EPA-650/4-74-006

FEBRUARY 1974

Environmental Monitoring Series

A METHODOLOGY FOR TREATING LARGE LOCALIZED EMISSIONS OF REACTIVE POLLUTANTS

Office of Research and Development
U.S. Environmental Protection Agency
Washington, DC 20460

EPA-650/4-74-006

A METHODOLOGY FOR TREATING LARGE LOCALIZED EMISSIONS OF REACTIVE POLLUTANTS

by

A.J. Fabrick, P.I. Nakayama, and E.J. Fredricksen

JRB Associates, Inc.
1200 Prospect Street
La Jolla, California 92037

Contract No. 68-02-1238
Program Element No. 1AA009

EPA Project Officer: Herbert Viebrock

Meteorology Laboratory
National Environmental Research Center
Research Triangle Park, N.C. 27711

Prepared for

OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
WASHINGTON, D.C. 20460

February 1974

This report has been reviewed by the Environmental Protection Agency and approved for publication. Approval does not signify that the contents necessarily reflect the views and policies of the Agency, nor does mention of trade names or commercial products constitute endorsement or recommendation for use.

ABSTRACT

This report presents the results of a study to develop a numerical model to accurately calculate the trajectories and concentrations of reactive pollutants emitted from localized sources. The numerical model employs the solution of the full three-dimensional Navier-Stokes equations along with the solution of the species density equation by summing over Lagrangian mass points. The mass points are transported by the mean wind field and moved with a random walk technique that simulates the turbulent diffusion. The effect of chemical reactions are modeled by reweighting the Lagrangian mass points. The equations are solved for regions of space where terrain features or buildings may cause strong distortions in the flow field.

The numerical techniques are verified by comparisons with analytic formulas, including the boundary layer above a plate and the Gaussian plume. The method was used to calculate photochemically reacting plumes using a simplified inorganic photochemical reaction model.

It is concluded that the numerical model shows promise of being able to assess the impact on air quality of point source emission over complex terrain. Furthermore with the incorporation of a valid photochemical reaction kinetics module, the model developed during this study could be used to construct a parametric study of various plumes in order to quantify the errors introduced by regional models in averaging point sources over large areas.

CONTENTS

		Page
Abstract		iii
List of Figures		v
Acknowledgments		vi
Sections		
I	Conclusions	1
II	Recommendations	3
III	Introduction	5
IV	Solution of the Hydrodynamic Equations	7
V	Verification of Numerical Solutions	17
VI	Solution of the Density Equation Using Particles	23
VII	Verification of the Particle Method	27
VIII	References	37
IX	Appendices	39

FIGURES

No.		Page
1.	Location of variables in computational mesh.	9
2.	Flow diagram for hydrodynamics routine.	14
3.	Grid used for laminar boundary layer calculation	19
4.	Initial velocity field for laminar boundary layer calculation.	19
5.	Velocity vectors after 150 computational cycles.	19
6.	Horizontal velocity as a function of height for three locations along a flat plate.	20
7.	Computational grid for plate over notch calculation.	21
8.	Initial velocity field for flow over notch calculation.	21
9.	Velocity vectors after 21 computational cycles.	21
10.	Initial velocity field for V notch problem.	22
11.	Velocity vectors after 16 computational cycles.	22
12.	Particle locations during a VARMINT simulation of a Gaussian Plume with and without subcycle staggering.	28
13.	Comparison of VARMINT calculation and analytic solution for a point source.	29
14.	Comparison of VARMINT calculation and analytic solution for a point source with exponential decay.	31
15.	Schematic diagram for Eulerian-Lagrangian code.	32
16.	Calculation of NO_2 plume emitted into clear ambient air.	34
17.	Calculation of an equilibrium plume emitted into equilibrium ambient air.	35
18.	Nitrogen oxide source emitted into $\frac{1}{2}$ ppm ozone background.	36

ACKNOWLEDGMENTS

The authors would like to express their appreciation to C. W. Hirt and J. L. Cook for their pioneering work on the techniques employed under this contract and for their valuable technical assistance during their study.

This work was supported, in part, by the Environmental Protection Agency under contract (EPA-68-02-1238).

SECTION I

CONCLUSIONS

The transport and dispersion of reactive emissions from point sources has been modeled and incorporated into a numerical solution technique that was developed under this program. The technique shows promise of being able to assess the impact on air quality from point source emissions of reactive pollutants over complex terrain; a capability that has heretofore not been available.

With the incorporation of an accurate photochemical reaction kinetics module, the techniques outlined in this report can be used to estimate the errors introduced by smearing point source emissions over large areas in regional photochemical air quality simulation models.

(This Page Intentionally Left Blank)

SECTION II

RECOMMENDATIONS

As will be seen in the text, this program has demonstrated the feasibility of a numerical simulation of reactive plumes over irregular terrain. With the addition of an appropriate photochemical reaction kinetics module, the techniques outlined in this report could be used to estimate the errors introduced by smearing point source emissions over large areas in current regional photochemical air quality simulation models.

In order to utilize the methods developed for the calculation of the impact of localized sources in complex terrain several additional features must be added. Atmospheric density stratification and the dynamics of buoyant plumes must be incorporated. Turbulence models that specify eddy viscosity as a function of time and space should be incorporated. To allow greater generality in specifying terrain features arbitrary boundaries treatment should be developed to complement the existing obstacle blocks and half blocks already encoded.

(This Page Intentionally Left Blank)

SECTION III

INTRODUCTION

The importance of the role played by localized emissions on air quality in a large air basin is evident. A source inventory of the Los Angeles Air Basin, for example, shows that about 20% of the oxides of nitrogen are emitted from point as compared to area sources (mainly attributed to the automobile).[1] As automobile emission standards are made more stringent and as additional fossil fuel power plants are built to meet energy demands the percentage of air pollutants emitted from point sources will increase. Because the pollutants of interest are reactive and their reaction rates are a function of concentration, smearing point source emissions over a large area or prescribing an apriori analytical distribution of initial concentration may result in erroneous estimates of the effect of these emissions on air quality. In order to assess the magnitude of the errors resulting from the approximate treatment of point sources employed in current photochemical simulation models a methodology must be developed to correctly relate point source emissions of reactive pollutants to ambient air quality.

The methods developed under this contract solve two major problems associated with reacting plumes. These problems are the specification of the wind and turbulence fields over complex terrain and the solution of the species density equations, including chemistry. The solutions to these problems are respectively, the direct numerical calculation of the three dimensional wind field and a particle-random walk technique to solve the species density equation. In the former, the full Navier-Stokes equations with appropriate boundary conditions are used,

and while the turbulence is not calculated directly its effect on the mean flow and the dispersal of pollutants can be prescribed as a function of space and time by assuming the turbulence to be in a state of equilibrium where the creation of turbulent energy by velocity gradients and decay by molecular viscosity are in balance. In the latter, the density equation, including chemistry is solved by representing pollutant species as discrete particles that move within an Eulerian grid. This technique eliminates the numerical smoothing of diffusion of species concentration inherent in finite difference solutions of the density equation. The turbulent diffusion of pollutants is accomplished by a random walk technique imposed on individual particles and is independent of the Eulerian grid. The standard deviation of the random walk is scaled to the local turbulence intensity. Changes in concentration due to chemical reactions are accomplished by adjusting the masses of the individual particles. The techniques discussed above have been incorporated into a computer code, VARMINT. The discussion of this code is divided into four sections. The first two sections discuss the solutions of the Navier-Stokes equations and the verification of the numerical methods employed. The third section deals with the methods employed to solve the species density equation. The fourth section presents the verification of the particle method used to solve the species density equation and preliminary investigation of reactive plumes using a simplified photochemical reaction mechanism. A program users guide and listing are provided in appendices.

SECTION IV

SOLUTION OF THE HYDRODYNAMIC EQUATIONS

The VARMINT code solves the standard Eulerian incompressible equations using a variation of the Marker and Cell Technique which is similar to that employed by Hirt et. al.[2,3] A pressure iteration is utilized to insure the incompressibility condition. The governing equations are

$$\frac{\partial u_i}{\partial t} + \frac{\partial}{\partial x_j}(u_i u_j) = -\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_k}\sigma\frac{\partial u_i}{\partial x_k} + g_i \tag{1}$$

$$\frac{\partial u_j}{\partial x_j} = 0 \quad . \tag{2}$$

In Cartesian coordinates

$$\begin{aligned}
\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} + \frac{\partial uw}{\partial z} &= -\frac{\partial P}{\partial x} + \frac{\partial}{\partial x}\sigma\frac{\partial u}{\partial x} + \frac{\partial}{\partial y}\sigma\frac{\partial u}{\partial y} + \frac{\partial}{\partial z}\sigma\frac{\partial u}{\partial z} + g_x \\
\frac{\partial v}{\partial t} + \frac{\partial vu}{\partial x} + \frac{\partial v^2}{\partial y} + \frac{\partial vw}{\partial z} &= -\frac{\partial P}{\partial y} + \frac{\partial}{\partial x}\sigma\frac{\partial v}{\partial x} + \frac{\partial}{\partial y}\sigma\frac{\partial v}{\partial y} + \frac{\partial}{\partial z}\sigma\frac{\partial v}{\partial z} + g_y \qquad (3) \\
\frac{\partial w}{\partial t} + \frac{\partial wu}{\partial x} + \frac{\partial wv}{\partial y} + \frac{\partial w^2}{\partial z} &= -\frac{\partial P}{\partial z} + \frac{\partial}{\partial x}\sigma\frac{\partial w}{\partial x} + \frac{\partial}{\partial y}\sigma\frac{\partial w}{\partial y} + \frac{\partial}{\partial z}\sigma\frac{\partial w}{\partial z} + g_z
\end{aligned}$$

$$\frac{\partial u}{\partial y} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad . \tag{4}$$

Here P is the ratio of pressure to constant density, σ is a momentum eddy diffusivity, and g_x, g_y, g_z are prescribed body accelerations that arise from gravity, buoyency etc. The region in which computations are to be performed is divided into a set of small rectangular cells having edge lengths Δx, Δy, Δz, which need not be the same for

all cells. With respect to this set of computational cells, velocity components are located at cell faces and pressure values are at cell centers as shown in Fig. 1. Cells are labeled with an index (i, j, k), which denotes the cell number as counted from the origin in the x, y, and z directions, respectively.

A time-dependent solution is obtained by advancing the flow field variables through a sequence of time steps of duration Δt. The advancement through one time step is calculated in two stages. The first stage or tilde phase consists of a simple explicit calculation, in which the velocity components ($\tilde{u}$, $\tilde{v}$, $\tilde{w}$), are advanced using the previous state of the flow to calculate the accelerations caused by convection, diffusion, body forces, and pressure gradients. This explicit time advancement does not necessarily lead to a velocity field with zero divergence, i.e., to one that insures mass conservation. We can, however, insure incompressibility by adjusting the pressure in each cell in such a way that there is no net mass flow in or out of any cell. Adjusting the pressure in one cell will affect neighboring cells so that this adjustment must be performed iteratively until all cells have achieved a mass change not exceeding some small prescribed amount. The explicit finite difference equations for the tilde phase are given in Appendix A. These equations allow the user to specify doner cell or cell centered convective flux approximations or any combination of the two methods.

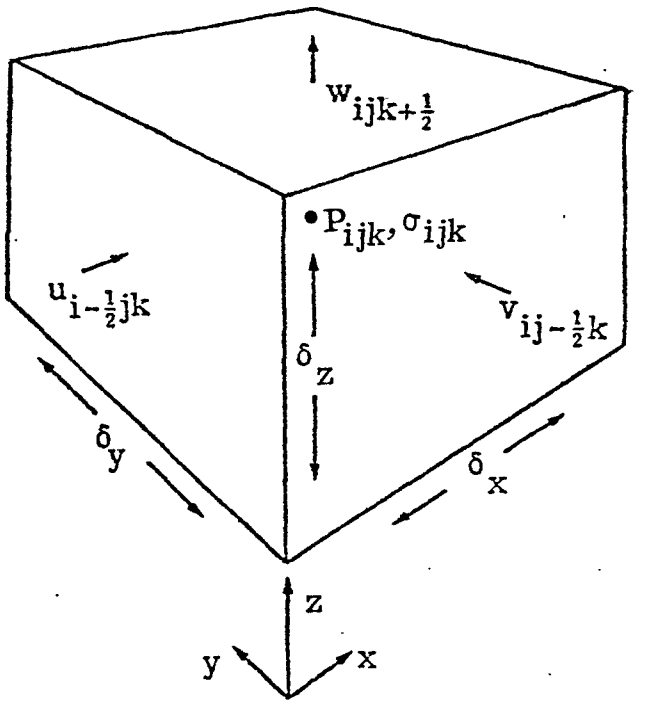


Figure 1. Location of variables in computational mesh.

BOUNDARY CONDITIONS

Several boundary condition options are available in VARMINT. These include no slip and free slip rigid walls, continuative output walls, and periodic boundaries. For simplicity VARMINT assumes all physical boundaries coincide with cell boundaries. Internal obstacles consisting of cells or half cells (diagonal) are also available with no slip and free slip boundary conditions.

The prescription of boundary conditions consists of a choice for both the normal and tangential velocities at the boundary. The normal velocity is easy to prescribe when the boundary coincides with a cell edge, since it is the normal velocity that is stored for each cell face. For a rigid boundary this velocity is set to zero. If the boundary is periodic the value must be chosen equal to the corresponding velocity one wavelength away. For outflow boundaries there is no unique prescription, but the general idea is to choose boundary conditions that have the least upstream influence. It has been found that for this purpose a useful prescription consists of setting the normal tilde velocity on the outflow boundary equal to the corresponding tilde velocity immediately upstream, and then letting the velocity on the boundary relax as it wishes during the pressure iteration.

The tangential velocities in cells immediately outside the fluid region are set equal to the adjacent velocities inside the fluid when it is desired that the boundary represent a free-slip wall, and they are set equal to the negative of the adjacent fluid velocities when the boundary is to be no slip. In other words, the external velocities tangent to a boundary are chosen to give either vanishing shear or vanishing velocity

at a rigid wall. If the boundary is periodic these external velocities are set equal to their counterparts one wavelength away. At an outflow boundary, they are set equal to the adjacent velocities inside the fluid.

The boundary conditions are set after the tilde velocities are calculated and after each pass through the pressure iteration.

The boundary conditions on internal obstacles are the same as for the actual mesh boundaries except that before each pass through the pressure iteration all velocities inside the obstacles are set to zero. This method allows for the correct stress boundary conditions during the tilde calculation and for no divergence boundary conditions during the pressure iteration.

PRESSURE ITERATION

Since tilde equations do not necessarily result in a velocity field that satisfies Eq. (4), some adjustment of the tilde velocities must be made to insure mass conservation.

The iterative process used to modify the tilde velocities is discussed below. In each cell (i, j, k) the value of the velocity divergence D is calculated as:

$$D_{ijk} = \frac{1}{\Delta x_i}\left(u_{i+\frac{1}{2}jk} - u_{i-\frac{1}{2}jk}\right) + \frac{1}{\Delta y_j}\left(v_{ij+\frac{1}{2}k} - v_{ij-\frac{1}{2}k}\right) + \frac{1}{\Delta z_k}\left(w_{ijk+\frac{1}{2}} - w_{ijk-\frac{1}{2}}\right) \quad . \tag{5}$$

The pressure is then changed by an amount

$$\delta P = -\beta D \tag{6}$$

where

$$\frac{1}{\beta} = \frac{1}{\beta_o}\frac{\partial D_{ijk}}{\partial P_{ijk}} = \frac{1}{\beta_o}\left[\frac{1}{\Delta x_i}\cdot\frac{2\Delta t}{(\Delta x_i+\Delta x_{i+1})} + \frac{1}{\Delta x_i}\cdot\frac{2\Delta t}{(\Delta x_i+\Delta x_{i-1})}\right.$$

$$+\frac{1}{\Delta y_j}\cdot\frac{2\Delta t}{(\Delta y_j+\Delta y_{j+1})} + \frac{1}{\Delta y_j}\cdot\frac{2\Delta t}{(\Delta y_j+\Delta y_{j-1})} + \frac{1}{\Delta z_k}\cdot\frac{2\Delta t}{(\Delta z_k+\Delta z_{k+1})}$$

$$\left.+\frac{1}{\Delta z_k}\cdot\frac{2\Delta t}{(\Delta z_k+\Delta z_{k-1})}\right]$$

or

$$\beta = \frac{\beta_o}{2\Delta t}\cdot\left[\frac{1}{\frac{1}{\Delta x_i}\left(\frac{1}{\Delta x_i+\Delta x_{i+1}} + \frac{1}{\Delta x_i+\Delta x_{i-1}}\right)} \cdots\right.$$

$$\frac{}{\frac{1}{\Delta y_j}\left(\frac{1}{\Delta y_j+\Delta y_{j+1}} + \frac{1}{\Delta y_j+\Delta y_{j-1}}\right)} \cdots$$

$$\left.\frac{}{\frac{1}{\Delta z_k}\left(\frac{1}{\Delta z_k+\Delta z_{k+1}} + \frac{1}{\Delta z_k+\Delta z_{k-1}}\right)}\right] . \tag{6a}$$

Here, β_o is a relaxation factor, where overrelaxation and underrelaxation correspond to β_o greater than or less than unity. For stability in the iteration, β_o must remain less than 2.

Once ΔP has been calculated it is necessary to add it to the total pressure P_{ijk} and to adjust the velocities:

$$
\begin{aligned}
u_{i+\frac{1}{2}jk} &\rightarrow u_{i+\frac{1}{2}jk} + \left(\frac{2\Delta t}{\Delta x_i + \Delta x_{i+1}}\right) \delta P \quad , \\
u_{i-\frac{1}{2}jk} &\rightarrow u_{i-\frac{1}{2}jk} - \left(\frac{2\Delta t}{\Delta x_i + \Delta x_{i-1}}\right) \delta P \quad , \\
v_{ij+\frac{1}{2}k} &\rightarrow v_{ij+\frac{1}{2}k} + \left(\frac{2\Delta t}{\Delta y_j + \Delta y_{j+1}}\right) \delta P \quad , \\
v_{ij-\frac{1}{2}k} &\rightarrow v_{ij-\frac{1}{2}k} - \left(\frac{2\Delta t}{\Delta y_j + \Delta y_{j-1}}\right) \delta P \quad , \\
w_{ijk+\frac{1}{2}} &\rightarrow w_{ijk+\frac{1}{2}} + \left(\frac{2\Delta t}{\Delta z_k + \Delta z_{k+1}}\right) \delta P \quad , \\
w_{ijk-\frac{1}{2}} &\rightarrow w_{ijk-\frac{1}{2}} - \left(\frac{2\Delta t}{\Delta z_k + \Delta z_{k-1}}\right) \delta P \quad . \qquad (7)
\end{aligned}
$$

This process is repeated successively in all cells until no cell has a magnitude of D greater than ϵ. Where ϵ is a function of the magnitude of the velocities calculated.

When the iteration has converged, the adjusted velocities satisfy the mass conservation condition (4), and this completes the necessary calculations for advancing the flow field through one cycle in time.

