO 800                                                         PH24740
840   WSB=WSB+AMPY                                                      PH24750
      GO TO 810                                                         PH24760
850   WSOUT=-GAMC(J)                                                    PH24770
      GO TO 820                                                         PH24780
860   WSOUT=WSOUT-AMMY                                                  PH24790
870   DELM=WSA-WSB-WSOUT                                                PH24800
      IF (AMX(K)+DELM.GE.0.) GO TO 970                                  PH24810
C            *** INTERMEDIATE PRINT FOR CELLS OVER-EMPTYING.            PH24815
      IF (INTER.EQ.0) GO TO 880                                         PH24820
      WRITE (6,1290) I,J,AMX(K),DELM,AMMY,GAMC(J),AMPY,AMMP             PH24830
880   IF (WSOUT.GT.AMX(K)) GO TO 920                                    PH24840
C            *** OTHERWISE,MAKE WSB PLUS WSOUT EXACTLY                  PH24850
C            ***    EQUAL TO AMX(K)                                     PH24860
      WS=AMX(K)-WSOUT                                                   PH24870
      IF (AMMP.GT.0.) GO TO 900                                         PH24880
      AMPY=WS                                                           PH24890
890   DELM=WSA-WSOUT-AMMP-AMPY                                          PH24900
C            *** INTERMEDIATE PRINT FOR OVER-EMPTIED CELL AFTER         PH24902
C            *** RIGHT AND/OR TOP FLUXES ADJUSTED.                      PH24904
      IF(INTER.EQ.0) GO TO 970                                          PH24906
      WRITE(6,1350) AMX(K),DELM,AMMY,GAMC(J),AMPY,AMMP                  PH24908
      GO TO 970                                                         PH24910
900   IF (AMPY.GT.0.) GO TO 910                                         PH24920
      AMMP=WS                                                           PH24930
      GO TO 890                                                         PH24940
910   AMMP=WS/(AMMP+AMPY)*AMMP                                          PH24950
      AMPY=WS-AMMP                                                      PH24960
      GO TO 890                                                         PH24970
C                  *** CELL OVER-EMPTIED DOWN OR LEFT. PUT IT BACK.     PH24980
```

```
920   IF (AMMP.LT.0.) GO TO 930                                        PH24990
      AMMP=0.                                                          PH25000
      AMUR=0.                                                          PH25010
      AMVR=0.                                                          PH25020
      DELER=0.                                                         PH25030
930   IF (AMPY.LT.0.) GO TO 940                                        PH25040
      AMPY=0.                                                          PH25050
      AMUT=0.                                                          PH25060
      AMVT=0.                                                          PH25070
      DELET=0.                                                         PH25080
940   MASS=AMX(K)                                                      PH25090
      UMOM=MASS*U(K)                                                   PH25100
      VMOM=MASS*V(K)                                                   PH25110
      ENGY=MASS*(.5*U(K)**2+.5*V(K)**2+AIX(K))                         PH25120
      MASS=MASS-AMMP                                                   PH25130
      UMOM=UMOM-AMMP*U(K+1)                                            PH25140
      VMOM=VMOM-AMMP*V(K+1)                                            PH25150
      ENGY=ENGY-AMMP*(.5*U(K+1)**2+.5*V(K+1)**2+AIX(K+1))              PH25160
      MASS=MASS-AMPY                                                   PH25170
      UMOM=UMOM-AMPY*U(L)                                              PH25180
      VMOM=VMOM-AMPY*V(L)                                              PH25190
      ENGY=ENGY-AMPY*(.5*U(L)**2+.5*V(L)**2+AIX(L))                    PH25200
      MASS=MASS+AMMY                                                   PH25210
      UMOM=UMOM+AMMU                                                   PH25220
      VMOM=VMOM+AMMV                                                   PH25230
      ENGY=ENGY+DELEB                                                  PH25240
      MASS=MASS+GAMC(J)                                                PH25250
      UMOM=UMOM+FLEFT(J)                                               PH25260
      VMOM=VMOM+YAMC(J)                                                PH25270
      ENGY=ENGY+SIGC(J)                                                PH25280
      WSA=-AMIN1(0.,GAMC(J))/WSOUT                                     PH25290
      WSB=-AMIN1(0.,AMMY)/WSOUT                                        PH25300
      LB=K-IMAX                                                        PH25310
      IF (LB.LT.0) WSA=1.                                              PH25320
      IF (LB.LT.0) GO TO 950                                           PH25330
      IF (AMMY.EQ.0..OR.WSB.EQ.0.) GO TO 950                           PH25340
      WSC=AMX(LB)+WSB*MASS                                             PH25350
      WSD=AIX(LB)+.5*(U(LB)**2+V(LB)**2)                               PH25360
      U(LB)=(AMX(LB)*U(LB)+WSB*UMOM)/WSC                               PH25370
      V(LB)=(AMX(LB)*V(LB)+WSB*VMOM)/WSC                               PH25380
      AIX(LB)=(AMX(LB)*WSD+WSB*ENGY)/WSC-.5*(U(LB)**2+V(LB)**2)        PH25390
      AMX(LB)=WSC                                                      PH25400
950   IF (GAMC(J).EQ.0..OR.WSA.EQ.0.) GO TO 960                        PH25410
      WSC=AMX(K-1)+WSA*MASS                                            PH25420
      WSD=AIX(K-1)+.5*(U(K-1)**2+V(K-1)**2)                            PH25430
      U(K-1)=(AMX(K-1)*U(K-1)+WSA*UMOM)/WSC                            PH25440
      V(K-1)=(AMX(K-1)*V(K-1)+WSA*VMOM)/WSC                            PH25450
      AIX(K-1)=(AMX(K-1)*WSD+WSA*ENGY)/WSC-.5*(U(K-1)**2+V(K-1)**2)    PH25460
      AMX(K-1)=WSC                                                     PH25470
C            *** INTERMEDIATE PRINT FOR OVER-EMPTIED CELLS AFTER       PH25472
C                MASS PUT BACK.                                        PH25474
960   IF (INTER.EQ.0) GO TO 1100                                       PH25480
      WRITE(6,1360) AMX(K), DELM, AMMY, GAMC(J), AMPY, AMMP            PH25490
      GO TO 1100                                                       PH25500
970   IF (AMPY.EQ.0.) GO TO 960                                        PH25510
C                  CALCULATE ENERGY AND MOMENTUM FLUX AT TOP           PH25520
      AMUT=AMPY*UAMPY                                                  PH25530
```

```
      AMVT=AMPY*VAMPY                                                   PH25540
      DELET=AMPY*EAMPY                                                  PH25550
C                   IS THIS AT TOP BOUNDARY                             PH25560
      IF (J.NE.JMAX) GO TO 990                                          PH25570
C                   YES, TOP. ADJUST ENERGY.                            PH25580
      ETH=ETH-DELET                                                     PH25590
      EMOT=EMOT+DELET                                                   PH25600
      TOPM=TOPM+AMPY                                                    PH25610
      TOPMV=TOPMV+AMVT                                                  PH25620
      TOPMU=TOPMU+AMUT                                                  PH25630
C                   IS AMPY LARGE ENOUGH TO TRIGGER REZONE              PH25640
      IF (AMPY/(TAU(I)*DY(J)).GE.VT) REZ=1.                             PH25650
      GO TO 990                                                         PH25660
C                   AMPY=0. SET MOMENTUM AND ENERGY FLUX=0.             PH25670
980   AMUT=0.                                                           PH25680
      AMVT=0.                                                           PH25690
      DELET=0.                                                          PH25700
990   IF (AMMP.EQ.0.) GO TO 1000                                        PH25710
C                   CALCULATE ENERGY + MOMENTUM FLUX AT RIGHT           PH25720
      AMUR=AMMP*UAMMP                                                   PH25730
      AMVR=AMMP*VAMMP                                                   PH25740
      DELER=AMMP*EAMMP                                                  PH25750
C                   IS THIS AT RIGHT BOUNDARY                           PH25760
      IF (I.NE.IMAX) GO TO 1010                                         PH25770
C                   YES, RIGHT. ADJUST ENERGY.                          PH25780
      ETH=ETH-DELER                                                     PH25790
      EMOR=EMOR+DELER                                                   PH25800
      RTM=RTM+AMMP                                                      PH25810
      RTMV=RTMV+AMVR                                                    PH25820
      RTMU=RTMU+AMUR                                                    PH25830
C                   IS AMMP LARGE ENOUGH TO TRIGGER REZONE              PH25840
      IF (AMMP/(TAU(I)*DY(J)).GE.VT) REZ=1.                             PH25850
      GO TO 1010                                                        PH25860
C                   AMMP=0. SET MOMENTUM AND ENERGY FLUX=0.             PH25870
1000  AMUR=0.                                                           PH25880
      AMVR=0.                                                           PH25890
      DELER=0.                                                          PH25900
C                   REPARTITION ENERGY + MOMENTUM                       PH25910
1010  IF (DELM.EQ.0.) GO TO 1080                                        PH25920
1020  WSA=.5*(U(K)**2+V(K)**2)                                          PH25930
      SIGMU=-AMUT-AMUR+AMMU+FLEFT(J)                                    PH25940
      SIGMV=-AMVT-AMVR+AMMV+YAMC(J)                                     PH25950
      WS=DELM+AMX(K)                                                    PH25960
      IF (WS.LE.0.) GO TO 1070                                          PH25970
      UNEW=(SIGMU+AMX(K)*U(K))/WS                                       PH25980
      DELU=UNEW-U(K)                                                    PH25990
      IF (ABS(DELU).LT.UMIN) GO TO 1030                                 PH26000
      U(K)=UNEW                                                         PH26010
1030  VNEW=(SIGMV+AMX(K)*V(K))/WS                                       PH26020
      DELV=VNEW-V(K)                                                    PH26030
      IF (ABS(DELV).LT.UMIN) GO TO 1040                                 PH26040
      V(K)=VNEW                                                         PH26050
1040  WSB=-DELET-DELER+DELEB+SIGC(J)                                    PH26060
      SIENEW=((AIX(K)+WSA)*AMX(K)+WSB)/WS-.5*(U(K)**2+V(K)**2)          PH26070
      DELI=SIENEW-AIX(K)                                                PH26080
      IF (ABS(DELI).GT.SIEMIN) GO TO 1050                               PH26090
C           *** SUME SUMS ENERGY FLUXES TOO SMALL TO USE. SUME IS       PH26092
```

```
C                 SUBTRACTED FROM ETH AT THE END OF THIS ROUTINE.          PH26094
      SUME=SUME+DELI*WS                                                    PH26100
      GO TO 1060                                                           PH26110
 1050 AIX(K)=SIENEW                                                        PH26120
 1060 AMX(K)=WS                                                            PH26130
      GO TO 1090                                                           PH26140
 1070 AMX(K)=0.                                                            PH26150
      AIX(K)=0.                                                            PH26160
      U(K)=0.                                                              PH26170
      V(K)=0.                                                              PH26180
      GO TO 1100                                                           PH26190
C                   DELM=0. BUT IS THERE INDIVIDUAL FLOW                   PH26200
 1080 IF (AMMP.NE.0.) GO TO 1020                                           PH26210
      IF (AMPY.NE.0.) GO TO 1020                                           PH26220
      IF (AMMY.NE.0.) GO TO 1020                                           PH26230
      IF (GAMC(J).NE.0.) GO TO 1020                                        PH26240
 1090 IF (I.NE.I1) GO TO 1100                                              PH26250
      IF (U(K).NE.0..OR.V(K).NE.0..OR.AIX(K).NE.0.) NRC=1                  PH26260
C            *** SPECIAL INTERMEDIATE PRINT FOR CHECKING ENERGY            PH26262
C                 CONSERVATION - PRINTS ONLY IF INTER = 7 IN INPUT DECK.   PH26264
 1100 IF (INTER.NE.7) GO TO 1130                                           PH26270
      ENERGY=DELER+DELET-SIGC(J)                                           PH26280
      DO 1110 NN=1,JMAX                                                    PH26290
      ENERGY=ENERGY+SIGC(NN)                                               PH26300
 1110 CONTINUE                                                             PH26310
      DO 1120 LJD=2,KMAX                                                   PH26320
      ENERGY=ENERGY+AMX(LJD)*(AIX(LJD)+.5*(U(LJD)**2+V(LJD)**2))           PH26330
 1120 CONTINUE                                                             PH26340
      WRITE (6,1300) I,J,ENERGY                                            PH26350
      WRITE (6,1310) AMPY,AMMP,AMMY,GAMC(J)                                PH26360
      WRITE (6,1320) DELET,DELER,DELEB,SIGC(J)                             PH26370
 1130 CONTINUE                                                             PH26380
      GAMC(J)=AMMP                                                         PH26390
      FLEFT(J)=AMUR                                                        PH26400
      YAMC(J)=AMVR                                                         PH26410
      SIGC(J)=DELER                                                        PH26420
      AMMY=AMPY                                                            PH26430
      AMMU=AMUT                                                            PH26440
      AMMV=AMVT                                                            PH26450
      DELEB=DELET                                                          PH26460
C
C            *** END OF J-LOOP.                                            PH26465
C
 1140 K=K+IMAX                                                             PH26470
      LL=K-IMAX                                                            PH26480
      IF (U(LL).NE.0..OR.V(LL).NE.0..OR.AIX(LL).NE.0.) NRT=1               PH26490
C
C            *** END OF I-LOOP.                                            PH26495
C
 1150 CONTINUE                                                             PH26500
C            *** ADVANCE ACTIVE GRID.                                      PH26505
      I1=I1+NRC                                                            PH26510
      I2=I2+NRT                                                            PH26520
      IF (IMAX-I1) 1160,1170,1180                                          PH26530
 1160 I1=IMAX                                                              PH26540
 1170 CONTINUE                                                             PH26550
 1180 IF (JMAX-I2) 1190,1200,1210                                          PH26560
```

```
1190  I2=JMAX                                                          PH26570
1200  CONTINUE                                                         PH26580
1210  GO TO 1230                                                       PH26590
C                    *** NEGATIVE MASS                                 PH26600
1220  NK=315                                                           PH26610
      NR=9                                                             PH26620
      CALL ERROR                                                       PH26630
1230  SUM=0.0                                                          PH26640
C            *** EVAPORATE LOW-DENSE CELLS ON BASIS OF EVAP, INPUT     PH26642
C                PARAMETER.                                            PH26644
      DO 1280 I=1,I1                                                   PH26650
      K=I+1                                                            PH26660
      DO 1270 J=1,I2                                                   PH26670
      IF (AMX(K).EQ.0.) GO TO 1270                                     PH26680
      IF (AMX(K)/(TAU(1)*DY(J)).GT.EVAP*RHINI) GO TO 1250              PH26690
      WS=(U(K)**2+V(K)**2)/2.0                                         PH26700
      EVAPM=EVAPM+AMX(K)                                               PH26710
      WS=AMX(K)*(AIX(K)+WS)                                            PH26720
      EVAPEN=EVAPEN+WS                                                 PH26730
      ETH=ETH-WS                                                       PH26740
      EVAPMU=EVAPMU+AMX(K)*U(K)                                        PH26750
      EVAPMV=EVAPMV+AMX(K)*V(K)                                        PH26760
C            *** INTERMEDIATE PRINT FOR CELLS EVAPORATED.              PH26765
      IF (INTER.EQ.0) GO TO 1240                                       PH26770
      WRITE(6,1340) I,J,AMX(K),AIX(K),U(K),V(K)                        PH26780
1240  AMX(K)=0.0                                                       PH26790
      AIX(K)=0.0                                                       PH26800
      P(K)=0.0                                                         PH26810
      U(K)=0.0                                                         PH26820
      V(K)=0.0                                                         PH26830
      GO TO 1270                                                       PH26840
C            *** SET NEGATIVE INTERNAL ENERGIES TO ZERO WHEN SN=0.     PH26842
C                (INPUT PARAMETER).                                    PH26844
1250  IF (AIX(K).GE.0..OR.SN.GT.0.) GO TO 1270                         PH26850
C            *** SUM SUMS NEGATIVE INTERNAL ENERGY SET TO ZERO.        PH26855
      SUM=SUM+AIX(K)*AMX(K)                                            PH26860
C            *** INTERMEDIATE PRINT FOR CELLS WHOSE NEGATIVE           PH26862
C                INTERNAL ENERGY IS SET TO ZERO.                       PH26864
      IF (INTER.EQ.0) GO TO 1260                                       PH26870
      WRITE (6,1330) I,J,AMX(K),AIX(K),U(K),V(K)                       PH26880
1260  AIX(K)=0.                                                        PH26890
1270  K=K+IMAX                                                         PH26900
1280  CONTINUE                                                         PH26910
C            *** ETH = THEORETICAL ENERGY SUM, USED IN EDIT FOR        PH26912
C                      ENERGY CHECK.                                   PH26914
C            *** EZPH2 = ENERGY SET TO ZERO  IN PH2 SINCE  TIME=0.     PH26916
C            *** SUM  = NEGATIVE INTERNAL ENERGY SET TO ZERO ON THIS   PH26917
C                       CYCLE.                                         PH26918
C            *** SUME = SUM OF THE ENERGY FLUXES IGNORED ON THIS CYCLE. PH26919
      ETH=ETH-SUM-SUME                                                 PH26920
      EZPH2=EZPH2-SUME-SUM                                             PH26930
      RETURN                                                           PH26940
C                                                                      PH26950
1290  FORMAT (5H NEGM,I3,I4,4H  M=,1PE14.7,6H DELM=,1PE14.7,6H  BOT=,1PEPH26960
     114.7,7H  LEFT=,1PE14.7,6H  TOP=,1PE14.7,5H  RT=,1PE14.7)         PH26970
1300  FORMAT (5H  I= I3,6X,5H  J= I3,6X,9H  ENERGY=1PE15.8)            PH26980
1310  FORMAT (7H  AMPY=1PE15.8,6X,6H AMMP=1PE15.8,6X,6H AMMY=1PE15.8,9H PH26990
```

```
     1GAMC(J)=1PE15.8)                                                  PH27000
 1320 FORMAT (7H DELET=1PE15.8,6X,6HDELER=1PE15.8,6X,6HDELEB=1PE15.8,9H PH27010
     1SIGC(J)=1PE15.8)                                                  PH27020
 1330 FORMAT (4H PH2,2I4,4H  M=,1PE15.8,6H  SIE=,1PE15.8,4H  U=,1PE15.8,PH27030
     14H  V=,1PE15.8,18H   SIE SET TO ZERO)                             PH27040
 1340 FORMAT (4H PH2,2I4,4H  M=,1PE15.8,6H  SIE=,1PE15.8,4H  U=,1PE15.8,PH27042
     14H  V=,1PE15.8,19H    CELL EVAPORATED)                            PH27044
 1350 FORMAT (12H ADJUST FLUX,4H  M=,1PE14.7,6H DELM=,1PE14.7,6H  BOT=, PH27045
     11PE14.7,7H  LEFT=,1PE14.7,6H  TOP=,1PE14.7,5H  RT=,1PE14.7)       PH27046
 1360 FORMAT (12H ADJUST MASS,4H  M=,1PE14.7,6H DELM=,1PE14.7,6H  BOT=, PH27047
     11PE14.7,7H  LEFT=,1PE14.7,6H  TOP=,1PE14.7,5H  RT=,1PE14.7)       PH27048
      END                                                               PH27050-
```

```
      SUBROUTINE REZONE                                                 REZ  10
C     ..................................................................REZ  20
C                                                                       REZ  30
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,      REZ  40
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,                REZ  50
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),   REZ  60
     3          GAMC(102),                                             REZ  70
     4          PK(15),   Z(150)   ,                                   REZ  80
     5          XP(26,51),YP(26,51),                                   REZ  90
     6          PL(204)  ,UL(204)  ,PR(204)  ,                         REZ 100
     7           RSN(52),   RST(52),                                   REZ 110
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,               REZ 120
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,               REZ 130
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)      REZ 140
C            *** DIMENSIONED ARRAYS                                     REZ 150
C            *** Z-BLOCK IS SAVED ON TAPE.                              REZ 160
      COMMON     Z                                                     REZ 170
      COMMON    PK                                                     REZ 180
      COMMON  YY,      XX                                              REZ 190
      COMMON  DDX,     DDY                                             REZ 200
      COMMON  AMX,     AIX,     U,     V,     P                       REZ 210
      COMMON  TAU,     JPM                                             REZ 220
      COMMON  UL ,     PL                                              REZ 230
      COMMON  XP ,     YP,      CMXP, CMYP                             REZ 240
C            *** NON-DIMENSIONED VARIABLES                              REZ 250
      COMMON         AID    ,AMMV   ,AMMY   ,AMPY   ,AMUR   ,AMUT   ,AMVR   ,  REZ 260
     1AMVT   ,DELEB ,DELER ,DELET ,DELM   ,DTODX ,DXYMIN,EAMMP ,EAMPY ,  REZ 270
     2E      ,ERDUMP,I     ,I3    ,IWS    ,J      ,K      ,KA     ,KB     ,  REZ 280
     3LL     ,MD    ,ME    ,MZT   ,NERR   ,NK     ,NPRINT,                REZ 290
     4NR     ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR    ,SNT    ,STR    ,SOLID ,  REZ 300
     5SUM    ,TESTRH,TWOPI ,URR   ,WS     ,WSA    ,WSB    ,WSC    ,WFLAGF,  REZ 310
     6WFLAGL,WFLAGP                                                     REZ 320
C                                                                       REZ 330
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE              REZ 340
C                X(0), Y(0), DX(0), DY(0)                               REZ 350
C                                                                       REZ 360
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))                         REZ 370
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))                     REZ 380
C                                                                       REZ 390
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY                      REZ 400
C                                                                       REZ 410
      EQUIVALENCE          (UL,FLEFT),         (UL(103),YAMC),          REZ 420
     1                     (PL,GAMC,PR),       (PL(103),SIGC)           REZ 430
C                                                                       REZ 440
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY                      REZ 450
C                                                                       REZ 460
      EQUIVALENCE          (UL,RSN),                                   REZ 470
     1                     (PL,RST),           (P,UK),                 REZ 480
     2                     (P(157),VK),        (P(313),SNB),           REZ 490
     3                     (P(365),STB),       (P(417),RHO)            REZ 500
C                                                                       REZ 510
C            *** SPECIAL EQUIVALENCES FOR EDIT                          REZ 520
C                                                                       REZ 530
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)                             REZ 540
C                                                                       REZ 550
C            *** Z-STORAGE EQUIVALENCES                                 REZ 560
C                                                                       REZ 570
      EQUIVALENCE                        (Z(  1),PROB  ),(Z(  2),CYCLE ), REZ 580
```

```
     1(Z(  3),DT    ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  REZ 590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ),(Z(  9),TOPMU ),(Z( 10),RTMU  ),  REZ 600
     3(Z( 11),STK1  ),(Z( 12),NUMREZ),(Z( 13),ETH   ),(Z( 14),UN14  ),  REZ 610
     4(Z( 15),RHINIT),(Z( 16),PROJI ),(Z( 17),UN17  ),(Z( 18),XMAX  ),  REZ 620
     5(Z( 19),NZ    ),(Z( 20),NREZ  ),(Z( 21),AMDM  ),(Z( 22),UVMAX ),  REZ 630
     6(Z( 23),UN23  ),(Z( 24),DMIN  ),(Z( 25),JSTR  ),(Z( 26),DTNA  ),  REZ 640
     7(Z( 27),CVIS  ),(Z( 28),STK2  ),(Z( 29),STEZ  ),(Z( 30),NC    ),  REZ 650
     8(Z( 31),UN31  ),(Z( 32),NRC   ),(Z( 33),IMAX  ),(Z( 34),IMAXA ),  REZ 660
     9(Z( 35),JMAX  ),(Z( 36),JMAXA ),(Z( 37),KMAX  ),(Z( 38),KMAXA )  REZ 670
      EQUIVALENCE.                                                      REZ 680
     1(Z( 39),BOTM  ),(Z( 40),BOTMV ),(Z( 41),NUMSPT),(Z( 42),CZERO ),  REZ 690
     2(Z( 43),NUMSCA),(Z( 44) PRLIM ),(Z( 45),PRDELT),(Z( 46),PRFACT)  REZ 700
      EQUIVALENCE                                                       REZ 710
     1(Z( 47),I1    ),(Z( 48),I2    ),(Z( 49),IPCYCL),(Z( 50),TSTOP ),  REZ 720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ),(Z( 53),N3    ),(Z( 54),IVARDY),  REZ 730
     3(Z( 55),VT    ),(Z( 56),N6    ),(Z( 57),RTM   ),(Z( 58),RTMV  ),  REZ 740
     4(Z( 59),UN59  ),(Z( 60),N10   ),(Z( 61),N11   ),(Z( 62),GAMMA ),  REZ 750
     5(Z( 63),TOPM  ),(Z( 64),BOTMU ),(Z( 65),SN    ),(Z( 66),TOPMV ),  REZ 760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP),(Z( 69),PRXRT ),(Z( 70),CYCPH3),  REZ 770
     7(Z( 71),REZFCT),(Z( 72),TARGI ),(Z( 73),PROJU ),(Z( 74),BBOUND),  REZ 780
     8(Z( 75),EVAP  ),(Z( 76),ECK   ),(Z( 77),NECYCL),(Z( 78),II    ),  REZ 790
     9(Z( 79),JJ    ),(Z( 80),NMP   ),(Z( 81),Y2    ),(Z( 82),EZPH1 )  REZ 800
      EQUIVALENCE                                                       REZ 810
     1(Z( 83),IVARDX),(Z( 84),T     ),(Z( 85),NMPMAX),(Z( 86),PMIN  ),  REZ 820
     2(Z( 87),INTER ),(Z( 88),TAYBOT),(Z( 89),TAYTOP),(Z( 90),IEMAP ),  REZ 830
     3(Z( 91),MC    ),(Z( 92),MR    ),(Z( 93),MZ    ),(Z( 94),MB    )  REZ 840
      EQUIVALENCE                                                       REZ 850
     1(Z( 95),REZ   ),(Z( 96),NODUMP),(Z( 97),UN97  ),(Z( 98),UN98  ),  REZ 860
     2(Z( 99),UN99  ),(Z(100),EVAPM ),(Z(101),EVAPEN),(Z(102),EVAPMU),  REZ 870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ),(Z(105),SNL   ),(Z(106),STL   ),  REZ 880
     4(Z(107),TAXRT ),(Z(108),IDNMAP),(Z(109),IPRMAP),(Z(110),ROEPS ),  REZ 890
     5(Z(111),RHINI ),(Z(112),VINI  ),(Z(113),FINAL ),(Z(114),IVMAP ),  REZ 900
     6(Z(115),RHOZ  ),(Z(116),ESA   ),(Z(117),ESEZ  ),(Z(118),ESB   ),  REZ 910
     7(Z(119),ESCAPA),(Z(120),ESESP ),(Z(121),ESESQ ),(Z(122),ESES  ),  REZ 920
     8(Z(123),ESALPH),(Z(124),ESBETA),(Z(125),ESCAPB),(Z(126),IUMAP ),  REZ 930
     9(Z(127),SS1   ),(Z(128),SS2   ),(Z(129),UMIN  ),(Z(130),SS4   )  REZ 940
      EQUIVALENCE                                                       REZ 950
     1(Z(131),FRTIME),(Z(132),EOR   ),(Z(133),EOT   ),(Z(134),EOB   ),  REZ 960
     2(Z(135),EMOR  ),(Z(136),DXF   ),(Z(137),DYF   ),(Z(138),RHOMIN),  REZ 970
     3(Z(139),STAB),  (Z(140),XIENRG),  (Z(141),XKENRG),  (Z(142),XTENRG),REZ 980
     4(Z(143),STT   ),(Z(144),DTMIN ),(Z(145),TRNSFC),(Z(146),EMOT  ),  REZ 990
     5(Z(147),JPROJ ),(Z(148),CNAUT ),(Z(149),BBAR  ),(Z(150),EMOB  )  REZ1000
C                                                                       REZ1010
C     ..................................................................REZ1020
C                                                                       REZ1030
C     END OF COMMON                                                     REZ1040
C                                                                       REZ1050
C     ..................................................................REZ1060
C                                                                       REZ1070
C            *** INITIALIZE P-STORAGE. CDT CALLED AGAIN AND PRESSURES   REZ1072
C                RECALCULATED AFTER GRID REZONED AND BEFORF PH1,PH3     REZ1074
C                AND PH2 ARE CALLED.                                    REZ1076

      DO 10 K=2,KMAX                                                    REZ1080
      P(K)=0.                                                           REZ1090
10    CONTINUE                                                          REZ1100
      NJMAX=JMAX/2                                                      REZ1110
```

```
      IF (IMAX.EQ.1) GO TO 20                                           REZ1120
      NIMAX=IMAX/2                                                      REZ1130
      GO TO 70                                                          REZ1140
C            *** 1-D                                                    REZ1145
20    IMAX=2                                                            REZ1150
      NIMAX=1                                                           REZ1160
      K=2                                                               REZ1170
      L=2*JMAX+2                                                        REZ1180
C            *** STORE PROPERTIES TEMPORARILY IN UNUSED PART OF ARRAYS. REZ1185
      DO 30 J=1,JMAX                                                    REZ1190
      AMX(L)=AMX(K)                                                     REZ1200
      U(L)=U(K)                                                         REZ1210
      V(L)=V(K)                                                         REZ1220
      AIX(L)=AIX(K)                                                     REZ1230
      K=K+1                                                             REZ1240
      L=L+1                                                             REZ1250
30    CONTINUE                                                          REZ1260
      K=2                                                               REZ1270
      L=2*JMAX+2                                                        REZ1280
C            *** ADD ANOTHER COLUMN OF CELLS. EACH CELL IN NEW COLUMN   REZ1282
C                WILL HAVE SAME VELOCITIES AND SIE AND 3 TIMES THE      REZ1284
C                MASS OF AXIS CELL.                                     REZ1286
      DO 50 J=1,JMAX                                                    REZ1290
      DO 40 I=1,2                                                       REZ1300
      AMX(K)=AMX(L)                                                     REZ1310
      U(K)=U(L)                                                         REZ1320
      V(K)=V(L)                                                         REZ1330
      AIX(K)=AIX(L)                                                     REZ1340
      K=K+1                                                             REZ1350
      AMX(L)=3.*AMX(L)                                                  REZ1360
40    CONTINUE                                                          REZ1370
50    L=L+1                                                             REZ1380
      L=2*JMAX+1                                                        REZ1390
C            *** ADJUST ETH BY ADDING ENERGY OF CELLS IN NEW COLUMN.    REZ1395
      DO 60 K=3,L,2                                                     REZ1400
      ETH=ETH+AMX(K)*(AIX(K)+(V(K)**2)/2.)                              REZ1410
60    CONTINUE                                                          REZ1420
70    DO 120 J=1,NJMAX                                                  REZ1430
      K=(J-1)*NIMAX+2                                                   REZ1440
      L=(J-1)*2*IMAX+2                                                  REZ1450
      DO 110 I=1,NIMAX                                                  REZ1460
      M=L+IMAX                                                          REZ1470
C            *** SUM MASS OF FOUR CELLS TO BE MADE INTO ONE CELL.       REZ1475
      WSA=AMX(L)+AMX(M)+AMX(L+1)+AMX(M+1)                               REZ1480
      IF (WSA.EQ.0.) GO TO 80                                           REZ1490
C            *** SUM KINETIC ENERGY OF FOUR CELLS.                      REZ1495
      WSB=AMX(L)*(U(L)**2+V(L)**2)+AMX(M)*(U(M)**2+V(M)**2)+AMX(L+1)*(U(REZ1500
     1L+1)**2+V(L+1)**2)+AMX(M+1)*(U(M+1)**2+V(M+1)**2)                 REZ1510
C            *** COMPUTE VELOCITIES OF NEW CELL FROM VELOCITIES OF      REZ1512
C                THE FOUR CELLS.                                        REZ1514
      U(K)=(U(L)*AMX(L)+U(M)*AMX(M)+U(L+1)*AMX(L+1)+U(M+1)*AMX(M+1))/WSAREZ1520
      V(K)=(V(L)*AMX(L)+V(M)*AMX(M)+V(L+1)*AMX(L+1)+V(M+1)*AMX(M+1))/WSAREZ1530
C            *** COMPUTE INTERNAL ENERGY OF NEW CELL.                   REZ1535
      AIX(K)=AIX(L)*AMX(L)+AIX(M)*AMX(M)+AIX(L+1)*AMX(L+1)+AMX(M+1)*AIX(REZ1540
     1M+1)                                                              REZ1550
      AMX(K)=WSA                                                        REZ1560
      WS=U(K)**2+V(K)**2                                                REZ1570
```