A flow diagram of the hydrodynamics routine of VARMINT is given in Figure 2.

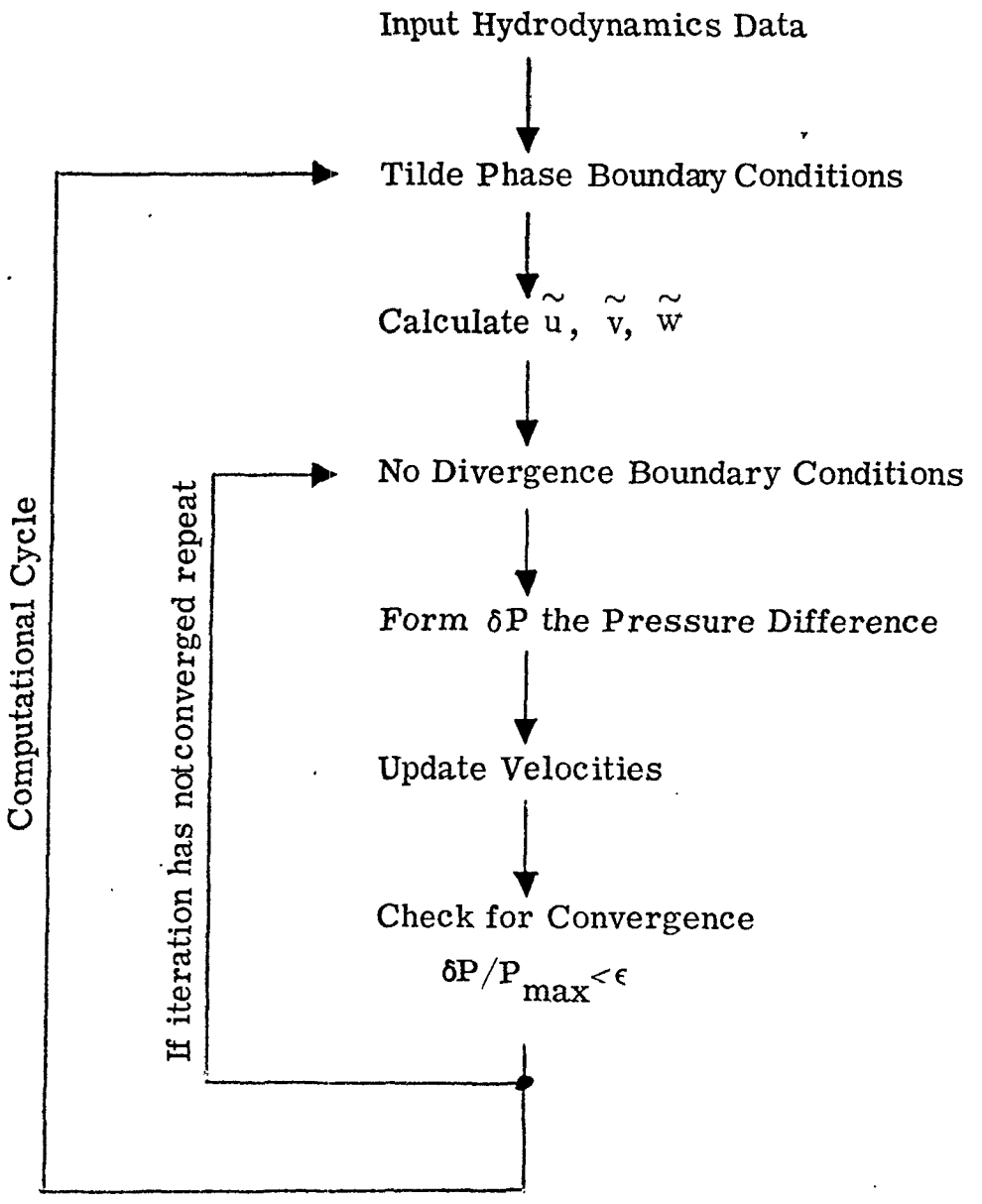


Figure 2. Flow diagram for hydrodynamics routine.

NUMERICAL STABILITY

The basic restriction on the size of the time step, Δt, is that the fluid must not be permitted to flow across more than one computation cell in one time step, that is

$$\Delta t < \min \frac{\Delta x}{|u|}, \frac{\Delta y}{|v|}, \frac{\Delta z}{|w|} \tag{8}$$

This is both a numerical accuracy condition (because the tilde equations assume exchanges between adjacent cells only) and a numerical stability condition.

The VARMINT code may be used either with donor cell or with centered difference for the convection term. When centered difference is used, the equations will be unstable unless the kinematic viscosity, σ, is approximately

$$\sigma > \frac{\Delta t}{2} \max \left[u^2, v^2, w^2\right] \tag{9}$$

When donor cell is used this restriction may be relaxed.

Finally, when low Reynolds number flows are simulated the time step may be restricted by the condition

$$\Delta t < \frac{1}{\nu} \frac{1}{2\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} + \frac{1}{\Delta z^2}\right)} \tag{10}$$

In analogy with the interpretation of equation (8) this restriction can be described as limiting the distance over which momentum diffuses during one time step to less than one cell width.

(This Page Intentionally Left Blank)

SECTION V

VERIFICATION OF THE NUMERICAL SOLUTION

Several qualitative checks were made in order to determine if the methodology incorporated into VARMINT was encoded correctly.

As a check on the variable zoning feature of the code as well as on the diffusion, advection and boundary condition sections, a laminar boundary layer building over a flat plate was calculated.

The grid used for this problem is shown in Figure 3. Initial conditions for the problem shown in Figure 4 is unit velocity over the plate. Figure 5 shows the flow after 150 cycles when the boundary layer has been established. An examination of the results of this problem is shown in Figure 6. The u velocity is plotted as a function of height for three locations at the front, middle and rear of the plate. Note the linear gradient in velocity at the plate and the increase in boundary layer thickness with increasing distance from the front of the plate. To the extent that comparison with the analytic solution is possible these results show good agreement.

The check of the internal obstacles was accomplished by calculating the flow over a flat plate with a notch. The first problem shown in Figures 7-9 is a notch formed by rectangular obstacles. The flow is from left to right with unit velocity over the plate and zero velocity in the notch. The flow at the left boundary is maintained at unity throughout the calculation. Figure 7 shows the computational grid with the plate and notch indicated. Figures 8 and 9 show the resultant velocity vectors at cycles 1 and 21 respectively. Note that by cycle 21 the rotational flow inside the notch has been established. Notice also that

the velocity vectors at the left are tilted upward, indicating a small upward component of velocity. This is the start of the boundary layer development.

Diagonal obstacles were also incorporated into the code and checked. Figures 10 and 11 show the results of the calculation of a flat plate with a V notch. The initial conditions were similar to the rectangular notch problem with unit velocity above the plate and zero velocity in the notch. Although too small to see in Figure 11, a rotational flow field has been established at the bottom of the notch.

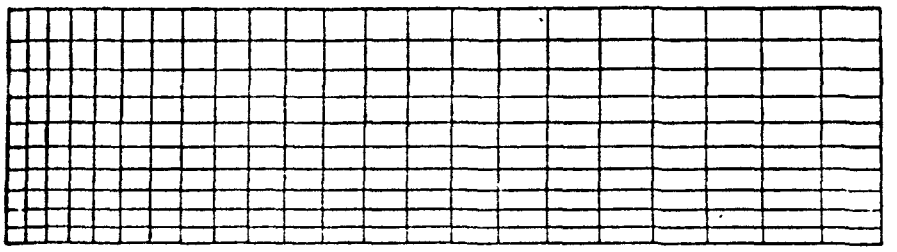


Figure 3. Grid used for laminar boundary layer calculation.

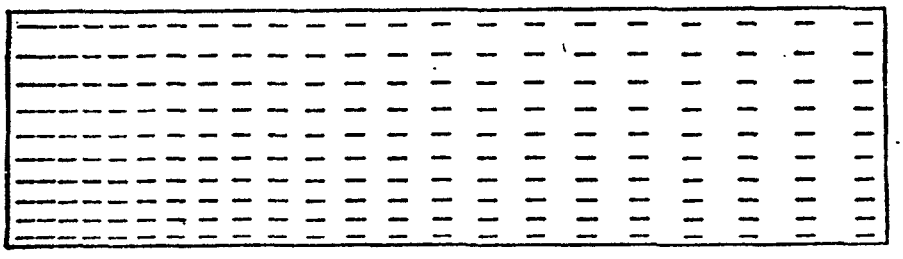


Figure 4. Initial velocity field for laminar boundary layer calculation.

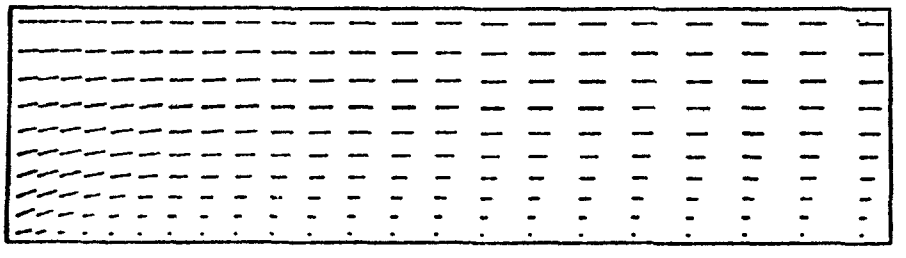


Figure 5. Velocity vectors after 150 computational cycles.

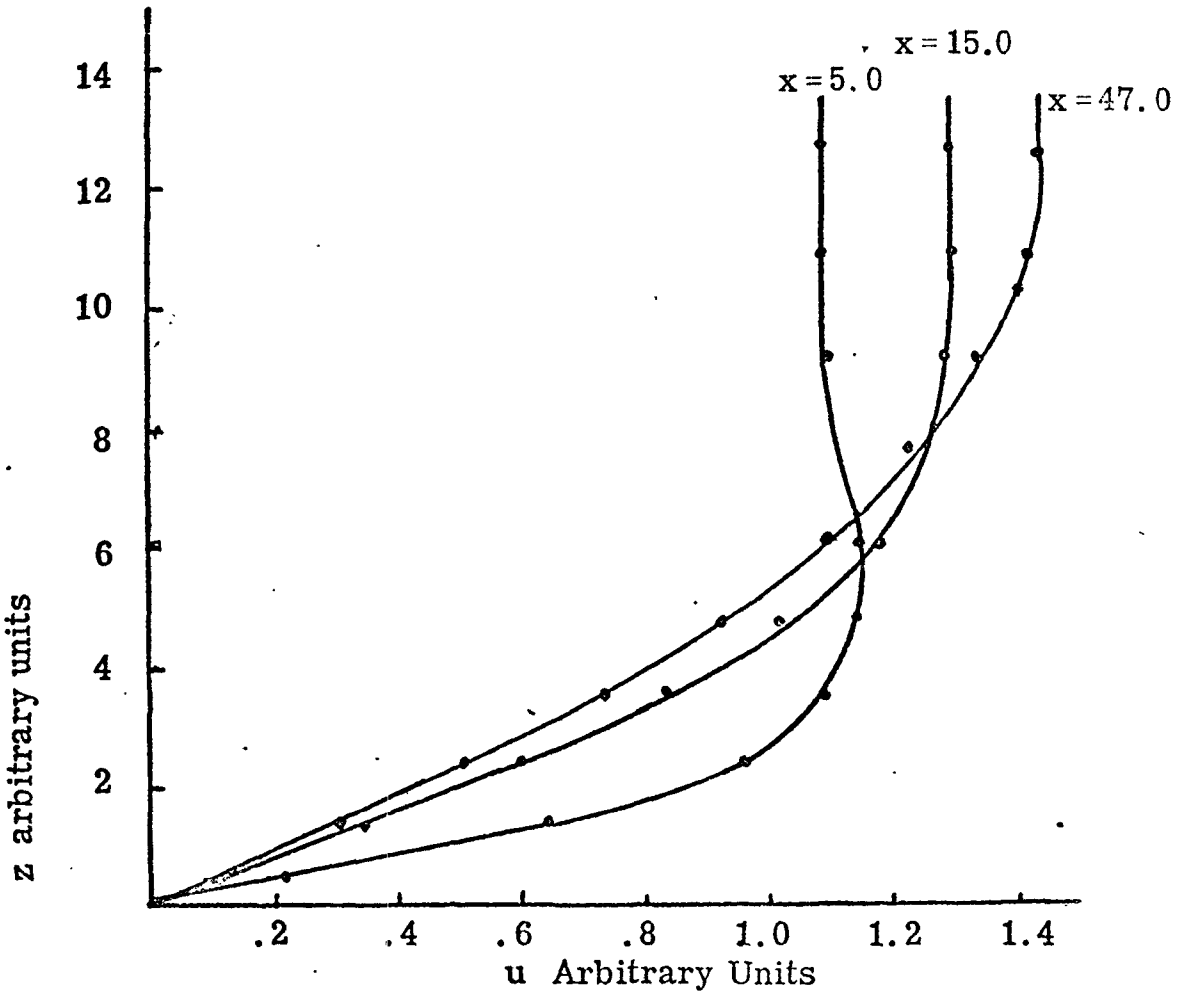


Figure 6. Horizontal velocity as a function of height for three locations along a flat plate.

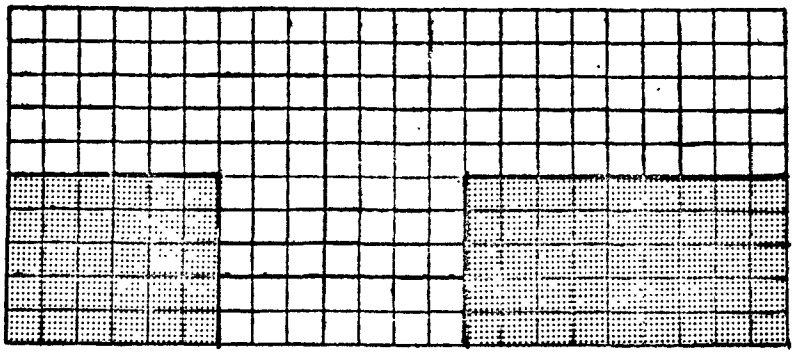


Figure 7. Computational grid for plate over notch calculation.

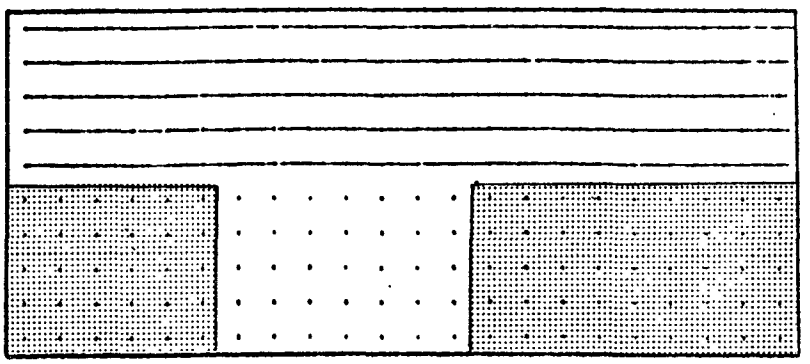


Figure 8. Initial velocity field for flow over notch calculation.

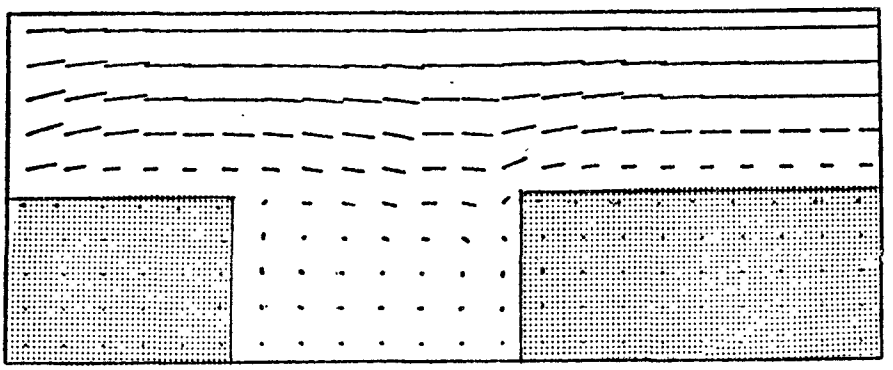


Figure 9. Velocity vectors after twenty-one cycles of computation.

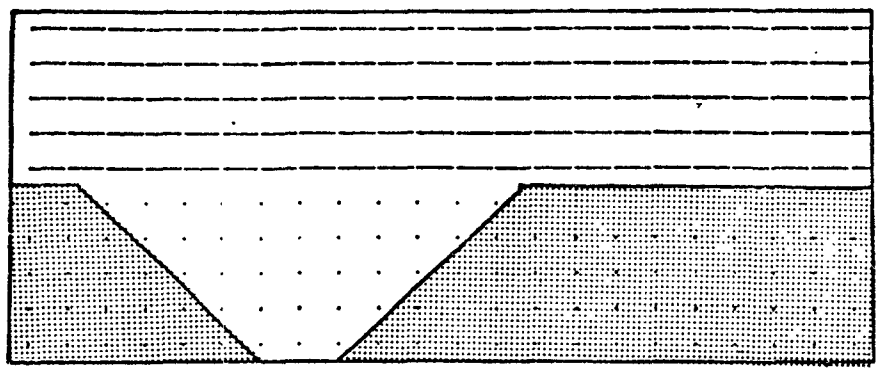


Figure 10. Initial velocity field for V notch problem.

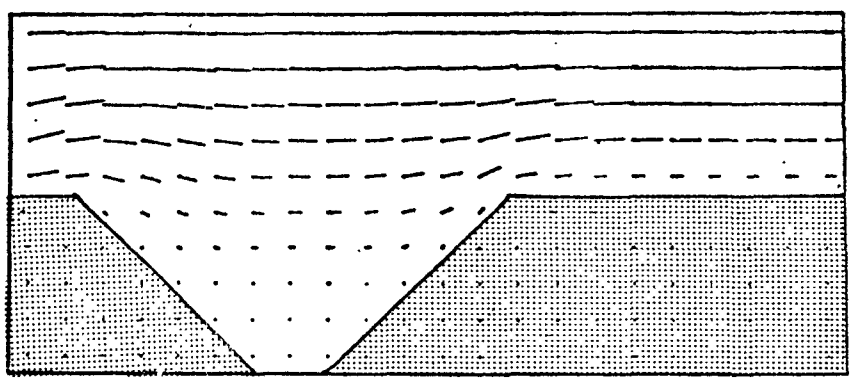


Figure 11. Velocity vectors after 16 computational cycles.

SECTION VI

SOLUTION OF THE DENSITY EQUATION USING PARTICLES

The equation describing turbulent atmospheric diffusion including pollutant sources and chemical reactions,

$$\frac{\partial c}{\partial t} = -\frac{\partial}{\partial x_i} u_i c + \frac{\partial}{\partial x_j} k_{jk} \frac{\partial c}{\partial x_k} + S + R \tag{11}$$

equates the time rate of change of concentration $\partial c/\partial t$, to the advective rate of change of $-(\partial/\partial x_i)\, u_i\, c$, due to the mean velocity field plus the gradient of the turbulent flux with the position dependent eddy diffusivity (k_{ij}), $(\partial/\partial x_j)\, k_{jk}\, (\partial c/\partial x_k)$, plus source of concentration, S, plus the change in concentration by chemical reactions, R.

The VARMINT code uses Lagrangian point particles to represent pollutant mass, so that Eulerian species concentration is obtained by summing the masses of the appropriate species in any volume of interest. Changes in particle position are calculated to simulate pollutant mass transport due to both advection and diffusion as specified by the turbulent atmospheric diffusion equation. In addition, changes in particle mass are calculated to simulate changes in pollutant concentration due to chemical reactions.

This technique was developed by Sklarew et al[4] and modified by Hotchkiss and Hirt.[5] In the limit of a large number of particles, equation (11) is solved without the numerical errors introduced by Eulerian finite difference techniques. To simulate advection and diffusion the particles are moved an increment dx_i in the ith direction where,

$$dx_i = u_i\, dt + dw_i \tag{12}$$

This increment is composed of two parts. The fluid velocity times the time step simulates the advection, while dw_i is a random displacement which simulates turbulent diffusion.

The random displacement dw_i is a variable whose average value is zero and whose variance is equal to $(2k_{ii}\,dt)^{\frac{1}{2}}$. This method developed by Hotchkiss and Hirt[5] approximates the process of eddy diffusion as a random walk scaled to the local eddy diffusivity. This technique cannot directly model the effect of the off diagonal terms in the eddy diffusion tensor k_{ij}, however if these terms are significant (i.e. comparable in magnitude to the diagonal terms) we note that a symmetric tensor can be brought into diagonal form by an orthogonal transformation. Therefore a rotation can always be defined such that the eddy diffusion tensor is diagonal. Thus the random movement of the particles can be done in the rotated frame.

Localized sources are modeled by emitting new particles of the appropriate mass and number densities each cycle. For a source strength Q(gm/sec), a time step of Δt (seconds) and a particle mass m, the number of particles added each cycle is $Q \cdot n/m$. The particle mass m must be chosen so that the total number of particles are large enough to accurately solve equation (11) yet minimize computer run time and storage requirements. For each localized source about 10 to 100 particles emitted per cycle seems to satisfy these criteria.

The concentration of a species is determined by summing the mass of the particles inside a prescribed fluid volume and then dividing by that volume. In the current version of VARMINT the fluid volumes chosen correspond to the cells used in the hydro calculation although this is not required. For computations involving a single species, volume weighting the mass of a particle in the neighborhood of a point where

the concentration is required can eliminate discontinuous jumps in the concentration as a mass particle enters and leaves a fluid volume, however this procedure leads to additional diffusion-like errors when chemistry involving multiple species is present.

Once the concentration of all the relevant species in a fluid element are known, the chemical reactions over a time period Δt can be solved and new concentrations computed. Any method of solution of the chemical rate equations is allowed. The VARMINT code treats the chemistry as a "black box" into which is entered Δt and the species concentrations, and from which the concentrations at time plus Δt is returned. Each particle mass is then adjusted so that

$$m^{n+1} = m^n \cdot \frac{c^{n+1}}{c^n}$$

where m^{n+1} is the particle mass at time t plus Δt, m^n is the particle mass at time t.

Multiple species can be handled in two ways. Each specie can be treated as individual particles or each particle can have multiple masses representing each specie associated with its location. Both methods seem to give the same accuracy while the second method saves both computer time and storage by reducing the total number of particles required.

Boundary conditions for the particles are handled in a manner consistent with the boundary conditions in the hydrodynamics section of VARMINT.

Rigid walls and obstacles are treated by reflecting the particles, continuitive boundaries are treated by allowing the particle to leave the computational mesh and periodic boundaries are treated by allowing the particle to leave the computation mesh and return at the other end of the mesh.

SECTION VII

VERIFICATION OF THE PARTICLE ROUTINE

The initial validation of the particle section of VARMINT was done by comparing the VARMINT calculation of a point source in a uniform wind field with the analytic solution of an inert Gaussian plume. The inert plume calculations was made to assess the advection, diffusion, concentration and boundary condition sections of the code. Initially, difficulties were experienced because without diffusion in the downwind direction all the particles emitted during a particular time step were constrained to have the same downwind coordinate, causing them to "line up." Emitting the particles throughout the time step instead of in a single puff corrected this difficulty. This effect is illustrated in Figure 12. The calculation illustrated on the top of Figure 12 is compared with the Gaussian formula in Figure 13. The concentrations are in general units of mass per length^3. The solid lines are the analytic results computed from the Gaussian formula. The plotted points represent the average value of the computed concentrations while the bars associated with each point represent the standard deviation of concentration values about this mean value. Both ground level concentration and concentrations at the level at which the plume was emitted are presented for both the down wind and the cross wind direction. Additionally the variation in concentration with height is also plotted. In all plots the mean values of the VARMINT calculation agrees with the Gaussian formula.

To test the method for adjusting particle masses due to changes by chemical reactions the simple rate equation $dC/dt = -\lambda C$ was chosen

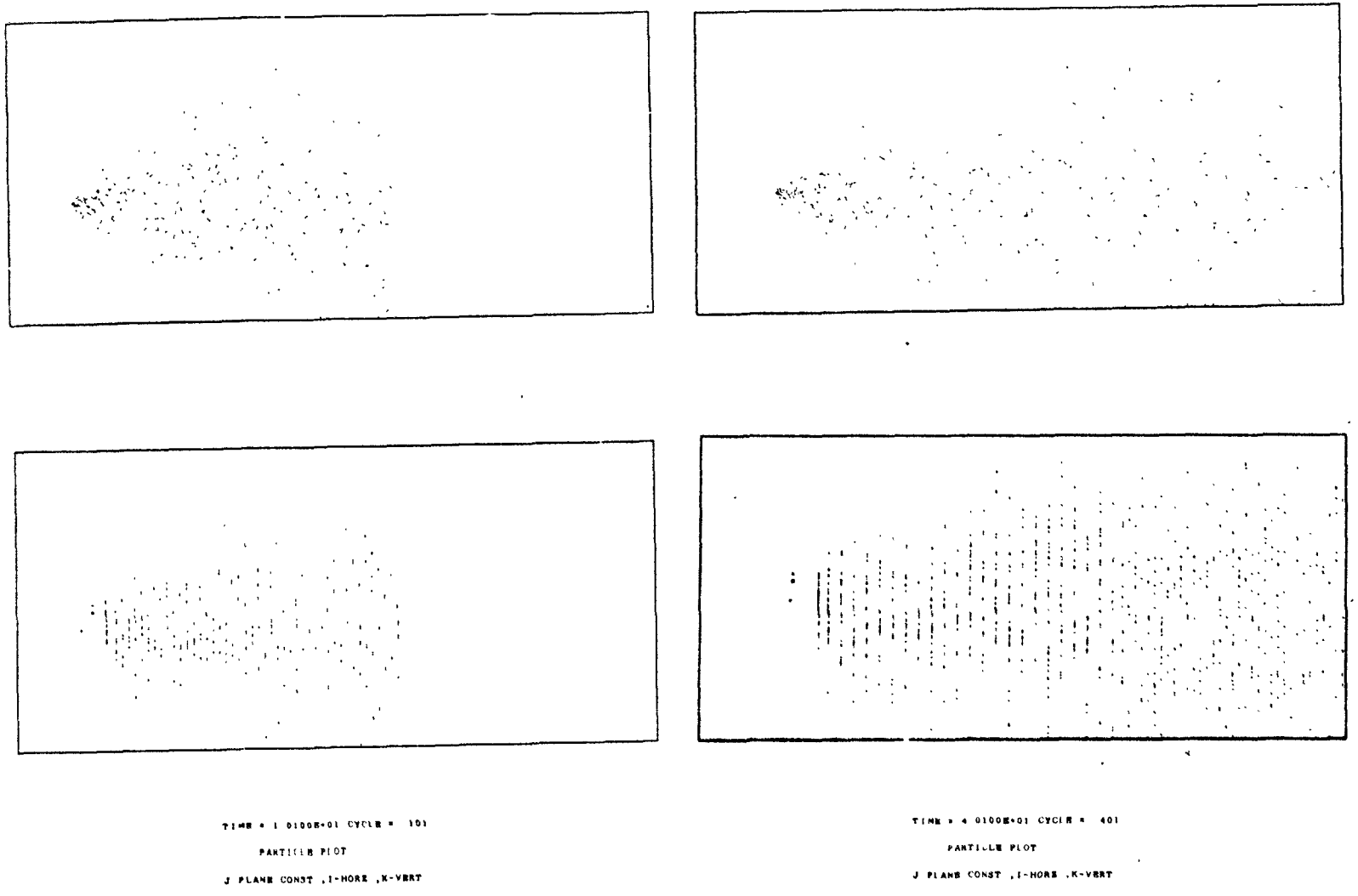


Figure 12. Particle Locations during a VARMINT Simulation of a Gaussian Plume with (Top Two Figures) and without (Bottom Two Figures) Subcycle Staggering.

$$C = \frac{Q}{2\pi\sigma_y\sigma_z\bar{u}} \exp\left(-y^2/2\sigma_y^2\right) \left\{\exp\left(-(z-H)^2/2\sigma_z^2\right) + \exp\left(-(z+H)^2/2\sigma_z^2\right)\right\}$$

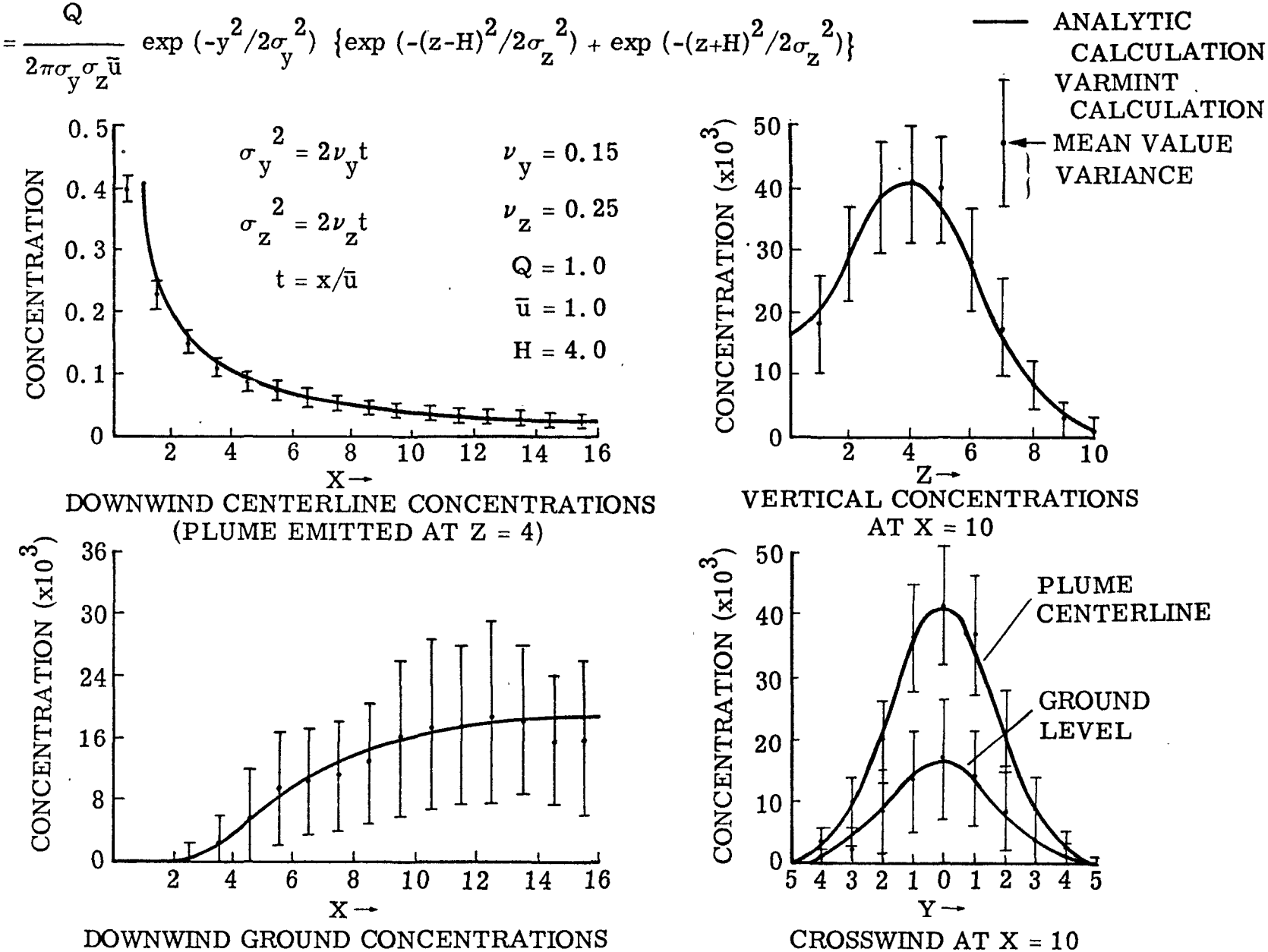


Figure 13. Comparison of VARMINT Calculations and Analytic Solution for a Point Source

to allow comparison with an analytic solution. The results of this test run is shown in Figure 14. Again the units of concentration are general. These two comparisons demonstrate that the methods incorporated into VARMINT can reproduce the analytical formulas and could simulate more realistic situations with wind shear and variable diffusivity, terrain and time varying source strengths.

As a further examination of the particle mass change logic and as a preliminary introduction to reactive plumes a chemistry model which solved the inorganic photochemical reactions

$$(h\nu) + NO_2 \rightarrow NO + O$$

$$O + (O_2) + (M) \rightarrow O_3 + (M)$$

$$O_3 + NO \rightarrow NO_2 + (O_2)$$

was incorporated into VARMINT and several problems were run. Since the Gaussian formulas cannot take into account nonlinear reactions, a simple code was written that solved the finite difference form of the diffusion equation in the $y-z$ plane while advancing along the x-axis in a Lagrangian sense. The method is illustrated in Figure 15.

A problem of interest is that of a nitrogen-dioxide source emitted into clear ambient air. This example was computed by both codes and compared in Figure 16. The initial source strength was 90 ppm NO_2, and most of the NO_2 remained intact with only a small amount becoming NO and O_3. The circles and dots represent the mean concentrations predicted by VARMINT, while the solid lines are the results calculated by the Eulerian-Lagrangian code. The bars through the dots and circles for the downwind centerline concentration are the variance of

$$C = \frac{Q}{2\pi\sigma_y\sigma_z\bar{u}} \exp(-y^2/2\sigma_y^2)\left\{\exp(-(z-H)^2/2\sigma_z^2) + \exp(-(z+H)^2/2\sigma_z^2)\right\}\exp(-\lambda t)$$

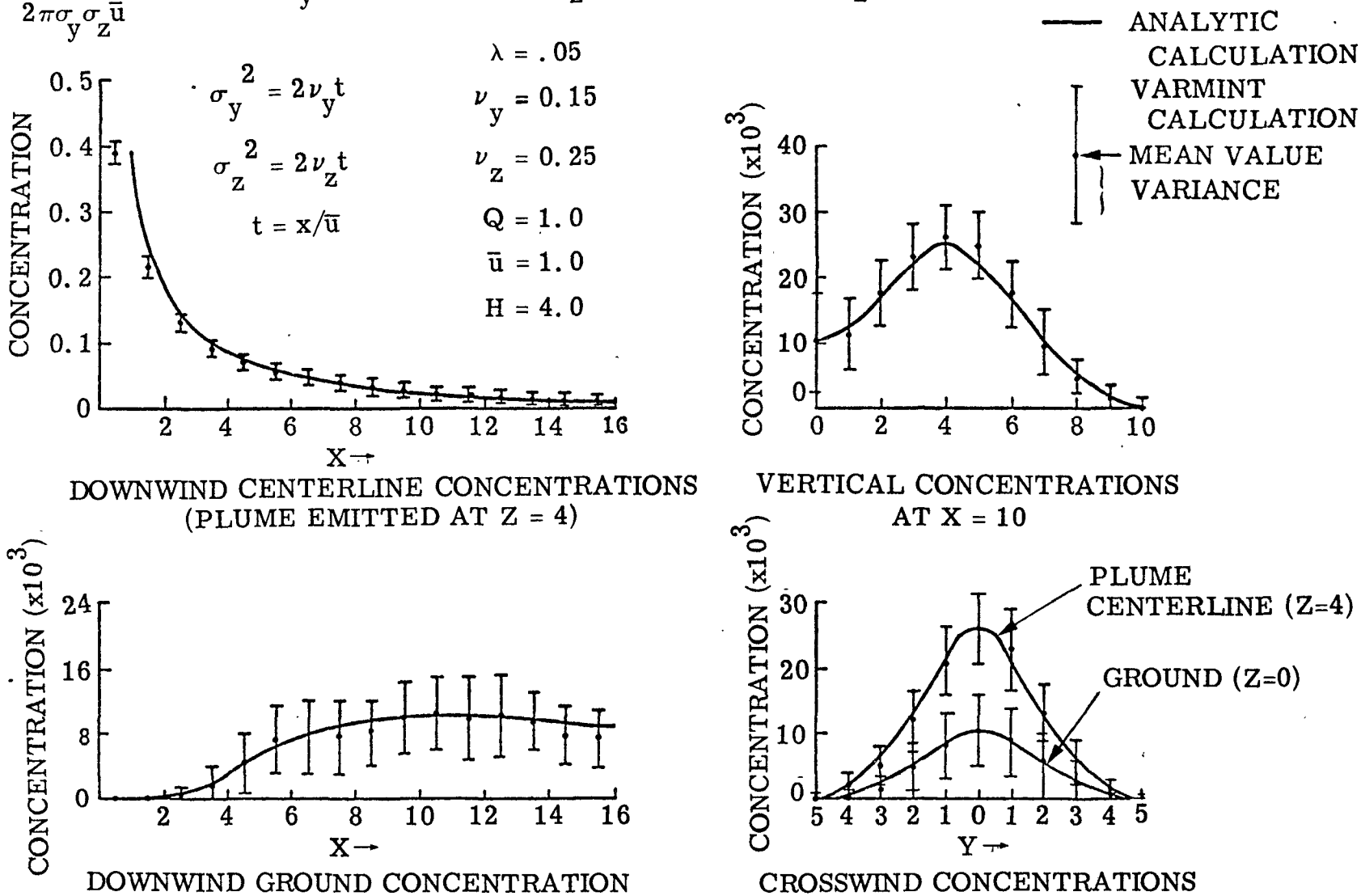


Figure 14. Comparison of VARMINT Calculation and Analytic Solution for a Point Source with Exponential Decay