```
      E=AIX(K)+WSB/2.0                                                  REZ1580
C           *** COMPUTE SIE OF NEW CELL.                                REZ1585
      AIX(K)=E/AMX(K)-.5*WS                                             REZ1590
      IF (K-2) 100,100,90                                               REZ1600
C           *** NEW CELL EMPTY.                                         REZ1605
80    AMX(K)=0.                                                         REZ1610
      AIX(K)=0.                                                         REZ1620
      U(K)=0.                                                           REZ1630
      V(K)=0.                                                           REZ1640
C           *** INITIALIZE STORAGE OF CELL QUANTIES OF OLD GRID.        REZ1650
90    AMX(L)=0.0                                                        REZ1660
      U(L)=0.0                                                          REZ1670
      V(L)=0.0                                                          REZ1680
      AIX(L)=0.0                                                        REZ1690
      AMX(M)=0.0                                                        REZ1700
      U(M)=0.0                                                          REZ1710
      V(M)=0.0                                                          REZ1720
      AIX(M)=0.0                                                        REZ1730
      AMX(L+1)=0.0                                                      REZ1740
      U(L+1)=0.0                                                        REZ1750
      V(L+1)=0.0                                                        REZ1760
      AIX(L+1)=0.0                                                      REZ1770
      AMX(M+1)=0.0                                                      REZ1780
      U(M+1)=0.0                                                        REZ1790
      V(M+1)=0.0                                                        REZ1800
      AIX(M+1)=0.0                                                      REZ1810
100   K=K+1                                                             REZ1820
      L=L+2                                                             REZ1830
C           *** END OF I-LOOP                                           REZ1835
110   CONTINUE                                                          REZ1840
C           *** END OF J-LOOP                                           REZ1845
120   CONTINUE                                                          REZ1850
C           *** OLD PART OF ENLARGED GRID HAS NOW BEEN REZONED.         REZ1852
C               PROPERTIES OF NEW PART OF GRID WILL BE ASSIGNED         REZ1854
C               BELOW.                                                  REZ1856
C
C           *** CALCULATE NEW DY'S UP TO  EDGE OF OLD GRID BY           REZ1860
C               COMBINING THE OLD DY'S . CALCULATE NEW Y'S FROM         REZ1862
C               THE NEW DY'S.                                           REZ1864
      DO 130 J=1,NJMAX                                                  REZ1870
      DY(J)=DY(2*J-1)+DY(2*J)                                           REZ1880
      Y(J)=Y(J-1)+DY(J)                                                 REZ1890
130   CONTINUE                                                          REZ1900
C           *** ASSIGN THE VALUE OF THE LAST DY CALCULATED ABOVE        REZ1902
C               TO ALL CELLS ABOVE THE OLD GRID.                        REZ1904
      NJMAX1=NJMAX+1                                                    REZ1910
      DO 140 J=NJMAX1,JMAX                                              REZ1920
      DY(J)=DY(NJMAX)                                                   REZ1930
      Y(J)=Y(J-1)+DY(J)                                                 REZ1940
140   CONTINUE                                                          REZ1950
C           *** IMAX IS  SET TO 2 IF DOING A 1-D PROBLEM                REZ1955
      IF (IMAX.EQ.2) DX(2)=DX(1)                                        REZ1960
      DX(1)=DX(1)+DX(2)                                                 REZ1970
      X(1)=DX(1)                                                        REZ1980
      WS=X(1)**2                                                        REZ1990
      TAU(1)=PIDY*WS                                                    REZ2000
C           *** ARE YOU DOING A 1-D PROBLEM                             REZ2005
```

```
      IF (IMAX.EQ.2) GO TO 300                                          REZ2010
C           *** CALCULATE NEW DX'S OUT TO EDGE OF OLD GRID BY           REZ2012
C               COMBINING OLD DX'S. CALCULATE NEW X'S AND CELL-FACE     REZ2014
C               AREAS FROM THE NEW DX'S.                                REZ2016
      DO 150 I=2,NIMAX                                                  REZ2020
      DX(I)=DX(2*I-1)+DX(2*I)                                           REZ2030
      X(I)=X(I-1)+DX(I)                                                 REZ2040
      WSA=X(I)**2                                                       REZ2050
      TAU(I)=PIDY*(WSA-WS)                                              REZ2060
      WS=WSA                                                            REZ2070
150   CONTINUE                                                          REZ2080
C           *** ASSIGN THE VALUE OF THE LAST DX CALCULATED ABOVE        REZ2082
C               TO ALL CELLS TO THE RIGHT OF THE OLD GRID.              REZ2084
      NIMAX1=NIMAX+1                                                    REZ2090
      DO 160 I=NIMAX1,IMAX                                              REZ2100
      DX(I)=DX(NIMAX)                                                   REZ2110
      X(I)=X(I-1)+DX(I)                                                 REZ2120
      WSA=X(I)**2                                                       REZ2130
      TAU(I)=PIDY*(WSA-WS)                                              REZ2140
      WS=WSA                                                            REZ2150
160   CONTINUE                                                          REZ2160
C                                                                       REZ2170
C           *** INITIALIZE CELL BOUNDARIES                              REZ2180
C                                                                       REZ2190
      JPB=0                                                             REZ2200
      JPA=0                                                             REZ2210
      JTB=0                                                             REZ2220
      JTA=0                                                             REZ2230
      IPRT=0                                                            REZ2240
      ITRT=0                                                            REZ2250
C                                                                       REZ2260
      IF (PRYTOP.LE.Y(NJMAX).AND.PRXRT.LE.X(NIMAX).AND.TAYTOP.LE.Y(NJMAXREZ2270
     1).AND.TAXRT.LE.X(NIMAX)) GO TO 300                                REZ2280
C                                                                       REZ2290
C           *** COMPUTE JPB, JPA - BOTTOM AND TOP CELL BOUNDARIES OF    REZ2300
C               PROJECTILE                                              REZ2310
C                                                                       REZ2320
      IF (PRYBOT.LT.0..OR.(PRYTOP.LE.Y(NJMAX).AND.PRXRT.LE.X(NIMAX))) GOREZ2330
     1 TO 230                                                           REZ2340
      J=0                                                               REZ2350
      IF (PRYBOT.EQ.0.) GO TO 180                                       REZ2360
      DYSUM=0.                                                          REZ2370
      DO 170 J=1,JMAX                                                   REZ2380
      DYSUM=DYSUM+DY(J)                                                 REZ2390
      IF (PRYBOT.GE.DYSUM-.5*DY(J).AND.PRYBOT.LT.DYSUM+.5*DY(J+1)) GO TOREZ2400
     1 180                                                              REZ2410
170   CONTINUE                                                          REZ2420
      GO TO 230                                                         REZ2430
180   JPB=MIN0(J+1,JMAX)                                                REZ2440
      DO 190 J=JPB,JMAX                                                 REZ2450
      DYSUM=DYSUM+DY(J)                                                 REZ2460
      IF (PRYTOP.GE.DYSUM-.5*DY(J).AND.PRYTOP.LT.DYSUM+.5*DY(J+1)) GO TOREZ2470
     1 200                                                              REZ2480
190   CONTINUE                                                          REZ2490
200   JPA=J                                                             REZ2500
C                                                                       REZ2510
C           *** COMPUTE IPRT - RIGHT CELL BOUNDARY OF PROJECTILE        REZ2520
```

```
C                                                                       REZ2530
      DXSUM=0.                                                          REZ2540
      DO 210 I=1,IMAX                                                   REZ2550
      DXSUM=DXSUM+DX(I)                                                 REZ2560
      IF (PRXRT.GE.DXSUM-.5*DX(I).AND.PRXRT.LT.DXSUM+.5*DX(I+1)) GO TO 2REZ2570
     120                                                                REZ2580
210   CONTINUE                                                          REZ2590
220   IPRT=I                                                            REZ2600
C                                                                       REZ2610
C            *** COMPUTE JTB, JTA - BOTTOM AND TOP CELL BOUNDARIES OF   REZ2620
C                 TARGET                                                REZ2630
C                                                                       REZ2640
230   IF (TAYBOT.LT.0..OR.(TAYTOP.LE.Y(NJMAX).AND.TAXRT.LE.X(NIMAX))) GOREZ2650
     1 TO 300                                                           REZ2660
      J=0                                                               REZ2670
      IF (TAYBOT.EQ.0.) GO TO 250                                       REZ2680
      DYSUM=0.                                                          REZ2690
      DO 240 J=1,JMAX                                                   REZ2700
      DYSUM=DYSUM+DY(J)                                                 REZ2710
      IF (TAYBOT.GE.DYSUM-.5*DY(J).AND.TAYBOT.LT.DYSUM+.5*DY(J+1)) GO TOREZ2720
     1 250                                                              REZ2730
240   CONTINUE                                                          REZ2740
      GO TO 300                                                         REZ2750
250   JTB=MIN0(J+1,JMAX)                                                REZ2760
      DO 260 J=JTB,JMAX                                                 REZ2770
      DYSUM=DYSUM+DY(J)                                                 REZ2780
      IF (TAYTOP.GE.DYSUM-.5*DY(J).AND.TAYTOP.LT.DYSUM+.5*DY(J+1)) GO TOREZ2790
     1 270                                                              REZ2800
260   CONTINUE                                                          REZ2810
270   JTA=J                                                             REZ2820
C                                                                       REZ2830
C            *** COMPUTE ITRT - RIGHT CELL BOUNDARY OF TARGET           REZ2840
C                                                                       REZ2850
      DXSUM=0.                                                          REZ2860
      DO 280 I=1,IMAX                                                   REZ2870
      DXSUM=DXSUM+DX(I)                                                 REZ2880
      IF (TAXRT.GE.DXSUM-.5*DX(I).AND.TAXRT.LT.DXSUM+.5*DX(I+1)) GO TO 2REZ2890
     190                                                                REZ2900
280   CONTINUE                                                          REZ2910
290   ITRT=I                                                            REZ2920
300   CONTINUE                                                          REZ2930
C            *** REDEFINE IMAX AND JMAX   FOR ORDERING THE K ARRAYS     REZ2932
C                 BELOW.                                                REZ2934
      IMAX=NIMAX                                                        REZ2940
      JFILB=JPA+1                                                       REZ2950
      JFILA=JTB-1                                                       REZ2960
      JMAX=NJMAX                                                        REZ2970
      I1=I1/2                                                           REZ2980
      I2=I2/2                                                           REZ2990
C            *** IS THIS A 1-D PROBLEM                                  REZ2995
      IF (IMAX.GT.1) GO TO 320                                          REZ3000
C            *** YES.ADD TARGET MATERIAL                                REZ3005
      JMP1=JMAX+1                                                       REZ3010
      JMAX=2*JMAX                                                       REZ3020
      DO 310 J=JMP1,JMAX                                                REZ3030
      K=J+1                                                             REZ3040
      AMX(K)=RHINIT*TAU(1)*DY(J)                                        REZ3050
```

```
      IF (TARGI.GT.0.) I2=J                                             REZ3060
      AIX(K)=TARGI                                                      REZ3070
      ETH=ETH+AMX(K)*AIX(K)                                             REZ3080
310   CONTINUE                                                          REZ3090
      JPROJ=JPROJ/2                                                     REZ3100
      I1=1                                                              REZ3110
      GO TO 520                                                         REZ3120
C            *** PREPARE TO SHUFFLE K ARRAYS SUCH AS TO PRESERVE        REZ3130
C                K=(J-1)*IMAX+I+1, THEN ADD MATERIAL TO NEW  PART       REZ3140
C                OF GRID.                                               REZ3145
320   DO 360 N=1,JMAX                                                   REZ3150
      J=JMAX+1-N                                                        REZ3160
      K=(J-1)*IMAX+1+IMAX                                               REZ3170
      L=(J-1)*(IMAX+IMAX)+1+IMAX                                        REZ3180
      DO 350 I=1,IMAX                                                   REZ3190
      AMX(L)=AMX(K)                                                     REZ3200
      AIX(L)=AIX(K)                                                     REZ3210
      U(L)=U(K)                                                         REZ3220
      V(L)=V(K)                                                         REZ3230
      IF (J-1) 340,340,330                                              REZ3240
330   AMX(K)=0.0                                                        REZ3250
      AIX(K)=0.0                                                        REZ3260
      V(K)=0.0                                                          REZ3270
      U(K)=0.0                                                          REZ3280
340   K=K-1                                                             REZ3290
      L=L-1                                                             REZ3300
350   CONTINUE                                                          REZ3310
360   CONTINUE                                                          REZ3320
C            *** REDEFINE IMAX,JMAX SO THEY WILL REPRESENT NUMBER       REZ3322
C                OF COLUMNS AND ROWS IN NEW GRID (SAME AS IN OLD GRID). REZ3324
      IMAX=NIMAX*2                                                      REZ3330
      JMAX=NJMAX*2                                                      REZ3340
      IL=NIMAX+1                                                        REZ3350
      JL=NJMAX+1                                                        REZ3360
      IF (PRYTOP.LE.Y(NJMAX).AND.PRXRT.LE.X(NIMAX).AND.TAYTOP.LE.Y(NJMAXREZ3370
     1).AND.TAXRT.LE.X(NIMAX)) GO TO 510                                REZ3380
C            *** ADD APPROPRIATE MATERIAL                               REZ3390
C                IN CELLS ABOVE (BUT NOT TO THE RIGHT OF) OLD GRID.     REZ3395
      DO 430 I=1,NIMAX                                                  REZ3400
      K=(JL-1)*IMAX+I+1                                                 REZ3410
      DO 420 J=JL,JMAX                                                  REZ3420
      IF (PRYBOT.LT.0.) GO TO 370                                       REZ3430
      IF (J.GE.JPB.AND.J.LE.JPA.AND.I.LE.IPRT) GO TO 390                REZ3440
C            *** NOT PROJECTILE-MATERIAL                                REZ3450
370   IF (RHOFIL.EQ.0.) GO TO 380                                       REZ3460
      IF (J.GE.JFILB.AND.J.LE.JFILA) GO TO 400                          REZ3470
C            *** NOT FILLER-MATERIAL                                    REZ3480
380   IF (TAYBOT.LT.0.) GO TO 420                                       REZ3490
      IF (J.GE.JTB.AND.J.LE.JTA.AND.I.LE.ITRT) GO TO 410                REZ3500
C            *** NOT TARGET. THUS,VACUUM.                               REZ3510
      GO TO 420                                                         REZ3520
C                *** ADD PROJ. MATERIAL                                 REZ3530
390   AMX(K)=RHINI*TAU(I)*DY(J)                                         REZ3540
      IF (PROJU.EQ.0..AND.VINI.EQ.0..AND.PROJI.EQ.0.) GO TO 420         REZ3550
      IF (I.GT.I1) I1=I                                                 REZ3560
      IF (J.GT.I2) I2=J                                                 REZ3570
      U(K)=PROJU                                                        REZ3580
```