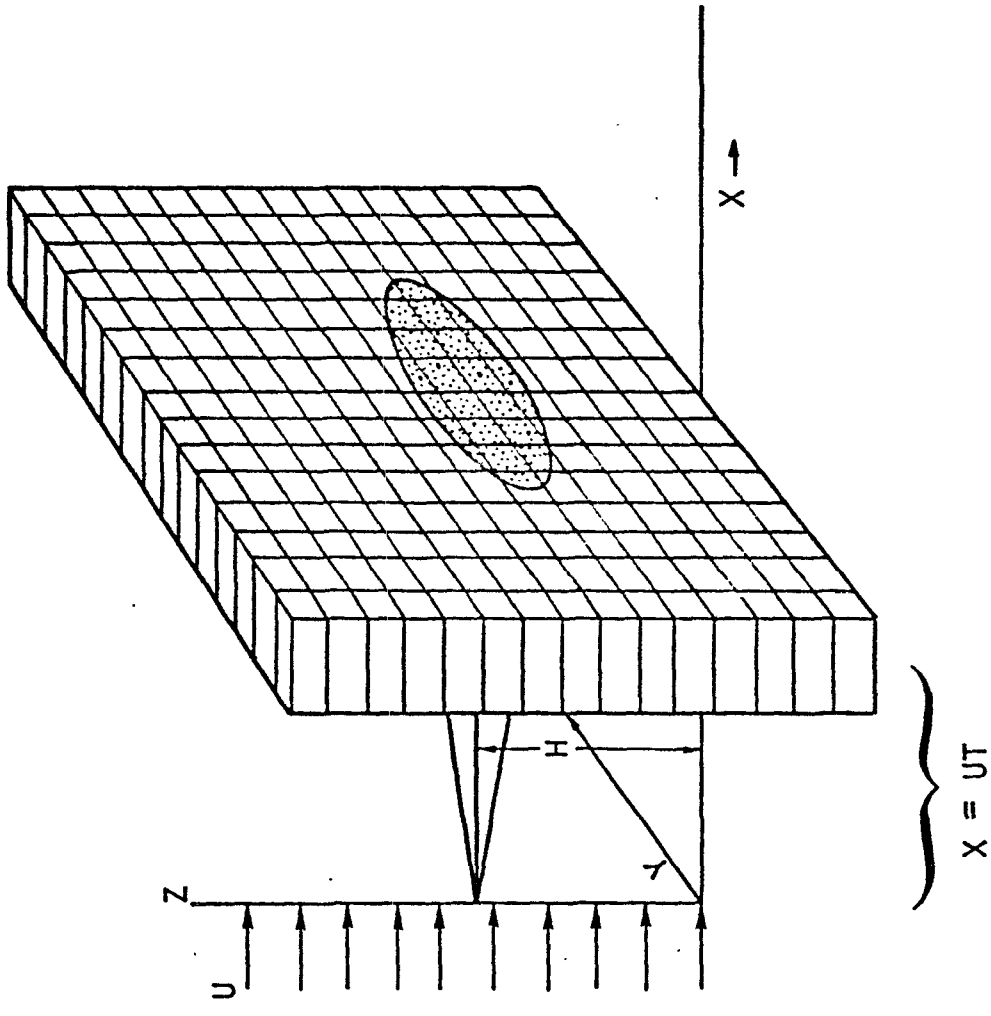


Figure 15. Schematic diagram for Eulerian-Lagrangian code.

the concentration predicted by VARMINT. The bars have been left off of the other graphs to make them more legible.

Another situation is shown in Figure 17. Here a source in chemical equilibrium (NO_2 = 73 ppm, NO = 1.7 ppm, O_3 = 1.7 ppm) is emitted into an ambient background that is also in equilibrium at a lower concentration (NO_2 = 0.2, NO = 0.035, O_3 = 0.24). At high pollution levels the chemical equilibrium shifts to NO_2 with little NO or O_3 while at low levels both O_3 and NO_2 are dominant. It must be remembered however that these reactions are simplifications of the real situation and represent a rough approximation to the actual polluted atmosphere. The incorporation of an ambient background concentration into VARMINT was accomplished by adding a fixed value representing background levels to the concentrations computed from the partial masses. As a consequence of this method, if the computed concentration drops below the background level the particles will develop negative mass. Thus, the outward diffusion of particles with negative mass would represent the inward diffusion of ambient pollutants. An example of this type of situation is shown in Figure 18. The Eulerian-Lagrangian code has simulated a source of nitrogen oxide (at 99 ppm) emitted into an ambient atmosphere containing $\frac{1}{2}$ ppm ozone. The conversion of nitrogen dioxide is inhibited at the plume center by the lack of ozone. If the NO source were spread and over a large area, almost all NO would be quickly converted into NO_2.

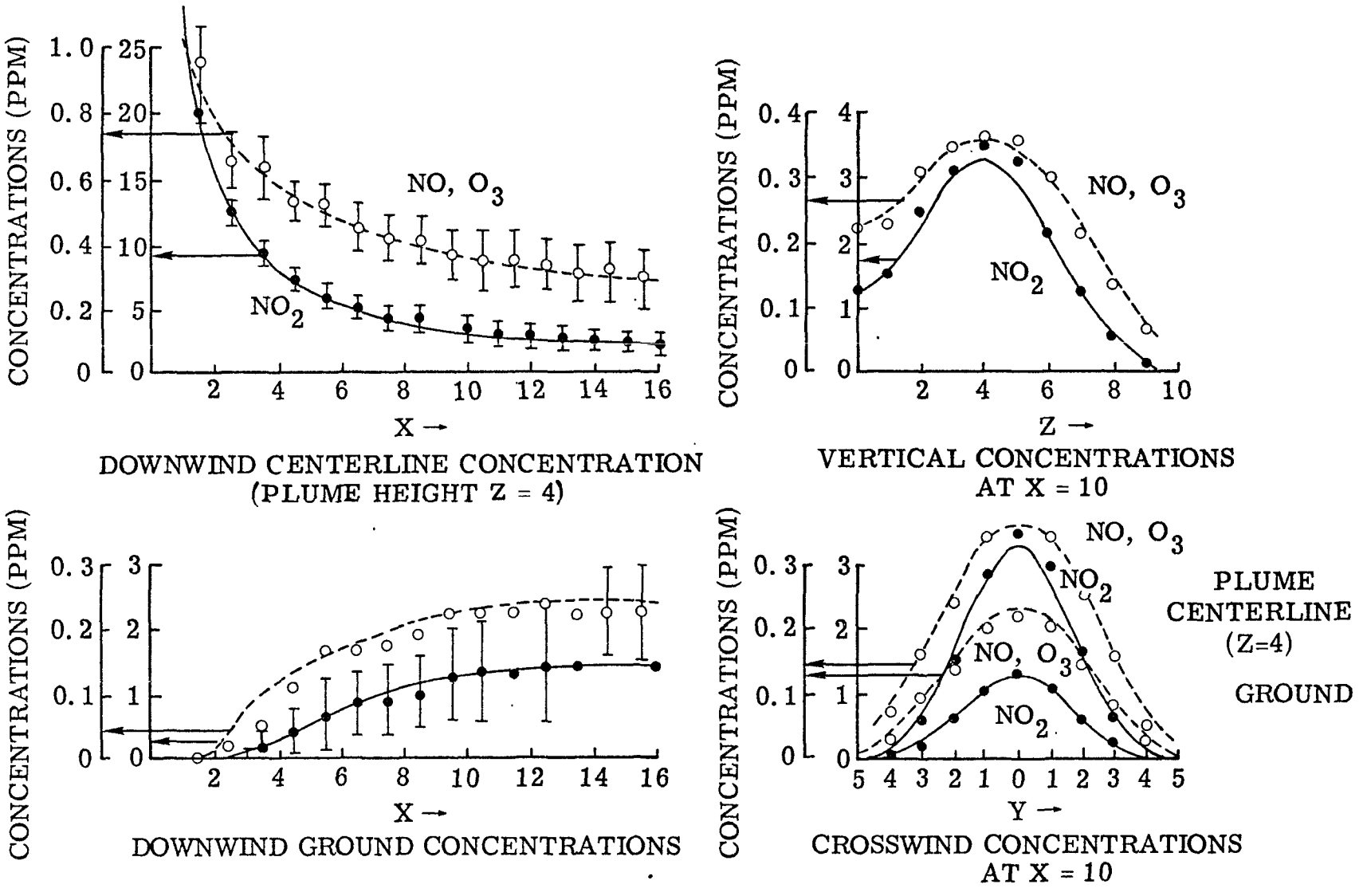


Figure 16. Nitrogen Dioxide Source Emitted into Clean Air

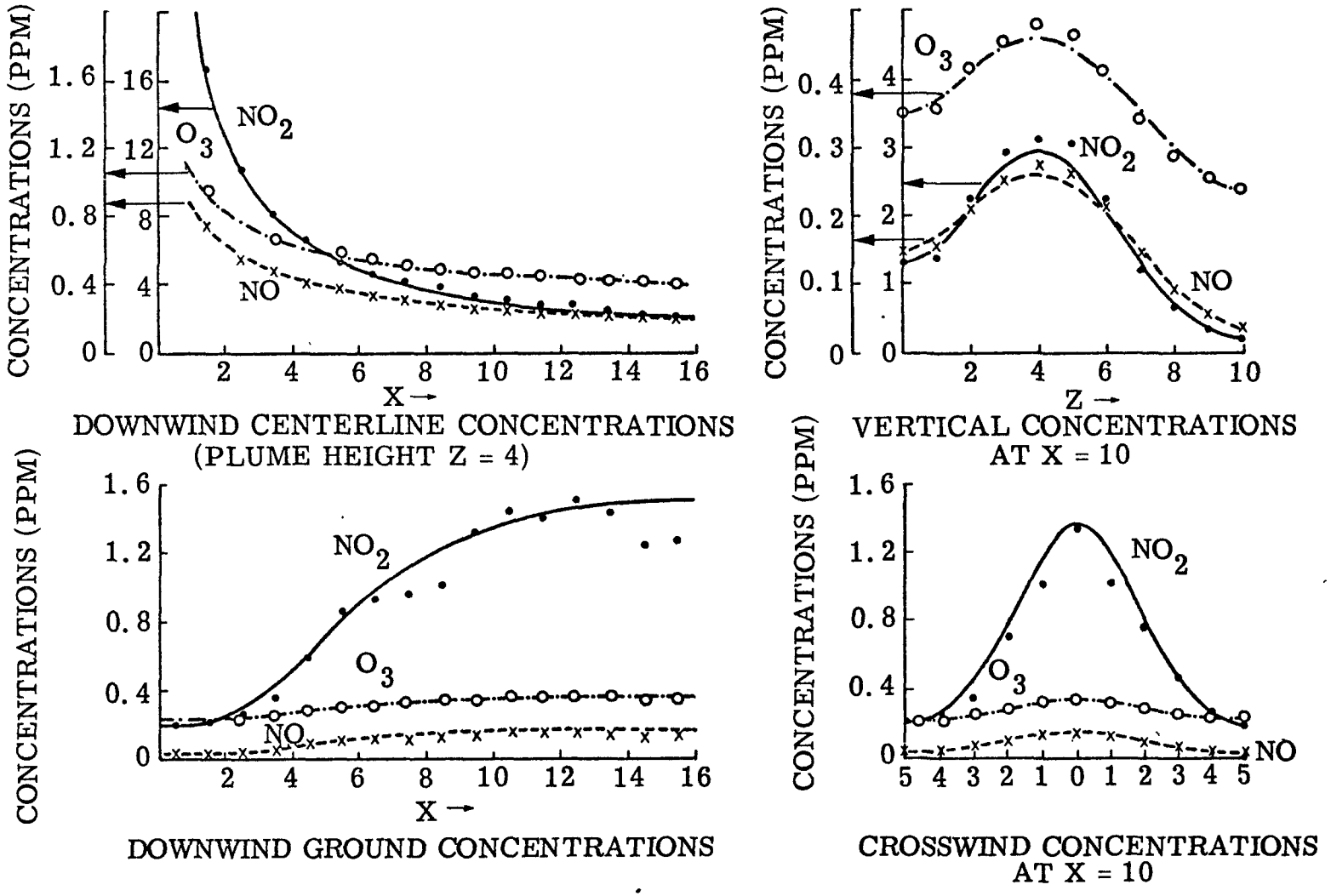


Figure 17. Equilibrium Source Emitted into Equilibrium Ambient Background

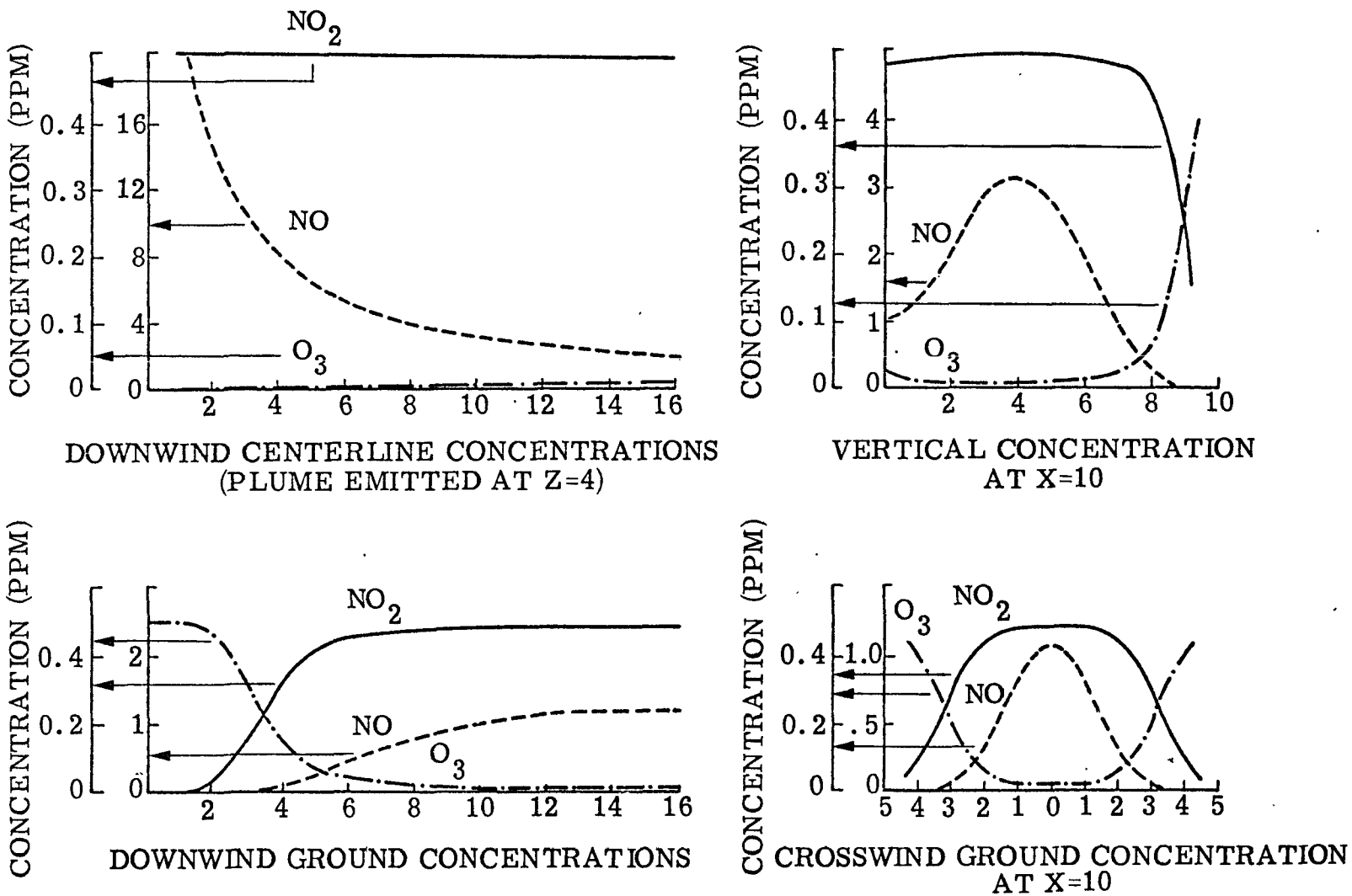


Figure 18. Nitrogen Oxide Source Emitted Into Ozone Background

SECTION VIII

REFERENCES

1. "Air Pollution in California – Annual Report 1970," Resources Agency – Air Resources Board, State of California (January 1969).

2. Hirt, C. W., and Cook, S. L., Calculating Three-Dimensional Flows Around Structures and Over Rough Terrain, Los Alamos Scientific Laboratory Report, LA-DC-13289 (November 1971).

3. Hirt, C. W. and Cook, S. L., Calculating Three-Dimensional Flows around Structures and over Rough Terrain, Journal of Computational Physics, Vol. 10, No. 2,(October 1972).

4. Sklarew, R. C., Fabrick, A. J. and Prager, S. E., A Particle-in-Cell Method for Numerical Solution of the Atmospheric Diffusion Equation and Applications to Air Pollution Problems, Systems, Science and Software Report 3SR-844, (November 1971).

5. Hotchkiss, R. S. and Hirt, C. W., Proceeding Summer Computer Simulation Conference, San Diego, California (June 1972).

(This Page Intentionally Left Blank)

SECTION IX

APPENDICES

		Page
A.	Users Guide for VARMINT	41
B.	VARMINT Listing	73

(This Page Intentionally Left Blank)

A. USERS GUIDE FOR THE VARMINT CODE

The VARMINT code has been written for and run on the CDC 7600, using FORTRAN IV. The main program (VRC) is the driver program. There are eight subroutines, seven of which are standard, with the eighth (KEM) being an offshoot of the particle movement section and not accessing information by means of COMMON storage. This chemistry subroutine will differ for types of species and kinds of reactions desired. The subroutines are:

VRS	-	setup subroutines for calculational arrays,
FILMCO	-	sets film constants,
FLMCAL	-	calls IGSB plotting routines,
VRM	-	main calculations,
VRPS	-	particle movement,
KEM	-	chemistry,
VRIO	-	output section for printing and plotting,
VRTAP	-	tape handling.

The logic flow for these is shown in Figure 1.