```
      V(K)=VINI                                                         REZ3590
      AIX(K)=PROJI                                                      REZ3600
      GO TO 420                                                         REZ3610
C                 *** ADD FILLER                                        REZ3620
400   AMX(K)=RHOFIL*TAU(I)*DY(J)                                        REZ3630
      GO TO 420                                                         REZ3640
C                 *** ADD TARGET MATERIAL                               REZ3650
410   AMX(K)=RHINIT*TAU(I)*DY(J)                                        REZ3660
      IF (TARGV.EQ.0..AND.TARGI.EQ.0.) GO TO 420                        REZ3670
      IF (I.GT.I1) I1=I                                                 REZ3680
      IF (J.GT.I2) I2=J                                                 REZ3690
      V(K)=TARGV                                                        REZ3700
      AIX(K)=TARGI                                                      REZ3710
420   K=K+IMAX                                                          REZ3720
430   CONTINUE                                                          REZ3730
C             *** ADD APPROPRIATE MATERIAL TO CELLS ON THE RIGHT        REZ3732
C                 OF THE OLD GRID.                                      REZ3734
      DO 500 I=IL,IMAX                                                  REZ3740
      K=I+1                                                             REZ3750
      DO 490 J=1,JMAX                                                   REZ3760
      IF (PRYBOT.LT.0.) GO TO 440                                       REZ3770
      IF (J.GE.JPB.AND.J.LE.JPA.AND.I.LE.IPRT) GO TO 460                REZ3780
C             *** NOT PROJECTILE MATERIAL.                              REZ3785
440   IF (RHOFIL.EQ.0.) GO TO 450                                       REZ3790
      IF (J.GE.JFILB.AND.J.LE.JFILA) GO TO 470                          REZ3800
C             *** NOT FILLER MATERIAL.                                  REZ3805
450   IF (TAYBOT.LT.0.) GO TO 490                                       REZ3810
      IF (J.GE.JTB.AND.J.LE.JTA.AND.I.LE.ITRT) GO TO 480                REZ3820
C             *** NOT TARGET MATERIAL. THUS,VACUUM.                     REZ3825
      GO TO 490                                                         REZ3830
C             *** ADD PROJECTILE MATERIAL.                              REZ3835
460   AMX(K)=RHINI*TAU(I)*DY(J)                                         REZ3840
      IF (PROJU.EQ.0..AND.VINI.EQ.0..AND.PROJI.EQ.0.) GO TO 490         REZ3850
      IF (I.GT.I1) I1=I                                                 REZ3860
      IF (J.GT.I2) I2=J                                                 REZ3870
      U(K)=PROJU                                                        REZ3880
      V(K)=VINI                                                         REZ3890
      AIX(K)=PROJI                                                      REZ3900
      GO TO 490                                                         REZ3910
C             *** ADD FILLER.                                           REZ3915
470   AMX(K)=RHOFIL*TAU(I)*DY(J)                                        REZ3920
      GO TO 490                                                         REZ3930
C             *** ADD TARGET MATERIAL.                                  REZ3935
480   AMX(K)=RHINIT*TAU(I)*DY(J)                                        REZ3940
      IF (TARGV.EQ.0..AND.TARGI.EQ.0.) GO TO 490                        REZ3950
      IF (I.GT.I1) I1=I                                                 REZ3960
      IF (J.GT.I2) I2=J                                                 REZ3970
      V(K)=TARGV                                                        REZ3980
      AIX(K)=TARGI                                                      REZ3990
490   K=K+IMAX                                                          REZ4000
500   CONTINUE                                                          REZ4010
C             *** REDEFINE JPROJ,USUALLY J-INDEX OF TOP CELL IN         REZ4012
C                 PROJECTILE (INPUT PARAMETER).                         REZ4014
510   JPROJ=JPROJ/2                                                     REZ4020
C             *** REDEFINE ACTIVE GRID MARKERS.                         REZ4030
C                                                                       REZ4040
      I1=I1+2                                                           REZ4050
```

```
520   I2=I2+2                                                          REZ4060
      IF (I1.GT.IMAX) I1=IMAX                                          REZ4070
      IF (I2.GT.JMAX) I2=JMAX                                          REZ4080
C            *** CALL TO REZONE AND CDT COUNTED AS A CALCULATIONAL      REZ4082
C                CYCLE, SO NC AND T ARE INCREMENTED BEFORE PROCEEDING   REZ4084
C                ON TO PH1,PH3 AND PH2.                                 REZ4086
      WS=T+DTNA                                                        REZ4090
      NK=NC+1                                                          REZ4100
C                                                                      REZ4110
      WRITE (6,620) WS,NK,DX(1)                                        REZ4120
C            *** REDEFINE CONSTANTS AND CELL LIMITS FOR CALCULATING     REZ4122
C                TENSIONS AND STRESSES.                                 REZ4124
      KMAX=IMAX*JMAX+1                                                 REZ4130
      IMAXA=IMAX+1                                                     REZ4140
      JMAXA=JMAX+1                                                     REZ4150
      KMAXA=KMAX+1                                                     REZ4160
      N6=N6/2                                                          REZ4170
      JSTR=JSTR/2                                                      REZ4180
      IF (NUMREZ.GT.NREZ) NREZ=NUMREZ                                  REZ4190
      NPLACE=NREZ-NUMREZ+2                                             REZ4200
C            *** CALCULATE NEW ETI                                      REZ4210
      ETH=0.                                                           REZ4230
      DO 530 K=2,KMAX                                                  REZ4240
      ETH=ETH+AMX(K)*(AIX(K)+.5*(U(K)**2+V(K)**2))                     REZ4250
530   CONTINUE                                                         REZ4260
C                  ***DIVIDE JPM(1) BY 2 TO GET NEW PEAK PRESSURE CELLS.REZ4270
      DO 550 I=1,IMAX                                                  REZ4280
      L=2*I                                                            REZ4290
      IF (L.GT.IMAX) GO TO 540                                         REZ4300
      JPM(I)=JPM(L)/2                                                  REZ4310
      GO TO 550                                                        REZ4320
540   JPM(I)=0                                                         REZ4330
550   CONTINUE                                                         REZ4340
      IF (Y2.GT.(-1.)) GO TO 610                                       REZ4350
C                                                                      REZ4360
C            *** SCALE EXISTING TRACER POINTS                           REZ4370
C                                                                      REZ4380
      DO 560 J=1,JJ                                                    REZ4390
      DO 560 I=1,II                                                    REZ4400
      XP(I,J)=XP(I,J)/2.                                               REZ4410
      YP(I,J)=YP(I,J)/2.                                               REZ4420
560   CONTINUE                                                         REZ4430
C                                                                      REZ4440
C            *** REMOVE TRACER POINTS FROM EVERY OTHER CELL AND EVERY   REZ4450
C                OTHER ROW                                              REZ4460
C                                                                      REZ4470
      NMP=0                                                            REZ4480
      M=0                                                              REZ4490
      DO 570 J=1,JJ,2                                                  REZ4500
      M=M+1                                                            REZ4510
      L=0                                                              REZ4520
      DO 570 I=1,II,2                                                  REZ4530
      L=L+1                                                            REZ4540
      XP(L,M)=XP(I,J)                                                  REZ4550
      YP(L,M)=YP(I,J)                                                  REZ4560
570   NMP=NMP+1                                                        REZ4570
      JTPB=1                                                           REZ4580
```

```
      JTPT=INT(FLOAT(JJ)/2.+.6)                                         REZ4590
      ITPL=INT(FLOAT(II)/2.+.6)+1                                       REZ4600
C                                                                       REZ4610
C           *** INITIALIZE REMAINING TRACER POINT STORAGE AND           REZ4620
C           *** PLACE NEW TRACER POINTS FIRST IN NEW CELLS ABOVE OLD    REZ4630
C               GRID THEN IN NEW CELLS TO THE RIGHT OF OLD GRID.        REZ4635
C                                                                       REZ4640
580   DO 600 J=JTPB,JTPT                                                REZ4650
      DO 600 I=ITPL,II                                                  REZ4660
      XP(I,J)=0.                                                        REZ4670
      YP(I,J)=0.                                                        REZ4680
      K=2*((J-1)*IMAX+I)                                                REZ4690
      IF (AMX(K).EQ.0.) GO TO 590                                       REZ4700
      ICELL=2*I-1                                                       REZ4710
      JCELL=2*J-1                                                       REZ4720
C           *** PLACE NEW TRACER POINTS IN CELLS SO THEY LINE UP        REZ4722
C               WITH EXISTING TRACER POINTS BY USING NPLACE WHICH       REZ4724
C               IS A FUNCTION OF THE NUMBER OF REZONES PERFORMED.       REZ4726
      XP(I,J)=FLOAT(ICELL-1)+1./2.**NPLACE                              REZ4730
      YP(I,J)=FLOAT(JCELL-1)+1./2.**NPLACE                              REZ4740
590   NMP=NMP+1                                                         REZ4750
600   CONTINUE                                                          REZ4760
      IF (J.GE.JJ) GO TO 610                                            REZ4770
C           *** GO BACK THROUGH LOOPS ADDING POINTS ON RIGHT SIDE OF    REZ4772
C               OLD GRID.                                               REZ4774
      JTPB=JTPT+1                                                       REZ4780
      JTPT=JJ                                                           REZ4790
      ITPL=1                                                            REZ4800
      GO TO 580                                                         REZ4810
610   RETURN                                                            REZ4820
C                                                                       REZ4830
620   FORMAT (1H ////22H PROBLEM REZONED AT T=,1PE12.6,6X,5HCYCLEI4,6X,6REZ4840
     1HDX(1)=,E12.6////)                                                REZ4850
      END                                                               REZ4860
```

```
      SUBROUTINE ERROR                                                  ERR  10
C     ..................................................................ERR  20
C                                                                       ERR  30
      DIMENSION AMX(2502),AIX(2502),U(2502)  ,V(2502)  ,P(2502)  ,      ERR  40
     1          X(52)    ,XX(54)   ,TAU(52)  ,JPM(52)  ,                ERR  50
     2          Y(102)   ,YY(104)  ,FLEFT(102), YAMC(102), SIGC(102),   ERR  60
     3          GAMC(102),                                             ERR  70
     4          PK(15),   Z(150)   ,                                   ERR  80
     5          XP(26,51),YP(26,51),                                   ERR  90
     6          PL(204)  ,UL(204)  ,PR(204)  ,                         ERR 100
     7           RSN(52),   RST(52),                                   ERR 110
     8          CMXP(5)  ,CMYP(5)  ,IJ(5)    ,JK(5)    ,               ERR 120
     9          DX(52)   ,DDX(54)  ,DY(102)  ,DDY(104) ,               ERR 130
     $          SNB(52)  ,STB(52)  ,UK(52,3) ,VK(52,3) ,RHO(52,3)      ERR 140
C            *** DIMENSIONED ARRAYS                                     ERR 150
C            *** Z-BLOCK IS SAVED ON TAPE.                              ERR 160
      COMMON     Z                                                     ERR 170
      COMMON    PK                                                     ERR 180
      COMMON  YY,     XX                                               ERR 190
      COMMON  DDX,    DDY                                              ERR 200
      COMMON  AMX,    AIX,    U,    V,    P                            ERR 210
      COMMON  TAU,    JPM                                              ERR 220
      COMMON  UL ,    PL                                               ERR 230
      COMMON  XP ,    YP,     CMXP, CMYP                               ERR 240
C            *** NON-DIMENSIONED VARIABLES                              ERR 250
      COMMON        AID   ,AMMV  ,AMMY  ,AMPY  ,AMUR  ,AMUT  ,AMVR  ,  ERR 260
     1AMVT  ,DELEB ,DELER ,DELET ,DELM  ,DTODX ,DXYMIN,EAMMP ,EAMPY ,   ERR 270
     2E     ,ERDUMP,I     ,I3    ,IWS   ,J     ,K     ,KA    ,KB    ,   ERR 280
     3LL    ,MD    ,ME    ,MZT   ,NERR  ,NK    ,NPRINT,                ERR 290
     4NR    ,NRZ   ,NULLE ,PIDTS ,SIEMIN,SNR   ,SNT   ,STR   ,SOLID ,  ERR 300
     5SUM   ,TESTRN,TWOPI ,URR   ,WS    ,WSA   ,WSB   ,WSC   ,WFLAGF,   ERR 310
     6WFLAGL,WFLAGP                                                    ERR 320
C                                                                       ERR 330
C            *** THE FOLLOWING EQUIVALENCES MAKE AVAILABLE              ERR 340
C                X(0), Y(0), DX(0), DY(0)                               ERR 350
C                                                                       ERR 360
      EQUIVALENCE (XX(2), X(1)),  (YY(2), Y(1))                         ERR 370
      EQUIVALENCE (DDX(2), DX(1)),  (DDY(2), DY(1))                     ERR 380
C                                                                       ERR 390
C            *** SPECIAL EQUIVALENCES FOR PH2 ONLY                      ERR 400
C                                                                       ERR 410
      EQUIVALENCE          (UL,FLEFT),        (UL(103),YAMC),          ERR 420
                           (PL,GAMC,PR),      (PL(103),SIGC)           ERR 430
C                                                                       ERR 440
C            *** SPECIAL EQUIVALENCES FOR PH3 ONLY                      ERR 450
C                                                                       ERR 460
      EQUIVALENCE          (UL,RSN),                                   ERR 470
     1                     (PL,RST),          (P,UK),                  ERR 480
     2                     (P(157),VK),       (P(313),SNB),            ERR 490
     3                     (P(365),STB),      (P(417),RHO)             ERR 500
C                                                                       ERR 510
C            *** SPECIAL EQUIVALENCES FOR EDIT                          ERR 520
C                                                                       ERR 530
      EQUIVALENCE (PR(1), IJ),  (PR(6), JK)                             ERR 540
C                                                                       ERR 550
C            *** Z-STORAGE EQUIVALENCES                                 ERR 560
C                                                                       ERR 570
      EQUIVALENCE                      (Z(  1),PROB  ),(Z(  2),CYCLE ), ERR 580
```