A job will normally operate with external hardware for storage as well as I/O. Besides the card reader and line printer, it can require large core memory (LARGE statement), microfiche (TAPE9 for data, TAPE10 for plotting), and magnetic tapes (TAPE7 for tape dump writes, TAPE8 for tape dump reads). Small core memory houses the common blocks:

VRCOM	contains the array A equivalenced to the flow field parameters, plotting parameters, and cell volumes;
CONCEN	contains chemical concentration related arrays;
AMIX	contains a temporary array;

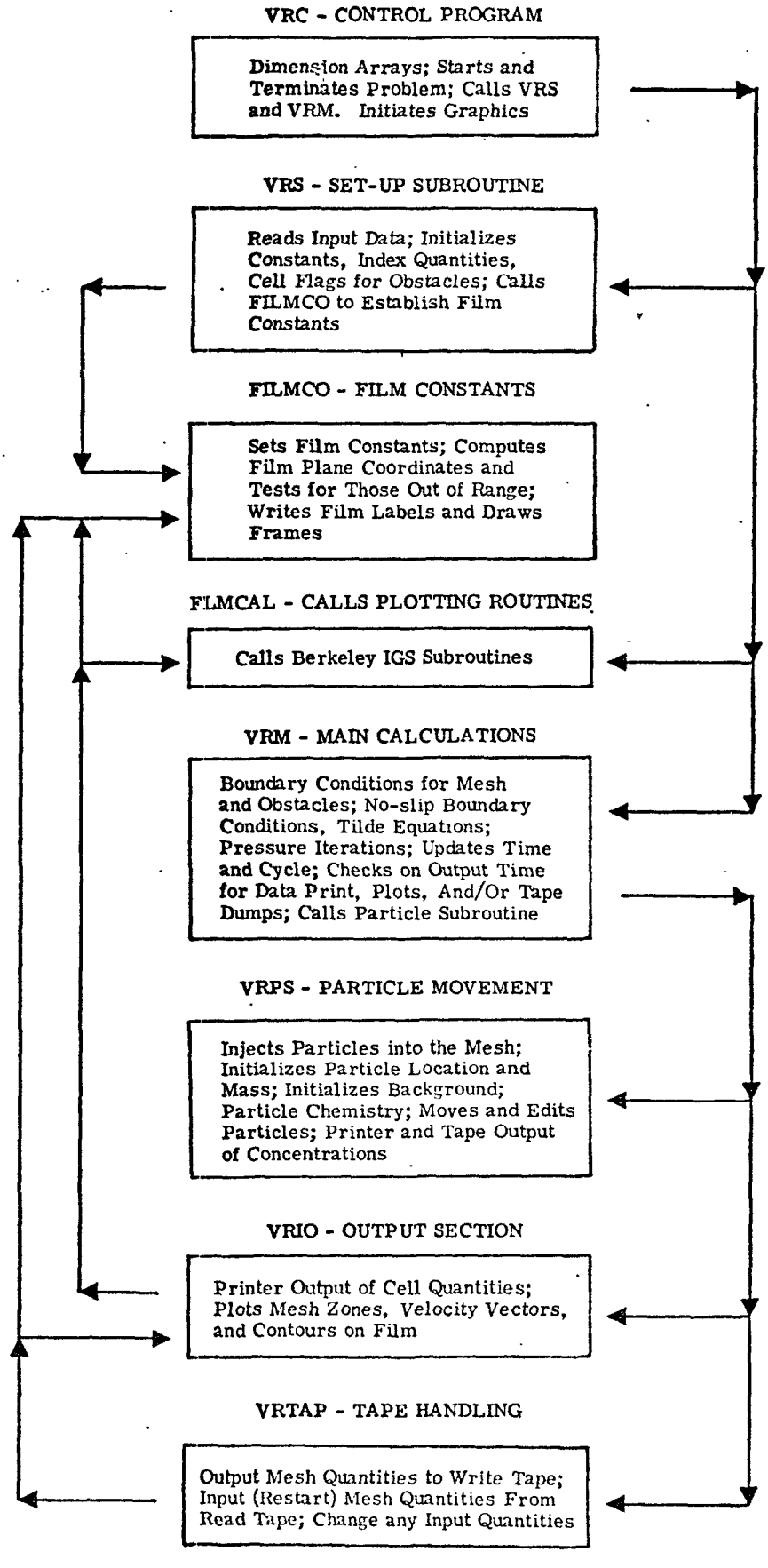


Figure 1. VARMINT flow chart.

FL	contains masking variables;
FLS	contains masking array;
VRCON	contains VARMINT constants;
MODSET	contains IGS film constants;
SAFLM	contains VARMINT generated film constants;
ZLAST	contains a dummy variable ZLAS which must always be the last variable in the last common block. The buffer length for tape dumps is calculated from the number of words between and including A(1) and ZLAS.

VARMINT INPUT DEFINITIONS

Rather than presetting all its constants within the code, VARMINT reads most of the non-calculated constants from cards. For each problem, the first two cards are read from VRC:

IBR	-	number of computational cells in x direction, if negative → stop, if 0 → restart;
JBR	-	number of computational cells in y direction;
KBR	-	number of computational cells in z direction, if IBR = 0 → KBR = tape dump number for restart;
PRFLM	-	film flag determines whether or not graphics are requested 0 → no film 1 → yes

LABEL: problem title.

VRS reads in the next cards:

DXC	-	size of cell in x direction
DYC	-	size of cell in y direction
DZC	-	size of cell in z direction
GX	-	body force in x direction
GY	-	body force in y direction
GZ	-	body force in z direction

ALX - donor or cell centered in x direction
ALY - donor or cell centered in y direction
ALZ - donor or cell centered in z direction
0 → cell centered
1 → donor

KWR - boundary condition flag for right wall
KWL - boundary condition flag for left wall
KWF - boundary condition flag for front wall
KWA - boundary condition flag for aft wall
KWT - boundary condition flag for top wall
KWB - boundary condition flag for bottom wall
1 → rigid
2 → continuative
3 → periodic
B0 - relaxation factor
CEP - convergence criterion for pressure iteration

DT - time step

$$\delta t \leq \frac{1}{4} \frac{\text{cell size}_{(\min)}}{\text{velocity}_{(\max)}}$$

TPLT - time increment between plots
TPRT - time increment between long prints
TFIN - time when to finish
TWTD - time when to dump on tape
NPRT - initial print options
0 → no first print
1 → print cycle 0
2 → print cycle 0, 1 and at time TPRT

NGOP - number of graphics options (5 are available)
NOVP - number of variables to be plotted by the contours option
LPR - long print option
1 → no long print
2 → long print on microfiche
3 → long print on paper
DROU - scaling factor for velocity vector plots

NWPC - number of words per cell equivalenced to A_{ijk}
NWPP - number of words per particle
- particle coordinates and mass
NTPE - particle type
1 → marker particles
2 → turbulent particles

GPX - body force on particles, x direction
GPY - body force on particles, y direction
GPZ - body force on particles, z direction
NPMX - maximum particles allowed
UI - initial u velocity
VI - initial v velocity
WI - initial w velocity

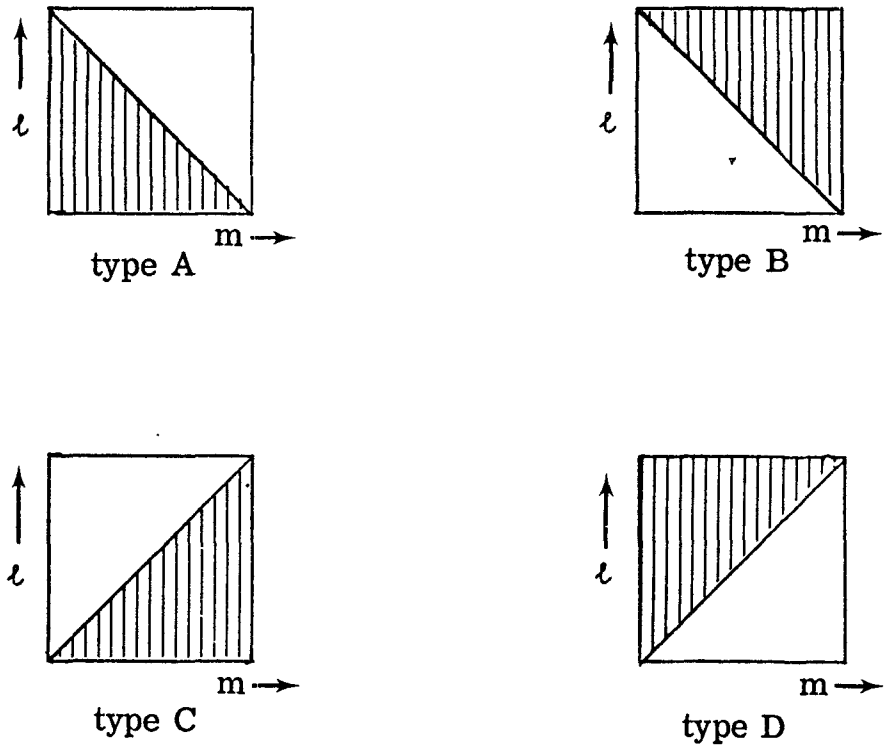
NU - diffusion constant
ISFLG - slip flag for boundary conditions
0 → free slip
1 → no slip

NRC - number of obstacle blocks

IL - left-most boundary cell of block
IR - right-most boundary cell of block
JA - aft-most boundary cell of block
JF - front-most boundary cell of block
KB - bottom-most boundary cell of block
KT - top-most boundary cell of block
OB - type of obstacle
0 → cube
1-12 → diagonal blocks (see Figure 2)

IF NGOP≠0,
$IGOP_i$ - graphics options
1 → particles
2 → zones
3 → velocity vectors
4 → contours
5 → finish (this must be an included option)

For Cube: code = 0



For Diagonals:

code	1	2	3	4	5	6	7	8	9	10	11	12
type	A	C	A	C	D	B	D	B	A	D	C	B
ℓ-axis	K	K	K	K	K	K	K	K	J	J	J	J
m-axis	I	I	J	J	I	I	J	J	I	I	I	I

I: horizontal
J: fore-aft
K: vertical

Figure 2. Obstacle axis code.

IF NOVP$\neq$0,

$VNTP_i$	-	contour options 1 → pressure 2 → u velocity 3 → w velocity 4 → v velocity
$NCVTP_i$	-	number of contours per contour plot
IPMM	-	number of cuts in I for J × K plane (vectors and contours)
IPM_m	-	locations of cuts (i.e., I cell no.)
JPPM	-	number of cuts in J for J × K plane
JPM_m	-	locations of cuts (i.e., J cell no.)
KPMM	-	number of cuts in K for I × J plane
KPM_m	-	location of cuts (i.e., K cell no.)
IPPP	-	particle plot projection for I plane
JPPP	-	particle plot projection for J plane
KPPP	-	particle plot projection for K plane 0 → not this plane 1 → this plane
IPROJ	-	perspective view plotting flag (not used at present so = 0) 0 → no perspective 1 → use perspective

The particle subroutine reads in the rest:

IPL	- left	(if IPL = 0, skip particle calculation)
IPR	- right	side of particle source block cell number
JPA	- aft	side of particle source block cell number
JPF	- front	side of particle source block cell number
KPT	- top	side of particle source block cell number
KPB	- bottom	side of particle source block cell number
XPCEN	- x coordinate for particle point source location	
YPCEN	- y coordinate for particle point source location	
ZPCEN	- z coordinate for particle point source location (if XPCEN = 0, particle will be placed at random inside source block above)	
NOPI	- number of particles to input	
PDX	- size of cell in x direction for particle mesh	(not used at present
PDY	- size of cell in y direction for particle mesh	(not used at present
PDZ	- size of cell in z direction for particle mesh	(not used at present
TWIP	- time when to input particles	
UPI	- particle velocity in x direction	
VPI	- particle velocity in y direction	
WPI	- particle velocity in z direction	
SIGMA	- diffusivity	

VARMINT is constructed for 3-dimensional simulation. For efficiency however, all arrays are singly dimensioned variables. For each time cycle, progression through the mesh is ordered by columns (K), next by rows (I), and finally by planes (J). The number of words per cell is essential for correct machine locations of the flow field variables operated on in each cell, for all positioning is done with indexing counters.

VRC is the control program of VARMINT. In VRC the problem date and time are called and printed before the input and output files are declared. The main array dimensions are read in along with the film option flag for plotting. The error flag is initialized; and if the film plot option is on, the graphics are initiated.

An array dimension determines the next logic sequence. If $\bar{I}$ is negative, the problem terminates; if $\bar{I}$ equals zero, restart procedures are instigated; if $\bar{I}$ is greater than zero, the value is assumed to be the size of the real mesh in the $\bar{I}$, or horizontal direction, and the problem is initialized.

When the problem is started by this last method, the problem label is the second data card read in. VRC then calls VRS to set up the problem. After VRS returns control, VRC calls VRM, which is the main calculational subroutine. If the problem were to be restarted, using the second method described above, VRC would call VRM immediately, bypassing VRS because the values therein would have been stored on magnetic tape which was generated during a previous run.

Following the calls to VRM and VRS the program checks the status of the error flag. If this had been tripped within one of the subroutines, graphics is concluded (when the plot flag is on), and the run is terminated with a printed diagnostic. Otherwise, we return to the beginning of the program and anticipate another problem. Hence, to end this run, the next data card must contain the negative $\bar{I}$. This negative $\bar{I}$ sends us to the section which concludes the graphics. We then exit, and the run terminates.

VRS sets up a specific problem for us. It reads the following from data cards:

- The cell size constants for three dimensions,
- The prescribed body accelerations for 3 dimensions,
- The constants determining amount by which flux is calculated by donor or cell centered method for 3 dimensions,
- The boundary condition flags for the 6 walls:
  1 → rigid
  2 → continuative
  3 → periodic,
- The relaxation factor for pressure iteration,
- The convergence criterion for pressure iterations,
- The time criteria,
- The diffusion constant,
- The obstacle no-slip flag, and
- The print options.

Then these input variables are printed.

VRS initializes constants and calculates cell index quantities. It checks for programming errors such as array dimensions exceeded. Whenever errors are found, the error flag is tripped and control returns to the calling program.

Next, VRS reads the obstacle limit cards and defines the obstacles in the mesh. Obstacle surface flags are set for no-slip boundary conditions. It reads and writes both graph options and contour plot values which are needed by VRIO. Graphic quantities are read in subsequently. Fluid flow arrays are initialized to zero. Velocities and cell sizes are initialized to prescribe input values. This section allows for variable zoning. Mesh coordinates are initialized. Cell volumes are calculated with allowances with respect to the obstacles. Plotting constants are set next with a call to FILMCO. Following this call, control is returned to VRC.

VRM is the main calculational subroutine. This is where the flow field is established and from which I/O and particle input are controlled.

First, VRM initializes PMAX for the pressure iteration and defines loop constants to be used in cell prints from VRIO. This section is never returned to during the course of this problem.

The calculational sequence for every time step begins at statement number 1000. Machine time for the cycle is noted and the pressure iteration counter is set to zero. A time dependent flag determines whether or not the hydro will be skipped; this flag allows passage on initial cycle. If the flag is tripped, we bypass the wind field calculations and decide whether or not to call the particle movement subroutine VRPS. This subroutine is only called when the maximum number of particles allowed in the system is greater than zero.

When the hydro is not to be bypassed, $\bar{I}$ must next be interpreted. If $\bar{I}$ equals zero, we jump to the final section of VRM where VRTAP, the tape I/O subroutine, is called. Hence, we are restarting a previously run problem. We read the dump tape until we arrive at the time T from which we wish to proceed. The information from the tape dump supplies all the parameters normally initialized in the set up subroutine VRS as well as the state of the flow field at time T. This data has been stored in the labeled common blocks VRCOM, CONCEN, AMIX, FL, FLS, VRCON, MODSET, SAFLM and ZLAST and are accessed by all subroutines and VRC. We return from VRTAP to statement number 1000 and continue with $\bar{I}$ now containing the integer number of computational cells in the x(or i) direction.

When $\overline{I}$ is greater than zero, it is assumed that we also have the number of computational cells in the y direction ($\overline{J}$) and the z direction ($\overline{K}$). First, the boundary conditions are set for the right and left walls, top and bottom walls, and then the fore and aft walls of the mesh. For each of the above, assigned GO TO statements route the user to the rigid, continuative, or periodic boundary conditions. No slip boundary conditions can also be calculated for these when the proper flag is set. Next, the free slip obstacle boundary conditions are set, conditions depending on the obstacle type. Following this, the no-slip boundary conditions for obstacles are calculated, but only if the flag is set and the pressure iteration is not in progress.

Once boundary conditions have been set, we are free to calculate the tilde velocities. First, face No. 1 (j = 1 plane) and the left column and bottom row of velocity components for computational cells are transferred to storage arrays. Cell quantity indices are calculated along with diffusion. The u-, v-, and w-tilde fluxes for each cell are then calculated. Where

$$\tilde{u}_{i+\frac{1}{2}jk} = u_{i+\frac{1}{2}jk} + \frac{2\Delta t}{(\Delta x_i + \Delta x_{i+1})}(P_{ijk} - P_{i+1jk}) + \Delta t g_x$$

$$+ \frac{\Delta t}{(\Delta x_i + \Delta x_{i+1})}[u_{i+1jk}(u_{i+\frac{1}{2}jk} + u_{i+\frac{3}{2}jk}) + \alpha_x |u_{i+1jk}|(u_{i+\frac{1}{2}jk} - u_{i+\frac{3}{2}jk})$$

$$- u_{ijk}(u_{i-\frac{1}{2}jk} + u_{i+\frac{1}{2}jk}) - \alpha_x |u_{ijk}|(u_{i-\frac{1}{2}jk} - u_{i+\frac{1}{2}jk})]$$

$$+ \frac{\Delta t}{2\Delta y_j}[v_{i+\frac{1}{2}j+\frac{1}{2}k}(u_{i+\frac{1}{2}jk} + u_{i+\frac{1}{2}j+1k}) + \alpha_y |v_{i+\frac{1}{2}j+\frac{1}{2}k}|(u_{i+\frac{1}{2}jk} - u_{i+\frac{1}{2}j+1k})$$

$$- v_{i+\frac{1}{2}j-\frac{1}{2}k}(u_{i+\frac{1}{2}j-1k} + u_{i+\frac{1}{2}jk}) - \alpha_y |v_{i+\frac{1}{2}j-\frac{1}{2}k}|(u_{i+\frac{1}{2}j-1k} - u_{i+\frac{1}{2}jk})]$$

$$+ \frac{\Delta t}{2\Delta z_k}[w_{i+\frac{1}{2}jk+\frac{1}{2}}(u_{i+\frac{1}{2}jk} + u_{i+\frac{1}{2}jk+1}) - \alpha_z |w_{i+\frac{1}{2}jk-\frac{1}{2}}(u_{i+\frac{1}{2}jk} + u_{i+\frac{1}{2}jk+1})$$

$$- w_{i+\frac{1}{2}jk-\frac{1}{2}}(u_{i+\frac{1}{2}jk-1} + u_{i+\frac{1}{2}jk}) - \alpha_z |w_{i+\frac{1}{2}jk-\frac{1}{2}}|(u_{i+\frac{1}{2}jk-1} - u_{i+\frac{1}{2}jk})]$$

$$+ \frac{2}{(\Delta x_i + \Delta x_{i+1})}[(u_{i+\frac{3}{2}jk} - u_{i+\frac{1}{2}jk})\sigma_{i+1jk}/\Delta x_{i+1} - (u_{i+\frac{1}{2}jk} - u_{i-\frac{1}{2}jk})\sigma_{ijk}/\Delta_{xi}]$$

$$+ \frac{2}{(\Delta y_j + \Delta y_{j+1})}[(u_{i+\frac{1}{2}j+1k} - u_{i+\frac{1}{2}jk})\sigma_{ij+1k}/\Delta y_{j+1} - (u_{i+\frac{1}{2}jk} - u_{i+\frac{1}{2}j-1k})\sigma_{ijk}/\Delta y_j]$$

$$+ \frac{2}{(\Delta z_k + \Delta z_{k+1})}[(u_{i+\frac{1}{2}jk+1} - u_{i+\frac{1}{2}jk})\sigma_{ijk+1}/\Delta z_{k+1} - (u_{i+\frac{1}{2}jk} - u_{i+\frac{1}{2}jk-1})\sigma_{ijk}/\Delta z_k]$$

$$\tilde{v}_{ij+\frac{1}{2}k} = v_{ij+\frac{1}{2}k} + \frac{2\Delta t}{(\Delta y_j+\Delta y_{j+1})}(P_{ijk}-P_{ij+1k})+\Delta t g_y$$

$$+ \frac{\Delta t}{2\Delta x_i}\left[u_{i+\frac{1}{2}j+\frac{1}{2}k}(v_{i+1j+\frac{1}{2}k}+v_{ij+\frac{1}{2}k}) + \alpha_x|u_{i+\frac{1}{2}j+\frac{1}{2}k}|(v_{i+1j+\frac{1}{2}k}-v_{ij+\frac{1}{2}k})\right.$$

$$\left.- u_{i-\frac{1}{2}j+\frac{1}{2}k}(v_{i-1j+\frac{1}{2}k}+v_{ij+\frac{1}{2}k}) - \alpha_x|u_{i-\frac{1}{2}j+\frac{1}{2}k}|(v_{i-1j+\frac{1}{2}k}-v_{ij+\frac{1}{2}k})\right]$$

$$+ \frac{\Delta t}{(\Delta y_j+\Delta y_{j+1})}\left[v_{ij+1k}(v_{ij+\frac{1}{2}k}+v_{ij+\frac{3}{2}k}) + \alpha_y|v_{ij+1k}|(v_{ij+\frac{1}{2}k}-v_{ij+\frac{3}{2}k})\right.$$

$$\left.- v_{ijk}(v_{ij-\frac{1}{2}k}+v_{ij+\frac{1}{2}k}) - \alpha_y|v_{ijk}|(v_{ij-\frac{1}{2}k}-v_{ij+\frac{1}{2}k})\right]$$

$$+ \frac{\Delta t}{2\Delta z_k}\left[w_{ij+\frac{1}{2}k+\frac{1}{2}}(v_{ij+\frac{1}{2}k}+v_{ij+\frac{1}{2}k+1}) + \alpha_z|w_{ij+\frac{1}{2}k+\frac{1}{2}}|(v_{ij+\frac{1}{2}k}-v_{ij+\frac{1}{2}k+1})\right.$$

$$\left.-w_{ij+\frac{1}{2}k-\frac{1}{2}}(v_{ij+\frac{1}{2}k-1}+v_{ij+\frac{1}{2}k}) - \alpha_z|w_{ij+\frac{1}{2}k-\frac{1}{2}}|(v_{ij+\frac{1}{2}k-1}+v_{ij+\frac{1}{2}k})\right]$$

$$+ \frac{2}{(\Delta x_i+\Delta x_{i+1})}\left[(v_{i+1j+\frac{1}{2}k}-v_{ij+\frac{1}{2}k})\sigma_{i+1jk}/\Delta x_{i+1}-(v_{ij+\frac{1}{2}k}-v_{i-1j+\frac{1}{2}k})\sigma_{ijk}/\Delta x_i\right]$$

$$+ \frac{2}{(\Delta y_j+\Delta y_{j+1})}\left[(v_{ij+\frac{3}{2}k}-v_{ij+\frac{1}{2}k})\sigma_{ij+1k}/\Delta y_{j+1}-(v_{ij+\frac{1}{2}k}-v_{ij-\frac{1}{2}k})\sigma_{ijk}/\Delta y_j\right]$$

$$+ \frac{2}{(\Delta z_k+\Delta z_{k+1})}\left[(v_{ij+\frac{1}{2}k+1}-v_{ij+\frac{1}{2}k})\sigma_{ijk+1}/\Delta z_{k+1}-(v_{ij+\frac{1}{2}k}-v_{ij+\frac{1}{2}k-1})\sigma_{ijk}/\Delta z_k\right]$$

$$\tilde{w}_{ij+\frac{1}{2}} = w_{ijk+\frac{1}{2}} + \frac{2\Delta t}{(\Delta z_k + \Delta z_{k+1})} (P_{ijk} - P_{ijk+1}) + \Delta t g_z$$

$$+ \frac{\Delta t}{2\Delta x_i} [u_{i+\frac{1}{2}j+\frac{1}{2}k} (w_{ij+\frac{1}{2}k} + w_{i+1j+\frac{1}{2}k})$$

$$+ \alpha_x |u_{i+\frac{1}{2}j+\frac{1}{2}k}| (w_{ij+\frac{1}{2}k} - w_{i+1j+\frac{1}{2}k}) - u_{i-\frac{1}{2}j+\frac{1}{2}k} (w_{i-1j+\frac{1}{2}k} + w_{ij+\frac{1}{2}k})$$

$$- \alpha_x |u_{i-\frac{1}{2}j+\frac{1}{2}k}| (w_{i-1j+\frac{1}{2}k} - w_{ij+\frac{1}{2}k})]$$

$$+ \frac{\Delta t}{2\Delta y_j} [v_{ij+\frac{1}{2}k+\frac{1}{2}} (w_{ijk+\frac{1}{2}} + w_{ij+1k+\frac{1}{2}}) + \alpha_y |v_{ij+\frac{1}{2}k+\frac{1}{2}}| (w_{ijk+\frac{1}{2}} - w_{ij+1k+\frac{1}{2}})$$

$$- v_{ij-\frac{1}{2}k+\frac{1}{2}} (w_{ij-1k+\frac{1}{2}} + w_{ijk+\frac{1}{2}}) - \alpha_y |v_{ij-\frac{1}{2}k+\frac{1}{2}}| (w_{ij-1k+\frac{1}{2}} - w_{ijk+\frac{1}{2}})]$$

$$+ \frac{\Delta t}{(\Delta z_k + \Delta z_{k+1})} [w_{ijk+1} (w_{ijk+\frac{1}{2}} + w_{ijk+\frac{3}{2}}) + \alpha_z |w_{ijk+1}| (w_{ijk+\frac{1}{2}} - w_{ijk+\frac{3}{2}})$$

$$- w_{ijk} (w_{ijk-\frac{1}{2}} + w_{ijk+\frac{1}{2}}) - \alpha_z |w_{ijk}| (w_{ijk-\frac{1}{2}} - w_{ijk+\frac{1}{2}})]$$

$$+ \frac{2}{(\Delta x_i + \Delta x_{i+1})} [(w_{i+1jk+\frac{1}{2}} - w_{ijk+\frac{1}{2}}) \sigma_{i+1jk} / \Delta x_{i+1} - (w_{ijk+\frac{1}{2}} - w_{i-1jk+\frac{1}{2}}) \sigma_{ijk} / \Delta x_i]$$

$$+ \frac{2}{(\Delta y_j + \Delta y_{j+1})} [(w_{ij+1k+\frac{1}{2}} - w_{ijk+\frac{1}{2}}) \sigma_{ij+1k} / \Delta y_{j+1} - (w_{ijk+\frac{1}{2}} - w_{ij-1k+\frac{1}{2}}) \sigma_{ijk} / \Delta y_j]$$

$$+ \frac{2}{(\Delta z_k + \Delta z_{k+1})} [(w_{ijk+\frac{3}{2}} - w_{ijk+\frac{1}{2}}) \sigma_{ijk+1} / \Delta z_{k+1} - (w_{ijk+\frac{1}{2}} - w_{ijk-\frac{1}{2}}) \sigma_{ijk} / \Delta z_k]$$

These equations combine donor cell and cell-centered difference schemes; $\alpha = 1$ giving full donor cell, $\alpha = 0$ yielding centered differencing, a fraction in between promoting a combination thereof. Quantities needed at positions other than when they are defined are calculated as simple averages. The donor cell method is suggested for most cases because of the greater stability in the calculations without an additional viscous term. The diffusivity constant, σ, is an input quantity read from data cards in VRS or from the dump tape in VRTAP.

The tilde velocities must now be modified to insure conservation of mass. This is done in the pressure iteration section after boundary conditions have again been set. The velocity divergence D is calculated, and then the pressure is changed by an amount

$$\delta P = -\beta D$$

where

$$\frac{1}{\beta} = \frac{1}{\beta_o} \frac{\delta D_{ijk}}{\delta P_{ijk}} = \frac{1}{\beta_o} \left[\frac{1}{\Delta x_i} \frac{2\Delta t}{(\Delta x_i + \Delta x_{i+1})} + \frac{1}{\Delta x_i} \frac{2\Delta t}{(\Delta x_i + \Delta x_{i-1})} \right.$$

$$+ \frac{1}{\Delta y_j} \frac{2\Delta t}{(\Delta y_j + \Delta y_{j+1})} + \frac{1}{y_j} \frac{2\Delta t}{(\Delta y_j + \Delta y_{j-1})} + \frac{1}{\Delta z_k} \frac{2\Delta t}{(\Delta z_k + \Delta z_{k+1})}$$

$$\left. + \frac{1}{\Delta z_k} \frac{2\Delta t}{(\Delta z_k + \Delta z_{k-1})} \right] ,$$

and β_0 is the relaxation factor. The velocities are then adjusted by adding the δP at the cell faces to the corresponding velocity component. This process is repeated successively in all cells until no cell has a magnitude of D greater than ϵ , where ϵ is a function of the magnitude of the velocities calculated.

When the pressures converge within 1000 iterations, the time counter is updated by dt and VRM calls for cell prints in VRIO. These will not necessarily be printed every cycle but rather at a predetermined time interval TPRT.

The boundary conditions are reset. If particles are to be allowed in the field, the particle movement subroutine VRPS is called. At appropriate time intervals, the particles will move within the wind field. This calling section is the first section of operation for the time cycle when the hydro is bypassed. When the hydro has been skipped, the timer is not updated, so the time is updated here under these circumstances.

The subroutine FILMCO contains entry points under which various film constants are computed or tested. Also, the film frame is appropriately labeled. A call to FILMCO checks that film option is on, sets mesh length reciprocals, sets film coordinate starting points, and initializes do loop constants. Entry at FPLAN computes film plane coordinates and constants for the $I \times K$, $I \times J$ or $J \times K$ plane. Entry at FCTEST performs a test to determine whether film coordinates are out of range and resets those which are. Entry at FLMID labels SAI, date, time and cycle and draws the frame.

The subroutine FLMCAL contains the calls to the Berkeley SC 4060 graphic routines. Entry FLMADV records the display and advances the film. Entry FLMFIN terminates the graphic output. Entry FLMINIT initializes the graphics package. Entry FLMLAB labels the film frame. Entry FLMLIN draws noncontiguous line segments. Entry FLMPNT plots points. Entry FLMOPT computes graphics options such as redefining the plot point, character size, line width and density, and point size.

The particle subroutine VRPS is responsible for the injection of particles into the mesh; for initializing particle location and mass, for initializing background, for particle chemistry (through calls to KEM), for moving and editing the particles, and for printer and tape output of particle information.

VRPS is called by the main calculations subroutine VRM when the maximum number of particles allowed in the mesh (NPMX, an input parameter) is greater than zero. The first pass through VRPS invokes the read statements which initialize particle input locations and specify point or area assignments, time when to input particles, particle velocities in x, y, and z directions, the number of particles to input per pass, and the diffusivity. Background and concentration constants are set, as well as the probability curves used in the turbulent dispersion section. If the time to input particles is greater than the time of the initial call to VRPS, control is returned to VRM.

At the appropriate time interval, particles are emitted by one of two methods: assignments to a point or an area. When the input variable specifying the x-coordinate of the point location (XPCEN) is less than or equal to zero, the area source is calculated from a given left, aft, and bottom local boundary and a fraction of the corresponding upper bounds, the fraction being determined by a random number generator. When XPCEN is greater than zero, a point source location is designated using the coordinates given in the input variables (XPCEN, YPCEN, ZPCEN). The particle locations for both cases are stored in the dimensional variables XPC, YPC, and ZPC. Up

to ten species, or chemicals, are allotted to each particle location. Each has an initial mass which is assigned to the variable PM, PM increasing by the number of species. The number of particles emitted each time is the same for each species and has been assigned by the constant NOPI.

The first time that particles are emitted, masses and locations are assigned, and then control is returned to VRM. Thereafter, entries to VRPS bypass the initializing sequence and preassign the logic flow for the marker particles and the turbulent dispersion equations. The concentration array is zeroed out, if that option is on, and then begins the major loop for the calculation of particle transport.

The location of each particle composite (or block), a composite containing N species, is arrived at by taking the integer part of the location variable. This tells us which cell the block is in. To find out how far into the cell the block has progressed, we use the decimal part of the same variable. Using the velocity components of the flow field for this and adjacent cells, we interpolate the u, v, and w fluid velocities for simple marker particle advection in the following manner:

```
define  IX1 = IX + sgn (XW - .5)*1.0  ,  XW1 =  XW - .5
        IY1 = IY + sgn (YL - .5)*1.0  ,  YL1 =  YL - .5
        IZ1 = IZ + sgn (ZH - .5)*1.0  ,  ZH1 =  ZH - .5
```

where IX, IY, and IZ are the cell numbers in the x, y, and z directions, respectively, and XW, YL, and ZH are the corresponding fractions describing the distance the particle has moved within the cell.

I1, J1, and K1 are the resulting cell numbers described by IXI, IYI, and IZI, respectively.

The flux velocity components of the particle are:

$$
\begin{aligned}
U_P = XW \Big\{ & (1-YL1)\,[(ZH1)(U_{ijk1}) + (1-ZH1)(U_{ijk})] \\
& + (YL1)\,[(ZH1)(U_{ij|k1}) + (1-ZH1)(U_{ij1k})] \Big\} \\
& + (1-XW) \Big\{ (1-YL1)\,[(ZH1)(U_{i-1jk1} + (1-ZH1)(U_{i-1jk})] \\
& + (YL1)\,[(Zh1)(U_{i-1j|k1}) + (1-ZH1)(U_{i-1j|k})] \Big\} \quad ; \quad .
\end{aligned}
$$

$$
\begin{aligned}
W_P = ZH \Big\{ & (1-XW1)\,[YL1)(W_{ij|k}) + (1-YL1)(W_{ijk})] \\
& + (XW1)\,[(YL1)(W_{i|j|k}) + (1-YL1)(W_{i|jk})] \Big\} \\
& + (1-ZH) \Big\{ (1-XW1)\,[(YL1)(W_{ij|k-1}) + (1-YL1)(W_{ijk-1})] \\
& + (XW1)\,[(YL1)(W_{i|j|k-1}) + (1-YL1)(W_{i|jk-1})] \Big\} \quad ;
\end{aligned}
$$

$$
\begin{aligned}
V_P = YL \Big\{ & (1-ZH1)\,[(XW1)(V_{i|jk}) + (1-XW1)(V_{ijk}) \\
& + (XH1)\,[(XW1)(V_{i|jk1}) + (1-XW1)(V_{ijk1})] \Big\} \\
& + (1-YL) \Big\{ (1-ZH1)\,[(XW1)(V_{i|j-1k}) + (1-XW1)(V_{ij-1k})] \\
& + (ZH1)\,[(XW1)(V_{i|j-1k1}) + (1-XW1)(V_{ij-1k1})] \Big\} \quad .
\end{aligned}
$$

Second order particle advection can be calculated in like manner.

Another option available is turbulent dispersion. The displacement is added to the fluid velocity component and is defined as $\delta X = \sqrt{4\sigma\, dt}\ W \operatorname{sgn}(1-2\xi)$ where ξ is chosen randomly and satisfies the conditions

$$\xi = \frac{2}{\sqrt{\pi}} \int_0^W e^{-z^2}\, dz \quad , \quad 0 \le \xi \le 1 \quad ;$$

σ is the local diffusion coefficient;

dt is the time step;

$$W = W_k + \left(\frac{W_{k+1} - W_k}{R_{k+1} - R_k} \right) (\xi - R_k)$$ such that 20k-values are

are found for $W_k = \frac{W_{max}}{\#k}\, k$, k=1 , ... , #k and W_{max} is an upper limit such that particles will seldom shift more than one cell. R_k are solutions for

$$R_k = \frac{2}{\sqrt{\pi}} \int_0^{W_k} e^{-z^2}\, dz \quad , \quad R_k \le \xi \le R_{k+1} \quad .$$

The particle is then moved and given its new location as

$$XW_{new} = XW_{old} + \frac{U_p\, dt}{\Delta X} \quad ,$$

$$YW_{new} = YW_{old} + \frac{V_p\, dt}{\Delta Y} \quad ,$$

$$ZW_{new} = ZW_{old} + \frac{W_p\, dt}{\Delta Z}$$

after a particle has been reflected off an obstacle or the ground if its new displacement would have put it there. When a new displacement places the particle outside of the mesh, its new location is not retained. It is considered as outside the area of interest.

New concentrations must now be calculated for each cell. The particle masses for each species are summed for the cell into which the particle block has entered. This process continues until all particles have been moved.

We then loop through all species and all mesh cells to calculate the molecular concentrations for each cell from particle mass, cell volume, molecular weight, ppm conversion factor, weight of air, and background. At the appropriate time interval, these concentrations are printed or written on tape.

Next, the cells alone are looped over and the chemistry subroutine KEM is called to prescribe the reactions of the N species and deliver a new concentration for each. Then, each particle is re-weighted as a function of the ratio of old concentration to new concentration.

If the time is less than the next time when to input particles, then the time, cycle number, particle masses and locations, and the number of particles in the system are printed and control is returned to VRM. Otherwise, more particles, determined by the constant NOPI, are added to the system at the predetermined source location and mass is assigned to each one. Then the printing described above occurs and control is returned to VRM.

The current subroutine KEM solves the inorganic loop of the photochemical air pollution reactions

$$
\begin{aligned}
&1.\quad (h\nu) + NO_2 \overset{k_1}{\rightarrow} NO + O \\
&2.\quad O + (O_2) + (M) \overset{k_2}{\rightarrow} O_3 + (M) \qquad (1) \\
&3.\quad O_3 + NO \overset{k_3}{\rightarrow} NO_2 + (O_2)
\end{aligned}
$$

where $h\nu$ represents the ultraviolet photons of the solar flux, the third body, M, is the total concentration of all species present and the k's are the rate coefficients. The species in parens are assumed to remain constant and their values are incorporated into the rate constants. All concentrations are in parts per million (ppm). The values for k are estimated by Eschenneder to be

$$k_1 = 0.4 \text{ min}^{-1}$$

$$k_2 = 2.64 \times 10^6 \text{ min}^{-1}$$

$$k_3 = 30\text{-}40 \text{ ppm}^{-1} \text{ min}^{-1}$$

This set of reactions can be solved analytically as discussed below.

Noting that the value of k_2 is much greater than either k_1 or k_3, we can assume that the second reaction is in pseudo-equilibrium when compared to the other two reactions. Further, at equilibrium, we see that the ratio of O_3 to O is k_2 or that atomic oxygen, O, is about one millionth of the concentration of ozone, O_3.