```
     1(Z(  3),DT     ),(Z(  4),NUMSP ),(Z(  5),NFRELP),(Z(  6),NDUMP7),  ERR 590
     2(Z(  7),ICSTOP),(Z(  8),PIDY  ), (Z(  9),TOPMU ),(Z( 10),RTMU  ), ERR 600
     3(Z( 11),STK1  ),(Z( 12),NUMREZ), (Z( 13),ETH   ),(Z( 14),UN14  ), ERR 610
     4(Z( 15),RHINIT),(Z( 16),PROJI ), (Z( 17),UN17  ),(Z( 18),XMAX  ), ERR 620
     5(Z( 19),NZ    ),(Z( 20),NREZ  ), (Z( 21),AMDM  ),(Z( 22),UVMAX ), ERR 630
     6(Z( 23),UN23  ),(Z( 24),DMIN  ), (Z( 25),JSTR  ),(Z( 26),DTNA  ), ERR 640
     7(Z( 27),CVIS  ),(Z( 28),STK2  ), (Z( 29),STEZ  ),(Z( 30),NC    ), ERR 650
     8(Z( 31),UN31  ),(Z( 32),NRC   ), (Z( 33),IMAX  ),(Z( 34),IMAXA ), ERR 660
     9(Z( 35),JMAX  ),(Z( 36),JMAXA ), (Z( 37),KMAX  ),(Z( 38),KMAXA )  ERR 670
      EQUIVALENCE                                                       ERR 680
     1(Z( 39),BOTM  ),(Z( 40),BOTMV ), (Z( 41),NUMSPT),(Z( 42),CZERO ), ERR 690
     2(Z( 43),NUMSCA),(Z( 44),PRLIM ), (Z( 45),PRDELT),(Z( 46),PRFACT)  ERR 700
      EQUIVALENCE                                                       ERR 710
     1(Z( 47),I1    ),(Z( 48),I2    ), (Z( 49),IPCYCL),(Z( 50),TSTOP ), ERR 720
     2(Z( 51),RHOFIL),(Z( 52),TARGV ), (Z( 53),N3    ),(Z( 54),IVARDY), ERR 730
     3(Z( 55),VT    ),(Z( 56),N6    ), (Z( 57),RTM   ),(Z( 58),RTMV  ), ERR 740
     4(Z( 59),UN59  ),(Z( 60),N10   ), (Z( 61),N11   ),(Z( 62),GAMMA ), ERR 750
     5(Z( 63),TOPM  ),(Z( 64),BOTMU ), (Z( 65),SN    ),(Z( 66),TOPMV ), ERR 760
     6(Z( 67),PRYBOT),(Z( 68),PRYTOP), (Z( 69),PRXRT ),(Z( 70),CYCPH3), ERR 770
     7(Z( 71),REZFCT),(Z( 72),TARGI ), (Z( 73),PROJU ),(Z( 74),BBOUND), ERR 780
     8(Z( 75),EVAP  ),(Z( 76),ECK   ), (Z( 77),NECYCL),(Z( 78),II    ), ERR 790
     9(Z( 79),JJ    ),(Z( 80),NMP   ), (Z( 81),Y2    ),(Z( 82),EZPH1 )  ERR 800
      EQUIVALENCE                                                       ERR 810
     1(Z( 83),IVARDX),(Z( 84),T     ), (Z( 85),NMPMAX),(Z( 86),PMIN  ), ERR 820
     2(Z( 87),INTER ),(Z( 88),TAYBOT), (Z( 89),TAYTOP),(Z( 90),IEMAP ), ERR 830
     3(Z( 91),MC    ),(Z( 92),MR    ), (Z( 93),MZ    ),(Z( 94),MB    )  ERR 840
      EQUIVALENCE                                                       ERR 850
     1(Z( 95),REZ   ),(Z( 96),NODUMP), (Z( 97),UN97  ),(Z( 98),UN98  ), ERR 860
     2(Z( 99),UN99  ),(Z(100),EVAPM ), (Z(101),EVAPEN),(Z(102),EVAPMU), ERR 870
     3(Z(103),EVAPMV),(Z(104),EZPH2 ), (Z(105),SNL   ),(Z(106),STL   ), ERR 880
     4(Z(107),TAXRT ),(Z(108),IDNMAP), (Z(109),IPRMAP),(Z(110),ROEPS ), ERR 890
     5(Z(111),RHINI ),(Z(112),VINI  ), (Z(113),FINAL ),(Z(114),IVMAP ), ERR 900
     6(Z(115),RHOZ  ),(Z(116),ESA   ), (Z(117),ESEZ  ),(Z(118),ESB   ), ERR 910
     7(Z(119),ESCAPA),(Z(120),ESESP ), (Z(121),ESESQ ),(Z(122),ESES  ), ERR 920
     8(Z(123),ESALPH),(Z(124),ESBETA), (Z(125),ESCAPB),(Z(126),IUMAP ), ERR 930
     9(Z(127),SS1   ),(Z(128),SS2   ), (Z(129),UMIN  ),(Z(130),SS4   )  ERR 940
      EQUIVALENCE                                                       ERR 950
     1(Z(131),PRTIME),(Z(132),EOR   ), (Z(133),EOT   ),(Z(134),EOB   ), ERR 960
     2(Z(135),EMOR  ),(Z(136),DXF   ), (Z(137),DYF   ),(Z(138),RHOMIN), ERR 970
     3(Z(139),STAB), (Z(140),XIENRG),  (Z(141),XKENRG), (Z(142),XTENRG),ERR 980
     4(Z(143),STT   ),(Z(144),DTMIN ), (Z(145),TRNSFC),(Z(146),EMOT  ), ERR 990
     5(Z(147),JPROJ ),(Z(148),CNAUT ), (Z(149),BBAR  ),(Z(150),EMOB  )  ERR1000
C                                                                       ERR1010
C     ..................................................................ERR1020
C                                                                       ERR1030
C     END OF COMMON                                                     ERR1040
C                                                                       ERR1050
C     ..................................................................ERR1060
C                                                                       ERR1070
      IF (NERR.EQ.1) GO TO 120                                          ERR1080
      GO TO (10,20,30,40,50,60,70,80,90,100), NR                        ERR1090
10    WRITE (6,130) NK                                                  ERR1100
      GO TO 110                                                         ERR1110
20    WRITE (6,140) NK                                                  ERR1120
      GO TO 110                                                         ERR1130
30    WRITE (6,150) NK                                                  ERR1140
      GO TO 110                                                         ERR1150
```

```
40    WRITE (6,160) NK                                                  ERR1150
      GO TO 110                                                         ERR1170
50    WRITE (6,170) NK                                                  ERR1180
      GO TO 110                                                         ERR1190
60    WRITE (6,180) NK                                                  ERR1200
      GO TO 110                                                         ERR1210
70    WRITE (6,190) NK                                                  ERR1220
      GO TO 110                                                         ERR1230
80    WRITE (6,200) NK                                                  ERR1240
      GO TO 110                                                         ERR1250
90    WRITE (6,210) NK                                                  ERR1260
      GO TO 110                                                         ERR1270
100   WRITE (6,220) NK                                                  ERR1280
110   WRITE (6,230) I,J,K,(M,Z(M),Z(M),M=1,150)                         ERR1290
C           *** IF NR=1, ERROR IS IN INPUT DECK                         ERR1292
      IF(NR.EQ.1) GO TO 120                                             ERR1294
C           *** IF NR=5 AND NK=130, EDIT PRINT HAS JUST BEEN DONE. BY   ERR1296
C               SETTING ERDUMP=1.,EDIT WILL DO A TAPE DUMP BUT NOT      ERR1298
C               ANOTHER PRINT.                                          ERR1300
      IF (NR.EQ.5.AND.NK.EQ.130) ERDUMP=1.                              ERR1310
      NERR = 1                                                          ERR1315
      I3=I1                                                             ERR1320
      NPRINT=1                                                          ERR1330
      WFLAGL=1.                                                         ERR1340
      NUMSPT=NDUMP7                                                     ERR1350
      CALL EDIT                                                         ERR1360
120   CALL EXIT                                                         ERR1370
C                                                                       ERR1380
C                                                                       ERR1390
130   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1400
     1 INPUT )                                                          ERR1410
140   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1420
     1 SETUP )                                                          ERR1430
150   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1440
     1 CDT   )                                                          ERR1450
160   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1460
     1 ES    )                                                          ERR1470
170   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1480
     1 EDIT  )                                                          ERR1490
180   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1500
     1 MAP   )                                                          ERR1510
190   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1520
     1 PH1   )                                                          ERR1530
200   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1540
     1 PH3   )                                                          ERR1550
210   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1560
     1 PH2   )                                                          ERR1570
220   FORMAT (1H1,5X,36H*** ERROR EXIT - SEE STATEMENT NUMBER ,I5,10H INERR1580
     1 REZONE)                                                          ERR1590
230   FORMAT (//5X,6H    I=,I3,6H    J=I3,6H    K=I3//16X,7HZ-BLOCK//6X,ERR1600
     115H  REAL FORMAT  ,5X,15H INTEGER FORMAT/2X,1HI,8X,4HZ(I),17X,4HZ(ERR1610
     2I)//(I4,2X,E15.6,5X,I15))                                         ERR1620
      END                                                               ERR1630-
```

7. DICTIONARY

This section includes a description of the use and location of each of the variables in the program. The following terminology is used in the dictionary:

"Local"	Means name is local to subroutine (not in Blank Common).
"Local$^{(C)}$"	Means name is in Blank Common (or equivalenced to a variable in Blank Common), but its value is never passed to another subroutine.
"Global"	Means name is in Blank Common (or equivalenced to a variable in Blank Common) and its value is passed from one routine to another.
= Z(N)	Means variable is equivalenced to a member of the Z-array, the first array in Blank Common. These variables are usually used in setting up and restarting.

ADDVL	Local	Used in SETUP. Used in finding volume of cells containing sphere-boundary.
AID	Local$^{(C)}$	Used in EDIT in calculation of crater depth.
AIX	Global	Specific interral energy in a cell. (IMAX by JMAX array.)
ALE	Constants	Used in MAP. This array has alphabetic characters for pressure, density, velocity, and energy maps. (Defined in DATA statement.)
AMDM	= Z(21)	INPUT parameter. A cell with compression > AMDM is considered solid. Usual value: 0.95 to 0.99. Used in ES in testing whether to allow negative pressures (tensions). Used in INPUT to calculate SOLID = AMDM * RHOZ, which is used in CDT and PH3.
AMMP	Local	Used in PH2. Mass moving across right boundary of a cell. (See Appendix B)

AMMU	Local	Used in PH2. Radial momentum transported across the bottom boundary of a cell. (See Appendix B)
AMMV	Local(C)	Used in PH2. Axial momentum transported across the bottom boundary of a cell. (See Appendix B)
AMMY	Local(C)	Used in PH2. Amount of mass moving across bottom of a cell. (See Appendix B)
AMPY	Local(C)	Used in PH2. Amount of mass moving across top of a cell. (See Appendix B)
AMUR	Local(C)	Used in PH2. Radial momentum transported across right boundary of a cell. (See Appendix B)
AMUT	Local(C)	Used in PH2. Radial momentum transported across top boundary of cell. (See Appendix B)
AMVR	Local(C)	Used in PH2. Axial momentum transported across right boundary of a cell. (See Appendix B)
AMVT	Local(C)	Used in PH2. Axial momentum transported across top boundary of a cell. (See Appendix B)
AMX	Global	Mass in a cell. (IMAX by JMAX array.)
AREAFC	Local(C)	Used in SETUP. Area of a cell-face. Used in setting up a sphere. Equivalenced to DELEB.
B	Local	Used and calculated in PH3
BBAR	= Z(149)	Used in CDT. An INPUT parameter used in local sound-speed calculation whose value depends on the kind of material. (Local sound-speed is approximated as $C_0 + (BBAR) \cdot \sqrt{P(K)}$.)
BBOUND	= Z(74)	Calculated in PH3. Printed in EDIT under "Plastic-Work." Total work done by the plastic stresses.
BOTM	= Z(39)	Calculated in PH2. Printed in EDIT. Total mass lost out bottom of grid.
BOTMU	= Z(64)	Calculated in PH2. Printed in EDIT. Total radial-momentum lost out bottom of grid.

BOTMV	= Z(40)	Calculated in PH2. Printed in EDIT. Total axial-momentum lost out bottom of grid.
CMXP	Local(C)	Used and calculated in EDIT for printing the centimeter coordinates of the tracer points.
CMYP	Local(C)	
CNAUT	= Z(148)	Used in CDT, INPUT. Approximate sound-speed of material; calculated in INPUT as $C_0 = \sqrt{\frac{ESCAPA}{RHOZ}} = \sqrt{A/\rho}$.
CRAD	Local(C)	Used in EDIT for printing radii of crater depths. Equivalenced to UL array.
CVIS	= Z(27)	INPUT parameter. Used to describe the bottom boundary-condition. Used in PH1, PH2, PH3. Bottom boundary is transmittive when CVIS = -1., reflective when CVIS = 0.
CYCLE	= Z(2)	Used in INPUT, SETUP, CDT, EDIT. Cycle number (an integer value in floating point form).
CYCPH3	= Z(70)	Used in MAIN and PH3. INPUT parameter: Number of times to subcycle PH3. If value is -1., PH3 is omitted.
CZERO	= Z(42)	INPUT parameter. Value of Y_0 for yield strength calculation. Used in PH3. (See STRENG)
DDX	Global	An array equivalenced to the DX array such that DDX(1) = DX(0).
DDY	Global	An array equivalenced to the DY array such that DDY(1) = DY(0).
DELEB	Local(C)	Used in PH2. Total energy associated with mass transported across bottom boundary of a cell. (See Appendix B)
DELER	(Local(C)	Used in PH2. Total energy associated with mass transported across right boundary of a cell. (See Appendix B)
DELET	Local(C)	Used in PH2. Total energy associated with mass transported across top boundary of a cell. (See Appendix B)

DELI	Local	Used in PH2, PH3. Change of specific internal energy of a cell.
DEIM	Local(c)	Used in PH2 for total mass moving into or out of a cell.
DELU	Local	Used in PH2, PH3. Change of radial velocity of a cell.
DELV	Local	Used in PH2, PH3. Change of axial velocity of a cell.
DMIN	= Z(24)	INPUT parameter. Allowable relative error in energy sum. If error is > DMIN then calculation is terminated. Used in EDIT. If everything is working right you should be able to use 10^{-3} for DMIN.
DSCALE	Local	Used in MAP as linear scale factor for compression map.
DT	= Z(3)	Time step. Calculated in CDT. Used in SETUP, EDIT, PH1, PH2 and PH3.
DTFACT	Local	Used in PH3 in calculating a variable time step when subcycling the PH3 calculations.
DTMIN	= Z(144)	INPUT parameter. Used in CDT. After STAB = FINAL, if DT < DTMIN execution is stopped.
DTNA	= Z(26)	DT from previous time cycle. Used in INPUT, CDT, EDIT, REZONE and PH1.
DTNOW	Local	Used in EDIT. Used for saving DT when calling CDT to recalculate pressures after a REZONE.
DTODX	Local(c)	Used in PH2 for DT/DX.
DTODY	Local	Used in PH2 for DT/DY.
DTSTR	Local	Used in PH3. DT for recycling through PH3.
DUODX	Local	Used in PH3. DU/DX.
DUODY	Local	Used in PH3. DU/DY.
DVODX	Local	Used in PH3. DV/DX

DVODY	Local	Used in PH3. DV/DY.
DX	Global	The radial dimension of cells. Equivalenced to DDX such that DDX(1) = DX(0).
DXF	= Z(136)	An INPUT parameter used to calculate the DX array if the radial dimension of the cells is uniform.
DXSUM	Local	Used in SETUP and REZONE to find cell dimensions of packages when DX is not constant.
DXYMIN	Local(C)	Used in CDT. Minimum (DX, DY) of a cell. Used in calculation of SRATIO and DT.
DY	Global	The axial-dimension of cells. Equivalenced to DDY so that DDY(1) = DY(0).
DYF	= Z(137)	INPUT parameter. DY of all cells, if DY is constant.
DYSUM	Local	Used in SETUP and REZONE to find cell dimensions of packages when DY is not constant.
E	Local(C)	Used in REZONE, PH1, and PH3. Temporary storage for energy calculations.
EAMMP	Local(C)	Used in PH2. Specific internal energy of mass moving across right edge of cell.
EAMPY	Local(C)	Used in PH2. Specific internal energy of mass moving across top of cell.
ECK	= Z(76)	Used in EDIT. Relative error in energy sum. If |ECK| > DMIN, execution is stopped.
EMOB	= Z(150)	Calculated in PH2. Printed in EDIT. Energy change out of bottom of mesh.
EMOR	= Z(135)	Calculated in PH2. Printed in EDIT. Energy change out right side of mesh.
EMOT	= Z(146)	Calculated in PH2. Printed in EDIT. Energy change out of top of mesh.
ENERGY	Local	Used in PH2 to sum energy of cells.
ENGY	Local	Used in PH2 as temporary storage for energy of a cell.

EOB	= Z(134)	Calculated in PH1. Printed in EDIT. Energy change due to work done at bottom boundary.
EOR	= Z(132)	Calculated in PH1. Printed in EDIT. Energy change due to work done at right boundary.
EOT	= Z(133)	Calculated in PH1. Printed in EDIT. Energy change due to work done at top boundary.
ERDUMP	Local(c)	Used in EDIT and ERROR. Flags EDIT to stop execution because ERROR has been called.
ESA	= Z(116)	INPUT parameter. Value of "a" in equation of state. Used in ES. (= γ - 1 when using perfect gas equation of state.)
ESALPH	= Z(123)	INPUT parameter. Value of "α" in equation of state. Used in ES.
ESB	= Z(118)	INPUT parameter. Value of "b" in equation of state. Used in ES.
ESBETA	= Z(124)	INPUT parameter. Value of "β" in equation of state. Used in ES.
ESCAPA	= Z(119)	INPUT parameter. Value of "A" in equation of state. Used in ES.
ESCAPB	= Z(125)	INPUT parameter. Value of "B" in equation of state. Used in ES.
ESCLE	Local	Used in MAP as a logrithmic scale factor for energy map.
ESES	= Z(122)	INPUT parameter. Value of ES in equation of state. Used in ES.
ESESP	= Z(120)	INPUT parameter. Value of ES′ in equation of state. Used in ES.
ESESQ	= Z(121)	INPUT parameter. ESESQ is usually equal to ESESP. It is used to test whether a cell should be considered hot or cold in free-surface treatment. Used in CDT and PH2.

ESEZ	= Z(117)	INPUT parameter. Value of "E_o" in equation of state. Used in ES.
ESUM	Local	Used in EDIT to sum energy and calculate relative error of energy sum.
ETA	Local	Used and calculated in ES. = ρ/ρ_o.
ETH	= Z(13)	Theoretical value of total energy in the mesh. Used in SETUP, EDIT, REZONE, PH1, PH2, PH3. Calculated in SETUP initially; in PH2 thereafter. It is redefined in REZONE.
EVAP	= Z(75)	INPUT parameter. Used in PH2. Any cell with density less than EVAP times initial density of "projectile" is "evaporated" and its energy subtracted from theoretical energy of system. (10^{-3} to 10^{-8} are appropriate values.)
EVAPEN	= Z(101)	Calculated in PH2 and CDT. Printed in EDIT. Sum of energy lost through "evaporation" described under EVAP. Adjusted in CDT when "evaporating" energy of isolated cells. Initialized in SETUP. Equivalenced to SIESPH in SETUP.
EVAPM	= Z(100)	Calculated in PH2. Printed in EDIT. Sum of mass lost through "evaporation" described under EVAP. Used in PH2 and CDT when "evaporating" mass of isolated cells. Initialized in SETUP. Equivalenced to RHOSPH in SETUP.
EVAPMU	= Z(102)	Calculated in PH2. Printed in EDIT. Sum of radial momenta lost through "evaporation." Used in PH2 and CDT when "evaporating" momentum of isolated cells. Initialized in SETUP. Equivalenced to VINSPH in SETUP.
EVAPMV	= Z(103)	Calculated in PH2. Sum of axial momenta lost through "evaporation". Printed in EDIT. Used in PH2 and CDT when "evaporating" momentum of isolated cells. Initialized in SETUP. Equivalenced to RHOOUT in SETUP.

EZPH1	= Z(82)	Energy gained through setting negative internal energies to zero in PH1. Printed in EDIT.
EZPH2	= Z(104)	Calculated in PH2. Sum of specific internal energy fluxes less than SIEMIN and negative internal energies set to zero. Printed in EDIT.
FINAL	= Z(113)	INPUT parameter. Maximum value of stability fraction (STAB). If FINAL = 0; the stability fraction will be constant. Used in CDT.
FLEFT	Local(c)	Used in PH2. Radial momentum of mass moving across left side of cell. Equivalenced to UL array. (See Appendix B)
FLOTMA	Local	Used in MAP.
FRX	Local	Used in PH2 for moving tracer-points.
FRY	Local	Used in PH2 for moving tracer-points.
GAMC	Local(c)	Used in PH2. Mass moving across left side of cell. Equivalenced to PL and PR arrays. (See Appendix B)
GAMMA	= Z(62)	Calculated value of GAMMA = ESA +1. Calculated in INPUT. Used in CDT.
HOOP	Local	Used in PH3. Hoop stress.
I	Local (c)	Used in most subroutines as index in radial direction.
IAID	Local	Used in EDIT in crater depth calculation.
ICELL	Local	Used in REZONE when placing tracer points in new cells.
ICP3	Local	Used in PH3: = INT(CYCPH3).
ICSTOP	= Z(7)	INPUT parameter. Used in EDIT. Execution stops on ICSTOP cycle when stopping on cycles rather than time.
IDL	Local	Used in MAP. Number of columns in maps. On cycle 0, IDL = IMAX; otherwise IDL = I1.

IDNMAP	= Z(108)	Defined in INPUT. Used in MAP to specify the number of symbols to be used in the density, energy, pressure, u-velocity, and v-velocity maps, respectively.
IEMAP	= Z(90)	
IPRMAP	= Z(109)	
IUMAP	= Z(116)	
IVMAP	= Z(114)	
II	= Z(78)	Used in REZONE, EDIT, and SETUP. The number of tracer points in each row.
IJ	Local(c)	Used in EDIT. Used to identify which column a tracer point originated in. Equivalenced to PR(1) in EDIT.
IK	Local	Used in PH3 as index, = I + 1.
IKK	Local	Used in PH3, = IK + 1.
ILIM1	Local	Used in MAP as index for printing values of symbols.
ILIM2	Local	See ILIM1.
IMAX	= Z(33)	INPUT parameter. Number of columns in mesh. IMAX must be an even number if grid is to be rezoned with the exception that IMAX = 1 for a 1-D problem. Used in SETUP, CDT, REZONE, EDIT, PH1, PH2, and PH3.
IMAXA	= Z(34)	IMAX + 1. Used in SETUP and REZONE.
INTER	= Z(87)	INPUT parameter. If INTER ≠ 0, EDIT will print after CDT, PH1 and PH3. If INTER = 99, in addition to extra EDIT prints, stresses are printed in PH3. (LOTS of printing.) If INTER = 7, energy totals are printed in PH2 in addition to the extra EDIT prints. Used in MAIN, EDIT, PH1 and Ph2 and PH3.
INTMA	Local	Used in MAP.
IP	Local	Used in EDIT. The column a tracer point is in.
IPCYCL	= Z(49)	INPUT parameter. Used in EDIT. The number of cycles between EDIT prints when printing on cycles rather than time.
IPRT	Local	Used in REZONE. Number of columns in projectile after rezoning.

ISPHMX	Local	Used in SETUP. I-index of right-most column which contains sphere material.
ITPL	Local	Used in REZONE for adding tracer-points in added cells.