Assuming that all oxygen is either O_2 or O_3 we can simplify Eq. (1) to

$$
\begin{aligned}
(O_2) + NO_2 &\overset{k_1}{\rightarrow} NO + O_3 \\
NO + O_3 &\overset{k_3}{\rightarrow} NO_2 + (O_2)
\end{aligned}
\tag{2}
$$

We further note that the total of all nitrogen, NO_X, and the total of all oxidents, TOX, remain constant, i.e.,

$$
\begin{aligned}
TOX &= NO_2 + O_3 = \text{constant} \\
NO_X &= NO_2 + NO = \text{constant}
\end{aligned}
\tag{3}
$$

This means that NO and O_3 can be expressed as functions of NO_2 and the constants TOX and NO_X

$$
\begin{aligned}
NO &= NO_X - NO_2 \\
O_3 &= TOX - NO_2
\end{aligned}
\tag{4}
$$

We can, therefore, write the chemical rate equations in terms of only one species NO_2.

$$
\frac{dNO_2}{dt} = - k_1 NO_2 + k_3 (NO)(O_3)
$$

becomes

$$\frac{dNO_2}{dt} = - k_1 NO_2 + k_3 (NO_X - NO_2)(TOX - NO_2) \quad . \tag{5}$$

Defining $y \equiv NO_2$ we can express Eq. (5) in the standard quadratic form

$$\frac{dy}{dt} = ay^2 + by + c \tag{6}$$

where

$$a = k_3$$

$$b = - k_1 - k_3 (TOX + NO_X)$$

$$c = k_3 \cdot NO_X \cdot TOX \quad .$$

If the rate of change of y (NO_2) is zero, the chemical reactions are in equilibrium and we can solve for the equilibrium value of y .

$$y_e = \frac{- b \pm \sqrt{b^2 - 4ac}}{2a} \quad .$$

Expanding the coefficients we find that the physically correct solution is

$$y_e = \frac{- b - \sqrt{b^2 - 4ac}}{2a} \quad . \tag{7}$$

If the time rate of change of y is not equal to zero we must solve the integral equation

$$\int_0^t dt = \int_{y_o}^{y} \frac{dy}{ay^2 + by + c} \quad . \tag{8}$$

The solution for the integral

$$\int \frac{dy}{ay^2 + by + c}$$

is given as

$$\begin{array}{ll} -\frac{2}{R}\tanh^{-1}\left(\frac{2ay+b}{R}\right) & b^2 > 4ac \\ & (2ay+b)^2 < b^2 - 4ac \\ -\frac{2}{R}\,\mathrm{ctnh}^{-1}\left(\frac{2ay+b}{R}\right) & b^2 > 4ac \\ & (2ay+b)^2 > b^2 - 4ac \\ -\frac{2}{2ay+b} & b^2 = 4ac \end{array} \qquad (9)$$

where

$$R = (b^2 - 4ac)^{\frac{1}{2}} \quad .$$

Solving equation (8) we get

$$t = -\frac{2}{R} f^{-1}\left(\frac{2ay+b}{R}\right) + C \quad . \qquad (10)$$

for $R \neq 0$, where f is either tanh or ctnh depending on the value of $(2ay + b)^2 - b^2 - 4ac$, and C is a constant of integration.

We can solve for C by noting at $t = 0$, $y = y_o$ so

$$C = \frac{2}{R} f^{-1}\left(\frac{2ay_o + b}{R}\right) \quad . \qquad (11)$$

Solving Eq. (10) for y gives

$$y = (-b + R\cdot f(-(t-C)R/2))/2a \qquad (12)$$

Note as $t \to \infty$, $f \to 1$, and $y \to y_e$ as $t \to 0$ $y = -b\,R\cdot f(cR/2)/2a$ and by Eq. (11) $y = y_o$.

Subroutine VRIO handles the printing and graphics output. The first section prints cell quantities and the second handles graphic output. Each can be executed without entering the domain of the other.

The first section encountered is the line printer section. After the subroutine is entered, if long print is not desired, if the hydro has been skipped, or if printing is not desired at this time of entry, then the printing section is skipped. Otherwise, VRIO sets up printer counters and flags and writes the time, cycle number, the number of iterations needed to converge during the pressure iteration, and the time increment dt. At cycle zero, the cell coordinates and cell sizes are displayed. Then and thereafter, the cell numbers and their respective pressures and velocity components are printed.

The graphic section is entered whether or not long print is required. Only the input flag for plotting being off can return control to the calling routine at this intermediate step. Otherwise, control is returned at the completion of the requested plots.

The graphic section can be separated into several parts, each part dealing with one of the available graphic options. After the options are initialized, the option counter is incremented, sending VRIO to either particle, mesh zone, velocity, vector, and/or contour plotting.

The first part plots particle coordinates, initializes constants necessary in plotting the particles, and then sets constants for either the I × K, the I × J, or the J × K plane. It is possible to obtain a view from each of these planes, but each view must be a separate plot. The mesh zone plot follows the same general set up for plane constants as the particle plot. It then draws line segments to display the mesh

from any of the three planes. Zones, however, will only be plotted once; the first time VRIO is called for plotting purposes. The velocity vectors and the contours are viewed from a specific plane, or cut, within the mesh. This is unique in that, while particles and zones are viewed from only the front, top, or right of the mesh, looking across all of the cells, the cut limits the view to a 2-D case. The user designates his I × K, I × J, or J × K viewing plane and will receive plots of only the quantity within that plane.

The velocity vector plot follows the same pattern of setting up constants for vectors and film plane coordinates as the earlier parts. It uses the velocity vector components appropriate for the particular view plane in computing the coordinates between which it will draw a line segment proportional to the magnitude and in the direction of the velocity, neglecting the velocity component normal to the plane. It then plots the vector.

The contouring plot again initializes general constants to plot the contours on film. The number of contour lines is optional. It then sets loop constants for the I×K, I×J, or J × K planes and encodes the title and labels the film. It transfers the desired variable to a contour storage array and computes the maximum and minimum values of this variable. The contour loop flags each contour cell, sets coordinates for the desired plane, and checks the contour variable value in each cell with respect to neighboring cells. When contour demarcation values are within this neighborhood, the vertex position might need a shift. If so, it calculates volume constants, contour cell volume, and the shifted vertex position. It then plots the contour on film.

After each plot, VRIO can go to various exits. It will either set up for a new plane and plot the same quantity from a different view point, or it will return to the beginning of the plotting section to increment the plotting counter and, by means of a computed GO TO, go to another plotting part. The last plotting option must always be included (now equal to 5), because this directs logic flow to return to the calling routine.

VRTAP is called for magnetic tape manipulations. Upon entry, $\overline{I}$ is interogated. If it is not zero, VRTAP proceeds to write mag tape records of the buffer length, the common blocks starting from VRCOM down to ZLAST, and the particle array stored in large core memory. It prints the tape dump number, the program time, and the cycle number for the user's reference.

When $\overline{I}$ equals zero, the subroutine assumes a restart request. It rewinds the dump tape and reads records until it arrives at the tape dump number requested by the initial $\overline{K}$. It then prints the time, cycle, and tape dump number. After this point, the users may insert any desired changes to common stored information, such as time when to finish. It then calls FILMCO and VRIO before returning to the calling routine.

(This Page Intentionally Left Blank)

B. VARMINT LISTING

(This Page Intentionally Left Blank)

```
VRC
      PROGRAM VRC (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE10,        VARP.2
     1 TAPE7,TAPE8,TAPE9,TAPE13)                                        VE3.10
      LARGE PTM(50000),CON(15000)                                       VE3.11
      INTEGER F,ERF,VDI,VDO,BUFL,PYML,PXML,CPRT,VNTP,VTP,PF,TD,TD1,TD2, INREAL.2
     1 PRFLM,CQF,SHIFT,OBLOC,OBTYP,FI,FO,FB,FM,OBTYPL                   INREAL.3
      REAL NU                                                           INREAL.4
C                                                                       INREAL.5
      DIMENSION U(1),V(1),W(1),P(1),CQ(1),CQF(1)                        DIMEN.2
     1 ,VOL(1)                                                          DIMEN.3
      DIMENSION SPEC(10),TSP(10)                                        VE3.1
      DIMENSION TA(1),QCON(1)                                           DIMEN.6
C                                                                       DIMEN.7
      COMMON/VRCOM/A(36000)                                             VE3.2
      COMMON/CONCEN/ NSPS(10),PMAXWS(10),PMINWS(10),BKGS(10),FACS(10)   VE3.3
     1 ,IPEDIT(10),IPEL(10),IPES(10),PMIN(10)                           VE3.4
      COMMON /AMIX/ C(600)                                              COMM.5
      COMMON/FL/FO(20),FB(8),FM                                         COMM.6
      COMMON/FLS/FI(9)                                                  COMM.7
      COMMON/VRCON/ALX,ALY,ALZ,RO,BUFL,CEP,DT,DX(80),DXC,DY(80),DYC,    VE3.5
     1 DZ(80),DZC,EMO,EMG,ERF,GPX,GPY,GPZ,GX,GY,GZ,IBP1,IBP2,IBR,IBRED, VE3.6
     2 IBRIN,IBRST,IGOP(6),IJKMX,IJKNC,IJK2,IJP1K,IPL,IPR,ITER,I2JK2,   COMM.10
     * IKP2,IPARTF,IPLTF,ISFLG,RIBJBKB,LPR,DROU,                        COMM.11
     *IJKCON,ICON,JCON,                                                 COMM.12
     3 I2K2,JBP1,JBP2,JBR,JBRED,JBRIN,JBRST,JNC,JPA,JPF,J2NC,KBP1,KBP2, COMM.13
     4 KBR,KBRED,KBRIN,KBRST,KFP,KNC,KPR,KPT,KWA,KWB,KWF,KWL,KWR,KWT,   COMM.14
     5 K2NC,LABEL(8),LNPART,LNVRCM,LPARTC,NCTP,NCVTP(6),NCYC,NGOP,NOPI, COMM.15
     *NOCONC,                                                           COMM.16
     * OBLOC(1200),OBTYP(1200),NOB,NU,                                  VE3.7
     6 NOVP,NOVS,NPNX,NPRT,NTPE,NWPC,NWPP,PDX,PDY,PDZ,PI,PRFLM,RDT,     COMM.18
     7 RDX(80),RDY(80),RDZ(80),RRAN(20),SIGMA,TD,TD1,TD2,TFIN,TIMET,TIP,VE3.8
     8 TPL,TPL1,TPR,TPRT,TTD,TWTP,TWTD,UI,UPI,VDI,VDO,VI,VNTP(6),VPI,   COMM.20
     9 VTP,WI,WPI,WRAN(20),X(80),XPCEN,Y(80),YPCEN,Z(80),ZPCEN          VE3.9
      COMMON /MODSET/ SCFLR(200)                                        COMM.22
      COMMON /SAFLM/ XCONV,ZCONV,FX1,FX2,FX3,FX4,FY1,FY2,FY3,FY4,       COMM.23
     1 FXL,FYB,PXL,PYB,IFLB(3),FFX3(75),FFY3(75)                        COMM.24
     2 ,ITCYC(3),XFC1(4),XFC2(4),YFC1(4),YFC2(4),PBTIM(2),XML,YML,ZML   COMM.25
     3 ,IFMID,IFTEST,IPLOPT,IPM(20),IPMM,IPPP,IPROJ,JPM(20),JPMM,JPPP,  COMM.26
     4 KPM(20),KPMM,KPPP,NEC,NMC,NPS,NPV,PEC(10),PMC(10),YCONV          COMM.27
     5 ,IFLAB,IFLIN,IFPNT,IFOPT,NCH,XFSP,XT,YFSP,YT                     COMM.28
      COMMON /ZLAST/ ZLAS                                               COMM.29
C                                                                       COMM.30
      EQUIVALENCE (A(1),U),(A(2),V),(A(3),W),(A(4),P),(A(5),CQ),        EQUIV.2
     1 (A(6),CQF), (A(7),VOL)                                           EQUIV.3
      EQUIVALENCE (C(1),TA,QCON)                                        EQUIV.7
C                                                                       EQUIV.8
    1 FORMAT(1H1)                                                       VARP.8
    2 FORMAT(8A10)                                                      VARP.9
    3 FORMAT(3(5X,I5),5X,I2)                                            VARP.10
    4 FORMAT(1H ,4X,4HIBR=,I5/5X,4HJBR=,I5/5X,4HKBR=,I5/3X,6HPRFLM=,I2) VARP.11
    5 FORMAT(1H ,57H **** NEGATIVE IBR READ IN WHICH TERMINATES PROGRAM VARP.12
     1*****)                                                            VARP.13
    6 FORMAT(1H ,21H THE PROBLEM DATE IS ,A10,5X,21H THE PROBLEM TIME ISVARP.14
     1 ,A10)                                                            VARP.15
  100 CALL SECOND(S1)                                                   VARP.16
    4 CALL DATE(XT)                                                     VARP.17
    6 PBTIM(1)=XT                                                       VARP.18
```

```
VRC
      10          CALL HOUR(XT)                                          VARP.19
      11          PBTIM(2)=XT                                            VARP.20
      12          VDI=5                                                  VARP.21
      13          VDO=6                                                  VARP.22
      15          WRITE(VDO,1)                                           VARP.23
      20          READ(VDI,3) IBR,JBR,KBR,PRFLM                          VARP.24
      34          WRITE(VDO,4) IBR,JBR,KBR,PRFLM                         VARP.25
      50          WRITE(VDO,6) PBTIM(1),PBTIM(2)                         VARP.26
      62      200 ERF=0                                                  VARP.27
      63          IF( PRFLM.GT.0 ) CALL FLMINIT                          VARP.28
      66      300 IF(IBR) 600,500,400                                    VARP.29
      70      400 READ(VDI,2) LABEL                                      VARP.30
      76          WRITE(VDO,2) LABEL                                     VARP.31
     104      420 CALL VRS                                               VARP.32
     105          IF( ERF.EQ.1 ) GO TO 700                               VARP.33
     107      500 CALL VRM                                               VARP.34
     110          IF( ERF.EQ.1 ) GO TO 700                               VARP.35
     112          GO TO 100                                              VARP.36
     113      600 WRITE(VDO,5)                                           VARP.37
     117      700 IF( PRFLM.GT.0 ) CALL FLMFIN                           VARP.38
     122          CALL EXIT                                              VARP.39
     123          END                                                    VARP.40


PROGRAM LENGTH INCLUDING I/O BUFFERS

04045

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1       - 000143    2       - 000145    3       - 000147    4       - 000152    5       - 000162    6       - 000172
100     - 000003    200     - 000063    300     - 000067    400     - 000071    420     - 000105    500     - 000110
600     - 000114    700     - 000120

BLOCK NAMES AND LENGTHS
  *LCM&- 0127710
VRCOM   - 106240/01 CONCEN  - 000132/02 AMIX    - 001130/03 FL      - 000035/04 FLS     - 000011/05 VRCON   - 006352/06
MODSET  - 000310/07 SAFLM   - 000447/08 ZLAST   - 000001/11

VARIABLE ASSIGNMENTS
A       - 000000/01 BKGS    - 000036/02 BUFL    - 000004/06 C       - 000000/03 CON     - 0072460/LC CPRT   - 000207
CQ      - 000009/01 CQF     - 000005/01 DX      - 000007/06 DY      - 000130/06 DZ      - 000251/06 ERF     - 000374/06
F       - 000204    FACS    - 000050/02 FB      - 000024/04 FFX3    - 000021/08 FFY3    - 000134/08 FI      - 000000/05
FM      - 000034/04 FO      - 000000/04 IBR     - 000405/06 IFLB    - 000016/08 IGOP    - 000411/06 IPEDII  - 000062/02
IPEL    - 000074/02 IPES    - 000106/02 IPM     - 000302/08 ITCYC   - 000247/08 JBR     - 000444/06 JPM     - 000331/08
KBR     - 000456/06 KPM     - 000357/08 LABEL   - 000475/06 NCVTP   - 000511/06 NSPS    - 000000/02 NU      - 005264/06
OBLOC   - 000523/06 OBTYP   - 003003/06 OBTYPL  - 000212    P       - 000003/01 PBTIM   - 000272/08 PEC     - 000411/08
PF      - 000210    PMAXWS  - 000012/02 PMC     - 000423/08 PMIN    - 000120/02 PMINWS  - 000024/02 PRFLM   - 005300/06
PTM     - 0000000/LC PXML   - 000206    PYML    - 000205    QCON    - 000000/03 RDX     - 005302/06 RDY     - 005422/06
RDZ     - 005542/06 RRAN    - 005662/06 SCFLM   - 000000/07 SHIFT   - 000211    SPEC    - 000213    S1      - 000237
TA      - 000000/03 TD      - 005707/06 TD1     - 005710/06 TD2     - 005711/06 TSP     - 000225    U       - 000000/01
V       - 000001/01 VDI     - 005726/06 VDO     - 005727/06 VNTP    - 005731/06 VOL     - 000006/01 VTP     - 005740/06
W       - 000002/01 WRAN    - 005743/06 X       - 005767/06 XFC1    - 000252/08 XFC2    - 000256/08 XT      - 000444/08
Y       - 006110/06 YFC1    - 000262/08 YFC2    - 000266/08 Z       - 006231/06
```