ITRT	Local(c)	Used in REZONE. Number of columns in target after rezoning.
IVARDX	= Z(83)	Used in SETUP. Flag for variable radial dimension of cells.
IVARDY	= Z(54)	Used in SETUP. Flag for variable axial dimension of cells.
IWS	Local(c)	Used as local index in INPUT, SETUP, CDT, EDIT, PH2 and PH3.
IX	Local	Used as index in PH2 for tracer-point movement.
IY	Local	Used as index in PH2 for tracer-point movement.
I1	= Z(47)	INPUT parameter. I1 is used to limit calculation in radial direction to "active mesh." Beyond I1 nothing is yet disturbed from initial conditions. I1 is specified initially as (2 + the column-number of the last column in which there is a non-zero velocity or internal energy). However, I1 is never larger than IMAX. I1 is increased automatically as inactive cells beome active. If IMAX = 1, then I1 = 1. Used in CDT, EDIT, REZONE, PH1, PH2 and PH3.
I2	= Z(48)	INPUT parameter. Like I1 but for axial-disturbance-limit. I2 is specified initially as (2 + the number of the upper-most row in which there is a non-zero velocity or internal energy). I2 is increased automatically as inactive cells become active. However, I2 is never larger than JMAX. Used in SETUP, CDT, EDIT, REZONE, PH1, PH2, PH3.
I3	Local(c)	Used in EDIT as a flag for "short" or "long" prints.
J	Local(c)	Used as row-index in most subroutines.

JA	Local	Used in SETUP to calculate J-index of top of sphere.
JB	Local	Used in SETUP to calculate J-index of bottom of sphere.
JCELL	Local	Used in REZONE when placing tracer points in added cells.
JCENTR	Local	Used in SETUP. J-index of row just below center of sphere.
JDL	Local	Used in MAP. Number of rows in maps. On cycle 0, JDL = JMAX; otherwise, JDL = I2.
JFILA	Local	Used in REZONE. The J-index of the row immediately below the target.
JFILB	Local	Used in REZONE. The J-index of the row immediately above the projectile.
JFLAG	Local	Used in PH3. Used in connection with JPM for deciding where to stop calculating stresses.
JINTL	Local	Used in CDT in defining JPM array.
JJ	= Z(79)	Used in REZONE, EDIT and SETUP. Number of tracer points in each column.
JK	Local(c)	Used in EDIT. Used to identify which row a tracer point originated in. Equivalenced to PR(6).
JMAX	= Z(35)	INPUT parameter. Number of rows in mesh. JMAX must be an even number if grid is to be rezoned. Used in SETUP, CDT, EDIT, REZONE, PH1, PH2, PH3.
JMAXA	= Z(36)	JMAX + 1. Used in SETUP and REZONE.
JMP1	Local	Used in REZONE. Limit on do-loop, = JMAX/2 + 1.
JP	Local	Used as an index in CDT. Used in EDIT. The row a tracer point is in.
JPA	Local	Used in REZONE. The J-index of the top row of projectile.
JPB	Local	Used in REZONE. The J-index of the bottom row of projectile.

JPM	Global	Calculated in CDT; used in PH3. Initialized, adjusted and saved on tape in SETUP, INPUT, EDIT and REZONE. JPM(I) is J-index of cell with local maximum pressure in column I.
JPROJ	= Z(147)	INPUT parameter. Usually the J-index of top cell in projectile. Used in SETUP and EDIT. Adjusted in REZONE. The zero point in the crater depth calculation. A division for printout of total energies, mass and momenta.
JRADA	Local	Used in SETUP. The J-index of the top cell on the axis containing a part of the sphere.
JRADB	Local	Used in SETUP. The J-index of the bottom cell on the axis containing a part of the sphere.
JSPHBT	Local	Used in SETUP as index in placing sphere.
JSPHTP	Local	Used in SETUP as index in placing sphere.
JSTR	= Z(25)	INPUT parameter. When active-grid gets to JSTR in J direction, stress calculations begin and negative pressures are permitted. JSTR needs to be large enough so that a shock can become well established before stress calculations begin and negative pressures are allowed. Otherwise, meaningless perturbations are calculated in material which is still at rest. Used in PH3, CDT and REZONE.
JTA	Local	Used in REZONE as index.
JTB	Local	Used in REZONE as index.
JTPB	Local	Used in REZONE. Index for "weeding out" and adding tracer points.
JTPT	Local	Used in REZONE. See JTPB.
K	Local(C)	Used as cell-index in all subroutines.
KA	Local(C)	Used as index in CDT and PH2.
KB	Local(C)	Used as index in CDT.

KK	Local	Used as index in EDIT to remove tracer points from empty cells.
KMAX	= Z(37)	Calculated in SETUP (IMAX*JMAX+1). Used in PH3, SETUP, EDIT, REZONE. Largest value of K (cell-index).
KMAXA	= Z(38)	Calculated in SETUP (KMAX+1). Used in INPUT, SETUP, EDIT and REZONE.
KSPACE	Local	Used in EDIT for spacing printed output.
L	Local	Used as index in EDIT, INPUT, PH2, PH3, REZONE.
LA	Local	Used as index in PH2.
LB	Local	Used as index in PH2.
LJD	Local	Used as index in PH2 and PH3.
LL	Local(C)	Used as index in PH1, PH2 and PH3.
LOCA	Local	Used in EDIT and PH3 in assigned GO TO statements.
LOCB	Local	Used in EDIT in assigned GO TO statements.
M	Local	Used as index in SETUP, EDIT, PH2, REZONE, PH3, ERROR.
MA	Local	Used in MAP to specify symbol to be printed for each cell.
MASS	Local	Used in PH2 for temporary storage of the mass of a cell.
MAXEXP	Local	Used in MAP to define logarithmic scale factor for each map.
MB	= Z(94)	Used and calculated in SETUP. The J-index of the bottom row of projectile.
MC	= Z(91)	Used and calculated in SETUP. The J-index of the top row of projectile.
MD	Local(C)	Used in SETUP. Flag indicating whether or not there is a target.

ME	Local[c]	Used and calculated in SETUP. The number of columns in the target. (If target extends beyond mesh, ME = IMAX.)
MINEXP	Local	Used in MAP to define logarithmic scale factor for each map.
MR	= Z(92)	Used and calculated in SETUP. The number of columns in the projectile.
MSLAVE	Local	Used in PH2 as storage for slaved-cell index when transporting mass across top edge of cell.
MZ	= Z(93)	Used and calculated in SETUP. The J-index of the bottom row of the target.
MZT	Global	Defined in INPUT (MZT = 150). Used in SETUP and EDIT. The number of Z-block words.
N	Local	Used as an index in PH3, PH1, PH2, REZONE and EDIT. In SETUP, N is the J-index of the top row of the target.
N3	= Z(53)	Defined in SETUP. Used in SETUP, EDIT, INPUT in reading and writing tapes. = 0 if there are no tracer points; = 1 if tracer points are used.
N6	= Z(56)	INPUT parameter. Used in ES. Negative pressures are allowed in cells above J = N6 after active-J reaches JSTR value. The value of N6 is reset in REZONE. N6 = 0 allows negative pressures everywhere. On the other hand, to make sure that negative pressures are always set to zero give N6 a very large value (many times as large as JMAX) since in REZONE N6 is cut in half in order to keep it at the same distance (in cm.) from the bottom of the grid.
N10	Global	Used in CDT to identify I-index of cell which controls DT.
N11	Global	Used in CDT to identify J-index of cell which controls DT.

NC	= Z(30)	Cycle number. Set initially to -1 in INPUT. Incremented thereafter in CDT.
NDUMP7	= Z(6)	INPUT parameter. Used in EDIT to control frequency of tape dumps. A tape dump will occur every (NDUMP7) EDIT prints.
NECYCL	= Z(77)	Defined and printed in EDIT. The cycle number associated with the largest relative error in the energy sum.
NERR	Global	Used in ERROR as exit flag. Prevents ERROR from being called more than once during a single run.
NFRELP	= Z(5)	INPUT parameter. Used in EDIT to control frequency of "long" prints. A "long" print will occur every (NFRELP) "short" prints.
NIMAX	Local	Used in REZONE as storage for IMAX/2 when IMAX > 1. NIMAX = 1 when IMAX = 1.
NIMAX1	Local	Used in REZONE as storage for NIMAX + 1.
NJMAX	Local	Used in REZONE as storage for JMAX/2.
NJMAX1	Local	Used in REZONE as storage for NJMAX + 1.
NK	Global	Used in PH2, PH3, EDIT, INPUT, CDT, REZONE and ERROR. Tells which statement of a subroutine caused ERROR to be called.
NKA	Local	Used in PH3 as index.
NKB	Local	Used in PH3 as index.
NMP	= Z(80)	Number of tracer points in use. Used in INPUT, SETUP, REZONE, EDIT and PH2. Initially calculated in SETUP; recalculated in REZONE.
NMPMAX	= Z(85)	INPUT parameter. Maximum number of tracer points to be generated. If fewer points are needed, NMP will have the number actually generated. NMPMAX must not be larger than the number allowed in dimensions of XP and YP. Used in SETUP and REZONE.

NN	Local	Used as index in PH3 and PH2.
NODUMP	= Z(96)	INPUT parameter. Used in EDIT. When NODUMP = 1, no tape dumps are made except on cycle 0. Allows user to restart a problem without writing on the restart tape.
NPLACE	Local	Used in REZONE for lining up added tracer points with original ones.
NPRINT	Global	Used in MAIN, CDT and EDIT. Prevents DT and PRTIME from being altered on intermediate prints. Also, NPRINT = 1 flags EDIT to print and check energy discrepancy.
NR	Global	Used in PH2, EDIT, SETUP and CDT to identify which subroutine called ERROR. Used in ERROR for printing error message.
NRC	= Z(32)	Used in PH1 and PH2 in advancing active grid.
NREZ	= Z(20)	Defined in SETUP. Equals maximum number of rezones allowed. Used in REZONE to line up new tracer points with those already in grid. Used in EDIT to determine the original I and J of each tracer point.
NRT	Local	Used in PH1 and PH2 in advancing active grid.
NRZ	Global	Initialized in SETUP. Equals number of rezones so far performed. Used in EDIT for printout of 1-D problems and for determining the original I and J of each tracer point.
NSLAVE	Local	Used in PH2 as storage for slaved-cell index when transporting mass across right edge of cell.
NULLE	Global	Equivalenced to RHOW in CDT and ES.
NUMREZ	= Z(12)	INPUT parameter. Initially equals number of rezones allowed in one run. Diminished by one after each rezone. Used in EDIT and REZONE.

NUMSCA	= Z(43)	INPUT parameter. Number of times the print interval is to be rescaled. Used in EDIT. See PRDELT for further details.
NUMSP	= Z(4)	Used in EDIT to count the number of "short" prints since the last "long" print.
NUMSPT	= Z(41)	Used in EDIT to count the number of prints (short and long) since the last tape dump.
NZ	= Z(19)	Defined and used in EDIT for 1-D problems. NZ = 4**NRZ. After rezoning the grid NZ is used to scale the values printed by EDIT for the total mass, energy and momentum.
P	Global	Cell-pressure. IMAX by JMAX array. Calculated in ES. Used by PH1. The P-storage space is used for UK, VK, and RHO storage in PH3. The P-array is initialized at the beginning of PH3.
P1	Local	Used in ES as storage for various terms in the pressure equations.
P2	Local	
P3	Local	
P4	Local	
P5	Local	
P6	Local	
P8	Local	
P9	Local	
P12	Local	
PABOVE	Local	Used in PH1 as storage for pressure at top of cell.
PBLO	Local	Used in PH1 as storage for pressure at bottom of cell.
PIDTS	Local(C)	Defined and used in PH1 as 1./(π*DT*DY). Defined and used in PH2 as 1./(π*DT).
PIDY	= Z(8)	Defined in INPUT: = π. Used in REZONE, SETUP, PH1, PH2, and PH3.
PK	Global	Used in SETUP and INPUT for defining input parameters. (See Appendix A.) PK(3) used in EDIT to signal a "long" or "short" print on first cycle of a restart run.

PL	Local(C)	Used in PH1 for saving pressures on left side of cell. Used in EDIT for crater depth printout. Equivalenced in standard OIL-RPM as follows: PL = RST (for PH3) PL = PR = GAMC (for PH2) PL(103) = SIGC (for PH2)
PMIN	= Z(86)	Used as a pressure cut-off. Calculated and printed in CDT as $(C_0) \cdot (\rho_0) \cdot (U_{min})$. Initially defined in INPUT as 10^6.
PR	Local(C)	Used in INPUT and EDIT for temporary storage.
PRAMOA	Local	Printed and calculated in EDIT. The positive radial momentum above JPROJ. Equivalenced to PR(8) in EDIT only.
PRAMOB	Local	Printed and calculated in EDIT. The positive radial momentum below JPROJ. Equivalenced to PR(16) in EDIT only.
PRDELT	= Z(45)	INPUT parameter. Gives the initial time interval between EDIT prints. There are five parameters which control printing frequency: PRDELT, IPCYCL, PRLIM, PRFACT, and NUMSCA. If the user is printing on time (PRDELT $\neq$ 0. and IPCYCL = 0.), DT will be adjusted so that a print will occur exactly every PRDELT seconds. If the user is printing on cycles (PRDELT = 0., IPCYCL $\neq$ 0.), a print will occur every IPCYCL cycles. PRLIM, PRFACT and NUMSCA are used to increase the print interval. PRLIM is the time (or cycle) at which PRDELT (or IPCYCL) and PRLIM are multiplied by PRFACT. The new value of PRLIM establishes the next time (or cycle) when the print interval will be rescaled. This process continues at most NUMSCA times. EXAMPLE: You wish to print every 1×10^{-8} sec. until you reach 1×10^{-7} sec., then every 1×10^{-7} sec. until 1×10^{-6} sec. and every 1×10^{-6} sec. thereafter: PRDELT = $1. \times 10^{-8}$ PRFACT = 10. PRLIM = $1. \times 10^{-7}$ NUMSCA = 2.

PRFACT	= Z(46)	INPUT parameter. Used in EDIT as a factor for rescaling PRDELT, IPCYCL, and PRLIM when PRLIM-time or cycle is reached. (See PRDELT.) Should be > 1.
PRLIM	= Z(44)	INPUT parameter: time or cycle at which to rescale PRDELT (or IPCYCL) and PRLIM by PRFACT. (See PRDELT.)
PRMAS	Local(C)	Used and calculated in EDIT. Total mass below JPROJ. Equivalenced to PR(12) in EDIT only.
PRMV	Local(C)	Used and calculated in EDIT. Total axial momentum below JPROJ. Equivalenced to PR(12) in EDIT only.
PRMVP	Local(C)	Used and calculated in EDIT. Total positive axial momentum below JPROJ. Equivalenced to PR(14) in EDIT only.
PROB	= Z(1)	INPUT parameter. Identifying problem number. Used in EDIT and INPUT.
PROJI	= Z(16)	INPUT parameter. Initial specific internal energy of projectile. Used in SETUP and REZONE.
PROJU	= Z(73)	INPUT parameter. Initial radial velocity of projectile. (Usually = 0.) Used in SETUP and REZONE.
PROPI	Local	Calculated and used in EDIT. For 1-D problems the totals for energy, mass, momentum per unit area are printed. (i.e., they are divided by 4** (Number of rezones)) and stored in PROPI for printing.
PRR	Local	Used in PH1 as temporary storage of pressure and pressure averages.
PRTIME	= Z(131)	Initially set to PRDELT in INPUT. Thereafter calculated in EDIT. When T = PRTIME, it is time to print.
PRXRT	= Z(69)	INPUT parameter. The outer radius of projectile (in cms.). PRXRT must be at a cell-boundary. Used in SETUP and REZONE.
PRYBOT	= Z(67)	INPUT parameter. Y-value of bottom of projectile (in cms.). PRYBOT should be at a cell-boundary. If no "projectile" is to be generated, PRYBOT should be set to -1. Used in SETUP and REZONE.

PSCLE	Local	Used in MAP as a logarithmic scale factor for the pressure maps.
PTEMP	Local	Used in CDT when calculating JPM array (the maximum pressure location in each column).
PW	Local	Used in PH3 to calculate plastic work when INTER = 99.
RADIUS	= PK(12) = Z(162)	INPUT parameter. Radius of sphere (in cms.). The radius need not be a multiple of DX. Used in SETUP. Equivalenced to PK(12) in SETUP.
RAMOMA	Local	Printed and calculated in EDIT. Total radial momentum above JPROJ. Equivalenced to PR(7) in EDIT only.
RATIO	Local	Used in CDT in calculation of DT. Ratio of $(DX,DY)_{min}$ to $(U,V,$ local sound speed$)_{max}$.
RC	Local	Used and calculated in PH1 as distance from axis to center of a cell.
RELERR	Local	Used in EDIT for storing and printing maximum relative error in the energy sum.
REZ	= Z(95)	Flag defined in PH2 and used in EDIT. Signals when the REZONE subroutine should be called. (The rezone flag is turned on when material begins to flow out through transmittive boundaries. In REZONE each set of four cells in the mesh is made into one cell. The new mass is the sum of masses in the four original cells. Momentum and total energy are conserved but in so doing some kinetic energy is changed to internal. (The result is that rezoning has a stabilizing influence.) When all permitted rezones have been done, material is allowed to flow out through transmittive boundaries and the mass and energy are lost from the system.
REZFCT	= Z(71)	INPUT flag for rezoning. If = 1., the grid is rezoned (NUMREZ) times. If = 0., no rezoning is done. Tested in PH2 and EDIT.

RHINI	= Z(111)	INPUT parameter. Initial density of projectile. Used in SETUP and REZONE. In PH2 it is used to determine whether material should be evaporated. (See EVAP)
RHINIT	= Z(15)	INPUT parameter. Initial density of target. Used in SETUP and REZONE.
RHO	Local(c)	Used in PH3 for temporary storage of cell density.
RHOFIL	= Z(51)	INPUT parameter. Initial density of filler material between the projectile and target. Used in SETUP and REZONE.
RHOMIN	= Z(138)	INPUT parameter. Cells with ρ < RHOMIN are by-passed in calculation of DT.
RHOOUT	= Z(103)	INPUT parameter. Used in SETUP. In cells containing sphere boundary RHOOUT is the density of material outside sphere. Equivalenced to Z(103) and EVAPMV in SETUP.
RHOSPH	= Z(100)	INPUT parameter. Used in SETUP as the initial density of sphere. Equivalenced to Z(100) and EVAPM in SETUP.
RHOW	Global	Density of cell. Calculated in CDT, used in ES. Equivalenced to NULLE in CDT and ES.
RHOZ	= Z(115)	INPUT parameter. Normal density. Used in INPUT, CDT, ES, EDIT, PH2 and PH3.
ROEPS	= Z(110)	INPUT parameter. Round-off epsilon used in calculating cutoffs. Used in CDT to calculate UMIN. UMIN = (ROEPS)(|maximum u or v|) Used in SETUP, EDIT and PH2.
RR	Local	Used and calculated in PH1. Distance (cms.) from axis to center of cell on the right.
RTM	= Z(57)	Calculated in PH2. Printed in EDIT. Total mass lost out right side of grid.

RTMU	= Z(10)	Calculated in PH2. Printed in EDIT. Total radial-momentum lost out right side of grid.
RTMV	= Z(58)	Calculated in PH2. Printed in EDIT. Total axial-momentum lost out right side of grid.
SAVEK	Local	Used and calculated in PH2. Factor used in calculation of energy fluxes across right and top boundaries of cells on reflective boundaries.
SIEMIN	Global	Used in MAP, PH2. Calculated in CDT. SIE cut-off value = $(UMIN)^2$.
SIENEW	Local	Used and calculated in PH2. New value of specific internal energy.
SIESPH	= Z(101)	INPUT parameter. Initial value of specific internal energy of sphere. Equivalenced to Z(101) and EVAPEN in SETUP.
SIGC	Local(C)	Used in PH2 for energy carried by mass moving across left side of cell. Equivalenced to PL(103). (See Appendix B.)
SIGMU	Local	Used in PH2 for radial momentum moving across cell-boundaries.
SIGMV	Local	Used in PH2 for axial momentum moving across cell-boundaries.
SN	= Z(65)	INPUT flag: When = 0. negative internal energy is to be set to 0. When = 1. negative internal energy is to be left alone. Used in PH1 and PH2.
SNB	Local(C)	Used in PH3 for normal stress at bottom of a cell. Equivalenced to P(313).
SNL	= Z(105)	Used in PH3 for normal stress at left of a cell.
SNLX	Local	Used in PH3. = SNL * X(I-1).
SNR	Local(C)	Used in PH3 for normal stress at right of a cell. ($=S_{11}$ at right.)
SNT	Local(C)	Used in PH3 for normal stress at top of a cell. ($= S_{22}$ at top.)

SOLID	Global	Calculated in INPUT as (RHOZ * AMDM). Used in PH2, PH3 and CDT.
SRATIO	Local	Used in CDT to calculate DT. The smallest ratio of minimum cell dimension to maximum velocity.
SS1	= Z(127)	Calculated in INPUT. Used in ES: = 1./(ESESP-ESES).
SS2	= Z(128)	INPUT parameter controlling reflective (and axis) boundary treatment. Usually = 1. Used in PH2.
SS4	= Z(130)	INPUT parameter. If SS4 $\neq$ 0., REZONE is called on second cycle of run. Used in EDIT.
STAB	= Z(139)	INPUT parameter. Used in CDT. Initial value of "stability fraction" for the calculation of DT. If FINAL = 0., STAB is constant. Otherwise its value progresses from STAB to FINAL in a geometric progression.
STB	Local(C)	Used in PH3 for shear stress at bottom of cell. Equivalenced to P(365).
STEZ	= Z(29)	INPUT parameter: E_0. Used in yield-strength calculation in PH3. See STRENG.
STK1	= Z(11)	INPUT parameter: Y_1. Used in yield-strength calculation in PH3. See STRENG.
STK2	= Z(28)	INPUT parameter: Y_2. Used in yield-strength calculation in PH3. See STRENG.
STL	= Z(106)	Used in PH3 for shear stress at left of cell.
STLX	Local	Used in PH3. = STL * X(I-1).
STR	Local(C)	Used in PH3 for shear stress at right of cell.
STRENG	Local	Calculated and used in PH3: yield strength of material. (Y_0, Y_1, Y_2 supplied by user.) $$\text{STRENG} = (Y_0 + Y_1\mu + Y_2\mu^2) \cdot (1 - \frac{E}{E_0})$$

		If STRENG < 0., stresses are set to 0. If $E > E_0$, STRENG = 0. Y_0 is YZERO, Y_1 is STK1, Y_2 is STK2, ρ_0 is RHOZ, E is AIX of cell, E_0 is STEZ, ρ is density of cell, $\mu = \rho/\rho_0 - 1$.
STT	= Z(143)	Used in PH3 for shear stress at top of cell.
SUM	Local(c)	Used in PH2. Sums negative internal energy when negative internal energy is set to zero.
SUME	Local	Used in PH2. Sums energy changes.
SUM1	Local	Used in SETUP to define JRADA.
SUM2	Local	Used in SETUP to define JRADB.
T	= Z(84)	Time in seconds. Initially defined in INPUT. Incremented in CDT. Adjusted in EDIT for printing. Printed by CDT, EDIT, REZONE.
TARGI	= Z(72)	INPUT parameter. Initial value of specific internal energy of target. Used in SETUP and REZONE.
TARGV	= Z(52)	INPUT parameter. Initial value of axial-velocity of target. Used in SETUP and REZONE.
TARMAS	Local(c)	Used in EDIT. Total mass above JPROJ. Equivalenced to PR(4) in EDIT only.
TARMV	Local(c)	Used in EDIT. Total axial momentum above JPROJ. Equivalenced to PR(5) in EDIT only.
TARMVP	Local(c)	Used in EDIT. Total positive axial momentum above JPROJ. Equivalenced to PR(6) in EDIT only.
TAU	Global	Calculated in SETUP and REZONE. Area of cell face: $= \pi(X(I)^2 - X(I-1)^2)$. Used in most subroutines.

TAUDTS	Local	Used and calculated in PH1: = TAU*DT.
TAXRT	= Z(107)	INPUT parameter. Outer radius of target (in cms.). TAXRT should be at a cell-boundary. Used in SETUP and REZONE.
TAYBOT	= Z(88)	INPUT parameter. Axial location of bottom of target (in cms). TAYBOT should be at a cell-boundary. If no "target," TAYBOT = -1. Used in SETUP and REZONE.
TAYTOP	= Z(89)	INPUT parameter. Axial location of top of target (in cms). TAYTOP should be at a cell-boundary. Used in SETUP and REZONE.
TEPRO	Local(c)	Used in EDIT. Total energy below JPROJ. Equivalenced to PR(11) in EDIT only.
TESTRH	Global	Calculated in INPUT: = (.2)*RHOZ. Used in CDT in defining pressure scale factor.
TETAR	Local(c)	Used in EDIT. Total energy above JPROJ. Equivalenced to PR(3) in EDIT only.
THO3	Local	Used and calculated in PH3: = $\frac{1}{3}\left(\frac{\partial u}{\partial r} + \frac{\partial r}{\partial z} + \frac{u}{r}\right)$.
TIEPRO	Local(c)	Used in EDIT. Total internal energy below JPROJ. Equivalenced to PR(9) in EDIT only.
TIETAR	Local(c)	Used in EDIT. Total internal energy above JPROJ. Equivalenced to PR(1) in EDIT only.
TKEPRO	Local(c)	Used in EDIT. Total kinetic energy below JPROJ. Equivalenced to PR(10) in EDIT only.
TKETAR	Local(c)	Used in EDIT. Total kinetic energy above JPROJ. Equivalenced to PR(2) in EDIT only.
TMA	Local	Used in MAP to associate a given density with the printed symbol.
TNOW	Local	Used in EDIT: = time now; saved when EDIT calls CDT after calling REZONE.
TOPM	= Z(63)	Calculated in PH2. Printed in EDIT. Total mass lost out top of grid.

TOPMU	= Z(9)	Calculated in PH2. Printed in EDIT. Total radial-momentum lost out top of grid.
TOPMV	= Z(66)	Calculated in PH2. Printed in EDIT. Total axial-momentum lost out top of grid.
TOTSPH	Local	Used in SETUP. Total volume of sphere.
TRIAL	Local	Used in CDT. Maximum sound-speed or velocity used to define DT.
TRNSFC	= Z(145)	Defined in INPUT: = .4. Used in PH2 to define maximum transport velocity. (See UVMAX.)
TSTOP	= Z(50)	INPUT parameter. Value of T at which execution stops when stopping on time rather than cycles.
TWOPDT	Local	Calculated and used in PH2: = 2*π*DT.
TWOPI	Global	Calculated in INPUT. Used in PH3: = 2π.
U	Global	Radial velocity of cell. (IMAX by JMAX array.)
UAMMP	Local	Used in PH2 for U of mass moving across right cell-edge.
UAMPY	Local	Used in PH2 for U of mass moving across top cell-edge.
UBAR	Local	Used in PH3 in calculating energy sum.
UBEF	Local	Used in PH2 to move tracer-points.
UK	Local(C)	Used in PH3. Temporary storage for part of U array.
UKT	Local	Used in PH3. Temporary storage for U(K).
UL	Local(C)	Used in PH1 for U on left. Used in EDIT for crater depth printout. Equivalenced as follows in standard OIL-RPM: UL(103) = CRAD (for EDIT) UL = RSH (for PH3) UL = FLEFT (for PH2) UL(103) = YAMC (for PH2)
UMIN	= Z(129)	Calculated in CDT. Used as velocity cutoff in PH2, PH3, and MAP: = (ROEPS)*(maximum sound-speed or velocity).

UMK	Local	Calculated and used in PH2. Temporary storage for U(K)*AMX(K).
UMKP	Local	Calculated and used in PH2. Temporary storage for U(K+1)*AMX(K+1).
UMOM	Local	Calculated and used in PH2: = MASS*U(K).
UNxxx	= Z(xxx)	Unused Z-storage.
UNEW	Local	Calculated and used in PH2. New value of U.
UOTK	Local	Calculated and used in PH2: = UMK/TAU(I).
UOTKP	Local	Calculated and used in PH2: = UMKP/TAU(I+1).
UOX	Local	Used in PH3: = U/X.
URR	Local(c)	Used in PH1 and PH2. Temporary storage for velocity and velocity averages.
USCLE	Local	Used in MAP as logarithmic scale factor of radial velocity map.
UVMAX	= Z(22)	Used and calculated in PH2. Maximum transport velocity. In radial direction UVMAX = TRNSFC*DX(I)/DT. In axial direction UVMAX = TRNSFC*DY(J)/DT.
V	Global	Axial velocity of cell. (IMAX by JMAX array.)
VABOVE	Local	Used in PH1 and PH2 as storage for velocity at top of cell.
VALUE	Local	Used in MAP in printing maximum value of each map symbol.
VAMMP	Local	Used in PH2 for axial velocity of mass moving across right cell-edge.
VAMPY	Local	Used in PH2 for axial velocity of mass moving across top cell-edge.
VBAR	Local	Used in PH3 in calculating energy sum.
VBLO	Local	Used in PH1 as storage for velocity at bottom of cell.
VEFF	Local	Used in PH2 to move tracer points.

VEL	Local	Used in PH1 as subcycle flag.
VFACT	Local	Used in PH3 in setting boundary conditions.
VINI	= Z(112)	INPUT parameter. Initial axial velocity of projectile. Used in SETUP and REZONE.
VINSPH	= Z(102)	INPUT parameter. Initial axial velocity of sphere. Used in SETUP. Equivalenced to Z(102) and EVAPMU in SETUP.
VK	Local(C)	Used in PH3. Temporary storage for part of V array.
VKT	Local	Used in PH3. Temporary storage for V(K).
VMK	Local	Calculated and used in PH2: = V(K)*AMX(K).
VML	Local	Calculated and used in PH2: = V(K+IMAX)*AMX(K+IMAX).
VNEW	Local	Calculated and used in PH2. New value of V.
VOLSPH	Local	Calculated and used in SETUP. Volume of toroid generated by a cell. Used in setting up sphere.
VOW	Local	Calculated and used in ES. = (normal density)/(density of cell).
VSCLE	Local	Used in MAP as logarithmic scale factor for axial velocity map.
VT	= Z(55)	INPUT parameter. Used in PH2 as minimum mass flux (across top or right side of one boundary cell in one cycle) needed to trigger a rezone. Usually $VT \sim \rho_0 \times 10^{-4}$.
WDYF	Local	Used in PH2 as temporary storage for DY(J) or DY(J+1).
WFLAGF	Gobal	Used in INPUT and EDIT. Flags first cycle. Set = 1. in INPUT. Set = 0. in EDIT.
WFLAGL	Global	Used in MAIN and EDIT. Flags last cycle. Set = 1. in EDIT.
WFLAGP	Local(C)	Used in EDIT. Flags beginning of printout of properties for each cell in a given column.

WS	Local(C)	Used in most subroutines for local working storage.
WSA	Local(C)	Used in most subroutines for local working storage.
WSB	Local(C)	Used in most subroutines for local working storage.
WSC	Local(C)	Used in most subroutines for local working storage.
WSD	Local	Used in PH2 local working storage.
WSMAX	Local	Used in MAP to define scale factors.
WSMIN	Local	Used in MAP to define scale factors.
WSOUT	Local	Used in PH2 for adjusting over-emptied cells.
WT	Local	Used in CDT for local working storage.
WTA	Local	See WT.
WTB	Local	See WT.
X	Global	Distance (cms) from axis to outside of cell. Equivalenced to XX array such that X(0) = XX(1).
XIENRG	= Z(140)	Total internal energy. Calculated in EDIT and used for printing labels on tracer point plots.
XKENRG	= Z(141)	Total kinetic energy. Calculated in EDIT and used for printing labels on tracer point plots.
XL2	Local	Calculated and used in SETUP for placing sphere: $= (X(I-1))^2$.
XMAX	= Z(18)	Calculated in SETUP: = X(IMAX).
XP	Global	Tracer-point x-coordinates. Used in INPUT, SETUP, EDIT, PH2 and REZONE.
XR2	Local	Calculated and used in SETUP for placing sphere: $= (X(I))^2$.
XTENRG	= Z(142)	Total energy. Calculated in EDIT and used for printing labels on tracer point plots.
XUM	Constants	Used in MAP. Has negative alphabetic characters for maps. Defined in DATA statement.

XX	Global	Equivalenced to X array so as to make X(0) available.
Y	Global	Distance (cms) from bottom of grid to top of cell. Equivalenced to YY array such that Y(0) = YY(1).
Y2	= Z(81)	INPUT tracer point flag: when = -2, tracer points are calculated; when = 0, tracer points not calculated.
YAMC	Local[c]	Calculated and used in PH2. Axial momentum of mass moving across 1 ft side of cell. Equivalenced to UL(103). (See Appendix B)
YBOTTM	Local	Calculated and used in SETUP in placing sphere: = Y(J-1).
YC2	Local	Calculated and used in SETUP in placing sphere: = $(YCENTR)^2$.
YCENTR	= PK(13) = Z(163)	INPUT parameter. Distance (cms) of center of sphere from bottom of grid. YCENTR must be at a cell-boundary. Used in SETUP. Equivalenced to PK(13) in SETUP only.
YDIFFB	Local	Calculated and used in SETUP in placing sphere.
YDIFFI	Local	Calculated and used in SETUP in placing sphere.
YDIFFO	Local	Calculated and used in SETUP in placing sphere.
YDIFFT	Local	Calculated and used in SETUP in placing sphere.
YLINTA	Local	Calculated and used in SETUP in placing sphere.
YLINTB	Local	Calculated and used in SETUP in placing sphere.
YLOWER	Local	Calculated and used in SETUP in placing sphere.
YMAX	Local	Calculated in SETUP: = Y(JMAX).
YP	Global	Tracer-point y-coordinates. Used in INPUT, SETUP, EDIT, PH2 and REZONE.
YRINTA	Local	Calculated and used in SETUP in placing sphere.
YRINTB	Local	Calculated and used in SETUP in placing sphere.
YTOP	Local	Calculated and used in SETUP in placing sphere.

YUPPER	Local	Calculated and used in SETUP in placing sphere.
YY	Global	Equivalenced to Y array so as to make Y(0) available.
Z	Special Global	Storage for most of the input parameters. The Z-array (150 words) is written on tape for restarts. Used in INPUT, MAIN, and SETUP. (See Appendix A)

APPENDIX A
Z-STORAGE LISTED NUMERICALLY

See Dictionary for meaning and use.

*1.	PROB	30.	NC	59.	UN59
2.	CYCLE	31	UN31	60.	N10
3.	DT	32.	NRC	61.	N11
4.	NUMSP	2*33.	IMAX	62.	GAMMA
2*5.	NFRELP	34.	IMAXA	63.	TOPM
2*6.	NDUMP7	2*35.	JMAX	64.	BOTMU
2*7.	ICSTOP	36.	JMAXA	*65.	SN
8.	PIDY	37.	KMAX	66.	TOPMV
9.	TOPMU	38.	KMAXA	*67.	PRYBOT
10.	RTMU	39.	BOTM	*68.	PRYTOP
*11.	STK1	40.	BOTMV	*69.	PRXRT
2*12.	NUMREZ	41.	NUMSPT	*70.	CYCPH3
13.	ETH	*42.	CZERO	*71.	REZFCT
14.	UN14	2*43.	NUMSCA	*72.	TARGI
*15.	RHINIT	*44.	PRLIM	*73.	PROJU
*16.	PROJI	*45.	REDELT	74.	BBOUND
17.	UN17	*46.	PRFACT	*75.	EVAP
18.	XMAX	2*47.	I1	76.	ECK
19.	NZ	2*48.	I2	77.	NECYCL
20.	NREZ	2*49.	IPCYCL	78.	II
*21.	AMDM	1*R 50.	TSTOP	79.	IJ
22.	UVMAX	*51.	RHOFIL	80.	NMP
23.	UN23	*52.	TARGV	*81.	Y2
*24.	DMIN	53.	N3	82.	EZPH1
2*25.	JSTR	2*54.	IVARDY	2*83.	IVARDX
26.	DTNA	55.	VT	84.	T
*27.	CVIS	2*56.	N6	2*85.	NMPMAX
*28.	STK2	57.	RTM	86.	PMIN
*29.	STEZ	58.	RTMV	2*87.	INTER

---

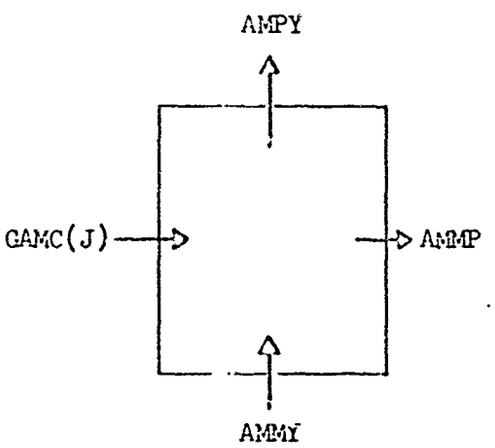
* User-supplied input-values.
2* Must have a "2" in column 1.
1* Must have a "1" in column 1.
R Must be included in a restart input input deck.

*88.	TAYBOT
*89.	TAYTOP
90.	IEMAP
91.	MC
92.	MR
93.	MZ
94.	MB
95.	REZ
2*96.	NODUMP
97.	UN97
98.	UN98
99.	UN99
*100.	EVAPM(RHOSPH)
*101.	EVAPEN(SIESPH)
*102.	EVAPMU(VINSPH)
*103.	EVAPMV(RHOOUT)
104.	EZPH2
105.	SNL
106.	STL
*107.	TAIRT
108.	IDNMAP
109.	IPRMAP
*110.	ROEPS
*111.	RHINI
*112.	VINI
*113.	FINAL
114.	IVMAP
*115.	RHOZ
*116.	ESA
*117.	ESEZ
*118.	ESB
*119.	ESCAPA
*120.	ESESP
*121.	ESESQ
*122.	ESES
*123.	ESALPH
*124.	ESBETA
*125.	ESCAPB
126.	IUMAP
127.	SS1
128.	SS2
129.	UMIN
*130.	SS4
131.	PRTIME
132.	EOR
133.	EOT
134.	EOB
135.	EMOR
*136.	DXF
*137.	DYF
*138.	RHOMIN
*139.	STAB
140.	XIENRG
141.	XFENRG
142.	XTENRG
143.	STT
*144.	DTMIN
145.	TRNSFC
146.	EMOT
2*147.	JPROJ
148.	CNAUT
*149.	BBAR
1*150.	EMOB = 0 (Last card of input unless restarting from tape.)
	PK array follows the Z array in Blank Common; so PK(1) from the beginning of Blank Common is equivalent to Z(151).
1*R151.	PK(1) should be the same as PROB.
*R152.	Cycle to restart on.
*R153.	PK(3): When = -1., program will restart from tape and do a "long" EDIT print of the pickup cycle. When = -2., program will restart from tape and do a "short" EDIT print of the pickup cycle.
*162.	Radius of sphere (RADIUS).
*163.	Y-center of sphere (YCENTR).
1*164.	When Z(164) > 0. CARDS will be called by SETUP to read "special" input. (See "OIL-RPM Input for Special Setup" on page 23.)

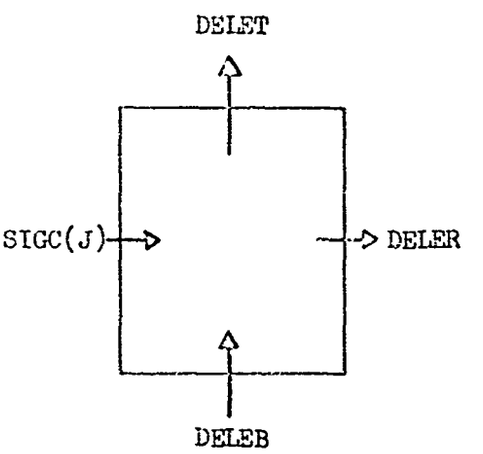
APPENDIX B

VARIABLES USED FOR FLUXES ACROSS CELL BOUNDARIES

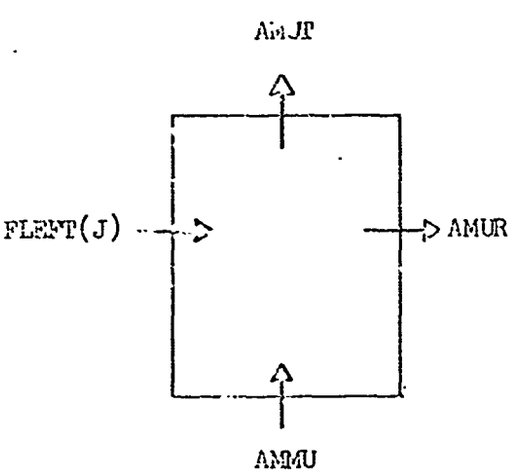
Mass



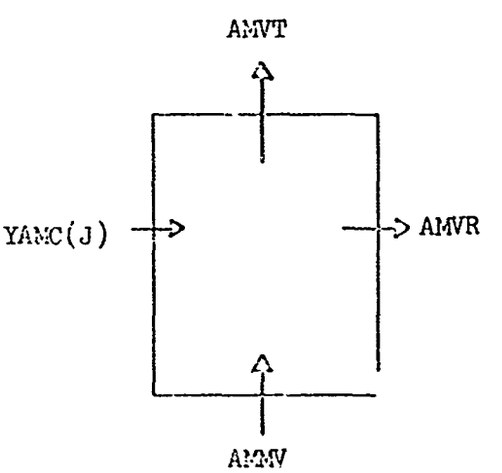
Energy



Radial Momentum



Axial Momentum



REFERENCES

1. Evans, M.W. and F.H. Harlow, "The Particle-in-Cell Method for Hydrodynamics Calculations," Los Alamos Scientific Laboratory Report LA-2139, November 1957.

2. Johnson, W.E., "Computer Development to Improve SHELL Code," General Atomic Report GA-4673, Contract AF29(601)-6028, October 1963.

3. Johnson, W.E., "OIL, A Continuous Two-Dimensional Eulerian Hydrodynamic Code," General Atomic Report GAMD-5580, October 1964.

4. Tillotson, J.H., "Metallic Equations of State for Hypervelocity Impact," General Atomic Report GA-3216, July 1962.

5. Dienes, J.K., W.E. Johnson and J.M. Walsh, "Annual Status Report on the Theory of Hypervelocity Impact," General Atomic Report GA-6509, June 1965.

6. Allen, R.T., "Equation of State of Rocks and Minerals," General Atomic Report GAMD-7834, March 1967.

7. Walsh, J.M., J.H. Tillotson and W.E. Johnson, "Theory of Hypervelocity Impact, Quarterly Progress Report," General Atomic Report GACD-4518, July 1963.

8. Dienes, J.K. and M.W. Evans, "Cratering Calculations with a Hydrodynamic Strength Code," General Atomic Report GAMD-7369, September 1966.