VRS

```
      SUBROUTINE VRS                                                    VARP.41
      LARGE PTM(30000), CON(15000)                                      VE3.12
      INTEGER F,ERF,VDI,VDO,BUFL,PYML,PXML,CPRT,VNTP,VTP,PF,TD,TD1,TD2, INREAL.2
     1 PRFLM,CQE,SHIFT,OBLOC,OBTYP,FI,FO,FB,FM,OBTYPL                   INREAL.3
      REAL NU                                                           INREAL.4
C                                                                       INREAL.5
      DIMENSION U(1),V(1),W(1),P(1),CQ(1),CQF(1)                        DIMEN.2
     1 ,VOL(1)                                                          DIMEN.3
      DIMENSION SPEC(10),TSP(10)                                        VE3.1
      DIMENSION TA(1),QCON(1)                                           DIMEN.6
C                                                                       DIMEN.7
      COMMON/VRCOM/A(36000)                                             VE3.2
      COMMON/CONCEN/ NSPS(10),PMAXWS(10),PMINWS(10),BKGS(10),FACS(10)   VE3.3
     1 ,IPEDIT(10),IPEL(10),IPES(10),PMIN(10)                           VE3.4
      COMMON /AMIX/ C(600)                                              COMM.5
      COMMON/FL/FO(20),FB(8),FM                                         COMM.6
      COMMON/FLS/FI(9)                                                  COMM.7
      COMMON/VRCON/ALX,ALY,ALZ,BO,BUFL,CEP,DT,DX(80),DXC,DY(80),DYC,    VE3.5
     1 DZ(80),DZC,EM0,EM6,ERF,GPX,GPY,GPZ,GX,GY,GZ,IBP1,IBP2,IBR,IBRED, VE3.6
     2 IBRIN,IBRST,IGOP(6),IJKMX,IJKNC,IJK2,IJP1K,IPL,IPR,ITER,I2JK2,   COMM.10
     * IKP2,IPARTF,IPLTF,ISFLG,RIBJBKB,LPR,DROU,                        COMM.11
     *IJKCON,ICON,JCON,                                                 COMM.12
     3 I2K2,JBP1,JBP2,JBR,JBRED,JBRIN,JBRST,JNC,JPA,JPF,J2NC,KBP1,KBP2, COMM.13
     4 KBR,KBRED,KBRIN,KBRST,KFP,KNC,KPB,KPT,KWA,KWB,KWF,KWL,KWR,KWT,  COMM.14
     5 K2NC,LABEL(8),LNPART,LNVRCM,LPARTC,NCTP,NCVTP(6),NCYC,NGOP,NOPT, COMM.15
     *NOCONC,                                                           COMM.16
     * OBLOC(1200),OBTYP(1200),NOB,NU,                                  VE3.7
     6 NOVP,NOVS,NPMX,NPRT,NTPF,NWPC,NWPP,PDX,PDY,PDZ,PI,PRFLM,RDT,     COMM.18
     7 RDX(80),RDY(80),RDZ(80),RRAN(20),SIGMA,TD,TD1,TD2,TFIN,TIMET,TIP,VE3.8
     8 TPL,TPLI,TPR,TPRT,TTD,TWTP,TWTD,UI,UPI,VDI,VDO,VI,VNTP(6),VPI,   COMM.20
     9 VTP,WI,WPI,WRAN(20),X(80),XPCEN,Y(80),YPCEN,Z(80),ZPCEN          VE3.9
      COMMON /MODSET/ SCFLM(200)                                        COMM.22
      COMMON /SAFLN/ XCONV,ZCONV,FX1,FX2,FX3,FX4,FY1,FY2,FY3,FY4,       COMM.23
     1 FXL,FYB,PXL,PYB,IFLB(5),FFX3(75),FFY3(75)                        COMM.24
     2 ,ITCYC(3),XFC1(4),XFC2(4),YFC1(4),YFC2(4),PBTIM(2),XML,YML,ZML   COMM.25
     3 ,IFMID,IFTEST,IPLOPT,IPM(20),IPMM,IPPP,IPROJ,JPM(20),JPMM,JPPP,  COMM.26
     4 KPM(20),KPMM,KPPP,NEC,NMC,NPS,NPV,PEC(10),PMC(10),YCONV          COMM.27
     5 ,IFLAB,IFLIN,IFPNT,IFOPT,NCH,XFSP,XT,YFSP,YT                     COMM.28
      COMMON /ZLAST/ ZLAS                                               COMM.29
C                                                                       COMM.30
      EQUIVALENCE (A(1),U),(A(2),V),(A(3),W),(A(4),P),(A(5),CQ),        EQUIV.2
     1 (A(6),CQF), (A(7),VOL)                                           EQUIV.3
      EQUIVALENCE (C(1),TA,QCON)                                        EQUIV.7
C                                                                       EQUIV.8
    1 FORMAT(1H ,47H *** ENTERED SUBROUTINE VRS - SETUP PROGRAM ***)    VARP.47
    2 FORMAT(9(F8.3))                                                   VARP.48
    3 FORMAT(6(I2),2(F8.3))                                             VARP.49
    4 FORMAT(1H ,44H *** ERROR 001 - MESH DIMENSION EXCEEDED ***)       VARP.50
    5 FORMAT(6X,6(I2))                                                  VARP.51
    6 FORMAT(1H ,6H NGOP ,6(2X,I2))                                     VARP.52
    7 FORMAT(3(5X,I2,1X,I3))                                            VARP.53
    8 FORMAT(1H ,75H *** ERROR 002 - ATTEMPTED TO READ IN MORE THAN 100 VARP.54
     1CONTOUR PLOT VALUES ***)                                          VARP.55
    9 FORMAT(1H ,6H VNTP=,I2,4X,7H NCVTP=,I3)                           VARP.56
   10 FORMAT(5(F8.3),I2)                                                VARP.57
   11 FORMAT(1H ,54H *** ERROR 005 - PARTICLE ARRAY DIMENSION EXCEEDED *VARP.58
```

```
      1**)                                                              VARP.59
   12 FORMAT(3(I3),F8.3)                                                VARP.60
   13 FORMAT(3(I3))                                                     VARP.61
   14 FORMAT(3(F8.3),I6,3(F8.3))                                        VARP.62
   15 FORMAT(1H ,71H *** ERROR 009 - NUMBER OF CONTOUR VARIABLES TO BE PVARP.63
     1LOTTED EXCEEDED ***)                                              VARP.64
   16 FORMAT(5X,I3,1X,20(I3))                                           VARP.65
   17 FORMAT(3X,I3,3X,I3,10(F6.2))                                      VARP.66
   18 FORMAT(3(5X,I3),6X,I2)                                            VARP.67
   20 FORMAT(1H ,* ---ERROR 010, OBSTACLE ARRAYS DIMENSION EXCEEDED*)   VARP.68
   21 FORMAT(1H ,2X,*NOB*,20X,*OBTYP*,5X,*OBLOC*,                       VARP.69
     1        5X,*I*,5X,*J*,5X,*K*,/)                                   VARP.70
   22 FORMAT(1H ,4X,* NREC=*,I6)                                        VARP.71
   23 FORMAT(1H ,4X,* IL= *,I3,* IR= *,I3,* JA= *,I3,* JF= *,I3,        VARP.72
     1              * KB= *,I3,* KT= *,I3,* OB= *,F6.1)                 VARP.73
   24 FORMAT(3X,I3,5X,O20,5X,I5,4X,I3,4X,I3,4X,I3)                      VARP.74
   25 FORMAT(F8.3,I2)                                                   VARP.75
   26 FORMAT(1H ,3X,4H NU=,F8.3/ 4X,7H ISFLG=,I2)                       VARP.76
   30 FORMAT(1H ,4X,4HDXC=,1PE12.5/5X,4HDYC=,E12.5/5X,4HDZC=,E12.5/     VARP.77
     1 6X,3HGX=,E12.5/6X,3HGY=,E12.5/6X,3HGZ=,E12.5/5X,4HALX=,E12.5/    VARP.78
     2 5X,4HALY=,E12.5/5X,4HALZ=,E12.5)                                 VARP.79
   31 FORMAT(1H ,4X,4HKWR=,I3/5X,4HKWL=,I3/5X,4HKWF=,I3/5X,4HKWA=,I3/   VARP.80
     1 5X,4HKWT=,I3/5X,4HKWB=,I3/6X,3HB0=,1PE12.5/5X,4HCEP=,E12.5)      VARP.81
   32 FORMAT(1H ,5X,3HDT=,1PE12.5/4X,5HTPLT=,E12.5/4X,5HTPRT=,E12.5/    VARP.82
     1 4X,5HTFIN=,E12.5/4X,5HTWTD=,E12.5/4X,5HNPRT=,I2)                 VARP.83
   33 FORMAT(1H ,3X,5HNGOP=,I3/4X,5HNOVP=,I3/5X,4HLPR=,I3/              VARP.84
     1 4X,5HDROU=,1PE12.5)                                              VARP.85
   34 FORMAT(1H ,3X,5HNWPC=,I3/4X,5HNWPP=,I3/4X,5HNTPE=,I3)             VARP.86
   35 FORMAT(1H ,4X,4HGPX=,1PE12.5/5X,4HGPY=,E12.5/5X,4HGPZ=,E12.5/     VARP.87
     1 4X,5HNPMX=,I6/6X,3HUT=,E12.5/6X,3HVI=,E12.5/6X,3HWI=,E12.5)      VARP.88
   36 FORMAT(1H ,6H IPMM ,I3,1X,20(I3))                                 VARP.89
   37 FORMAT(1H ,6H JPMM ,I3,1X,20(I3))                                 VARP.90
   38 FORMAT(1H ,6H KPMM ,I3,1X,20(I3))                                 VARP.91
   39 FORMAT(1H ,5H NPV ,I3,2X,5H NEC ,I3,10(2X,F6.2))                  VARP.92
   40 FORMAT(1H ,6H IPPP ,I3,2X,6H JPPP ,I3,2X,6H KPPP ,I3/3X,6HIPROJ=, VARP.93
     1 I2)                                                              VARP.94
   41 FORMAT(1H ,5H NPV ,I3,2X,5H NMC ,I3,10(2X,F6.2))                  VARP.95
C *****  THIS SECTION, 100 , READS AND PRINTS THE INPUT DATA            VARP.96
  100 WRITE(VDO,1)                                                      VARP.97
      READ(VDI,2)  DXC,DYC,DZC,GX,GY,GZ,ALX,ALY,ALZ                     VARP.98
      READ(VDI,3)  KWR,KWL,KWF,KWA,KWT,KWB,B0,CEP                       VARP.99
      READ(VDI,10) DT,TPLT,TPRT,TFIN,TWTD,NPRT                          VARP.100
      READ(VDI,12) NGOP,NOVP,LPR,DROU                                   VARP.101
      READ(VDI,13) NWPC,NWPP,NTPE                                       VARP.102
      READ(VDI,14) GPX,GPY,GPZ,NPMX,UI,VI,WI                            VARP.103
      READ(VDI,25) NU,ISFLG                                             VARP.104
      WRITE(VDO,30) DXC,DYC,DZC,GX,GY,GZ,ALX,ALY,ALZ                    VARP.105
      WRITE(VDO,31) KWR,KWL,KWF,KWA,KWT,KWB,B0,CEP                      VARP.106
      WRITE(VDO,32) DT,TPLT,TPRT,TFIN,TWTD,NPRT                         VARP.107
      WRITE(VDO,33) NGOP,NOVP,LPR,DROU                                  VARP.108
      WRITE(VDO,34) NWPC,NWPP,NTPE                                      VARP.109
      WRITE(VDO,35) GPX,GPY,GPZ,NPMX,UI,VI,WI                           VARP.110
      WRITE(VDO,26) NU,ISFLG                                            VARP.111
C *****  THIS SECTION, 120 , CALCULATES CONSTANTS                       VARP.112
  120 TIMET=0.0                                                         VARP.113
      TD=0                                                              VARP.114
```

```
332          NCYC=0                                                    VARP.115
333          RDT=1./DT                                                 VARP.116
335          PI=3.14159265                                             VARP.117
336          EN6=1.E-6                                                 VARP.118
337          EM0=-EM6                                                  VARP.119
340          TPL=TPLT                                                  VARP.120
342          TPR=TPRT                                                  VARP.121
343          ITD=TATD                                                  VARP.122
345          FO(1)=0B                                                  VARP.123
345          FO(2)=1B                                                  VARP.124
347          FB(1)= 1000B                                              VARP.125
350          FB(2)= 1000000B                                           VARP.126
352          FB(3)= 10000000B                                          VARP.127
353          FB(4)= 1000000000B                                        VARP.128
355          FB(5)= 1000000000000B                                     VARP.129
356          FB(6)= 100000000000000B                                   VARP.130
360          FB(7)= 1000000000000000B                                  VARP.131
361          FM=777B                                                   VARP.132
       C ***** THIS SECTION , 124 , CALCULATES CELL INDEX QUANITIES    VARP.133
363      124 IBP1=IBR+1                                                VARP.134
365          KBP1=KBR+1                                                VARP.135
366          IBP2=IBR+2                                                VARP.136
370          KBP2=KBR+2                                                VARP.137
370          I2K2=IBP2*KBP2*NWPC                                       VARP.138
373          JBP1=JBR+1                                                VARP.139
375          JBP2=JBR+2                                                VARP.140
375          IJK2=IBP2*JBP2*KBP2                                       VARP.141
400          IJKNC=IJK2*NWPC                                           VARP.142
402          IJKMX=IJKNC+I2K2                                          VARP.143
403          K2NC=KBP2*NWPC                                            VARP.144
405          IJP1K=I2K2*JBP1                                           VARP.145
407          I2JK2=I2K2*JBR                                            VARP.146
411          KNC=KBR*NWPC                                              VARP.147
413          JNC=JBR*NWPC                                              VARP.148
414          J2NC=JBP2*NWPC                                            VARP.149
416          IKP2=IBR*K2NC                                             VARP.150
421          I1K2=IBP1*KBP2*NWPC                                       VARP.151
424          IM1K2=(IBR-1)*K2NC                                        VARP.152
426          RIBJKB=1./FLOAT( IBR*JBR*KBR )                            VARP.153
       C ***** THIS SECTION, 125 , CHECKS FOR ARRAY DIMENSIONS EXCEED  VARP.154
433          LNVRCM= LOCF(NSPS(1)) -LOCF(A) +1                         VE3.13
441          LNPART= LOCF(C)- LOCF(NSPS(1))                            VE3.14
447      125 LPARTC= 30000                                             VE3.15
       C   THIS IS THE SIZE OF LARGE CORE STORAGE TO CLEAR PTM         VE3.16
450          BUFL=LOCF(ZLAS)-LOCF(A)+1                                 VARP.158
456          IF( IJKMX.GT.LNVRCM ) GO TO 127                           VARP.159
462          GO TO 130                                                 VARP.161
       C ***** *** ERROR 001 - IBAR OR KBAR ARRAY DIMENSION EXCEEDED ***  VARP.162
462      127 WRITE(VDU,4)                                              VARP.163
466          ERE=1                                                     VARP.164
467          GO TO 99999                                               VARP.165
       C *****  THIS SECTION, 130, INPUTS OBSTACLES                    VARP.170
470      130 NOB=0                                                     VARP.171
471          READ(VDI,12)  NREC                                        VARP.172
477          WRITE(VDO,22)  NREC                                       VARP.173
505          IF( NREC.LE.0 ) GO TO 135                                 VARP.174
```

```
507          IF( NREC.GT.200) GO TO 134                                  VARP.175
       C *****  SET OBSTACLE ARRAYS                                      VARP.176
512          DO 133  L=1,NREC                                            VARP.177
513          READ(VDI,3)  IL,IR,JA,JF,KB,KT,OB                           VARP.178
534          WRITE(VDO,23) IL,IR,JA,JF,KB,KT,OB                          VARP.179
556          IBB=IFIX(OB)+1                                              VARP.180
561          DO 132 I=IL,IR                                              VARP.181
563          DO 132 J=JA,JF                                              VARP.182
565          DO 132 K=KB,KT                                              VARP.183
567          IOB=0                                                       VARP.184
570          GO TO (1000,1010,1020,1030,1040,1050,1060,1070,1080,1090,   VARP.185
            1        1100,1110,1120) , IBB                                VARP.186
611     1010 IF (( KT-K) - (I-IL) ) 132,1200,1000                        VARP.187
616     1020 IF (( I-IL) - (K-KB) ) 132,1200,1000                        VARP.188
623     1030 IF (( KT-K) - (J-JA) ) 132,1200,1000                        VARP.189
630     1040 IF (( J-JA) - (K-KB) ) 132,1200,1000                        VARP.190
635     1050 IF (( K-KB) - (I-IL) ) 132,1200,1000                        VARP.191
642     1060 IF (( I-IL) - (KT-K) ) 132,1200,1000                        VARP.192
647     1070 IF (( K-KB) - (I-JA) ) 132,1200,1000                        VARP.193
654     1080 IF (( J-JA) - (KT-K) ) 132,1200,1000                        VARP.194
661     1090 IF (( JF-J) - (I-IL) ) 132,1200,1000                        VARP.195
666     1100 IF (( J-JA) - (I-IL) ) 132,1200,1000                        VARP.196
673     1110 IF (( I-IL) - (J-JA) ) 132,1200,1000                        VARP.197
700     1120 IF (( I-IL) - (JF-J) ) 132,1200,1000                        VARP.198
705     1200  IOB=IFIX(OB)                                               VARP.199
707     1000  NOB=NOB+1                                                  VARP.200
       C *****  CHECK FOR OBSTACLE ARRAYS EXCEEDED                       VARP.201
711          IF(NOB.GT.1200) GO TO 134                                   VF3.17
714          IJK= (I-1)*K2NC + (J-1)*I2K2 + (K-1)*NWPC + 1               VARP.203
727          OBLOC(NOB)=IJK                                              VARP.204
730          OBTYP(NOB)=IOB                                              VARP.205
732      132 CONTINUE                                                    VARP.206
742      133 CONTINUE                                                    VARP.207
       C ****  SET OBSTACLE SURFACE FLAG FOR NO SLIP B.C.                VARP.208
744          DO 1335 L=1,NOB                                             VARP.209
747          IJK= OBLOC(L)                                               VARP.210
750          KT= FB(1) * FO(2)                                           VARP.211
752          IR= FB(2) * FO(2)                                           VARP.212
755          IL= FB(3) * FO(2)                                           VARP.213
757          JA= FB(4) * FO(2)                                           VARP.214
762          JF= FB(5) * FO(2)                                           VARP.215
764          KB= FB(6) * FO(2)                                           VARP.216
767          IJKM= IJK-NWPC                                              VARP.217
770          IJKP= IJK+NWPC                                              VARP.218
771          IJMK= IJK-I2K2                                              VARP.219
773          IJPK= IJK+I2K2                                              VARP.220
773          IMJK= IJK-K2NC                                              VARP.221
775          IPJK= IJK+K2NC                                              VARP.222
776          DO 1330 LL=1,NOB                                            VARP.223
777          IJKL= OBLOC(LL)                                             VARP.224
1000         IF( IJKL.EQ.IJK) GO TO 1330                                 VARP.225
1002         IF(IJKL.EQ.IJKM) KB= FB(6)*FO(1)                            VARP.226
1006         IF(IJKL.EQ.IJKP) KT= FB(1)*FO(1)                            VARP.227
1013         IF(IJKL.EQ.IJMK) JA= FB(4)*FO(1)                            VARP.228
1017         IF(IJKL.EQ.IJPK) JF= FB(5)*FO(1)                            VARP.229
1023         IF(IJKL.EQ.IMJK) IL= FB(3)*FO(1)                            VARP.230
```

```
1027          IF(IJKL.EQ.IPJK) IR= FB(2)*FO(1)                                  VARP.231
1033     1330 CONTINUE                                                          VE3.18
1037          IJK= IJK-1                                                        VARP.232
1040          JJ= 1+ IJK/I2K2                                                   VARP.233
1043          II= 1+ (IJK- (I2K2*(JJ-1))) /K2NC                                 VARP.234
1051          KK= 1+ (IJK-(K2NC*(II-1))- (I2K2*(JJ-1)))/ NWPC                   VARP.235
1062          IF( II.EQ.2) IL= FB(3)* FO(1)                                     VARP.236
1067          IF( II.EQ. IBP1) IR= FB(2)* FO(1)                                 VARP.237
1073          IF( JJ.EQ.2) JA= FB(4)* FO(1)                                     VARP.238
1077          IF( JJ.EQ.JBP1) JF= FB(5)* FO(1)                                  VARP.239
1103          IF( KK.EQ.2) KB= FB(6)* FO(1)                                     VARP.240
1110          IF( KK.EQ.KBP1) KT= FB(1)* FO(1)                                  VARP.241
1114          OBTYP(L)= (((((( OBTYP(L).OR.KT) .OR.IR) .OR.IL) .OR.JA) .OR.JF)  VARP.243
             1 .OR.KB                                                           VARP.244
1122     1335 CONTINUE                                                          VARP.245
        C ***** *** ERROR 010 - ATTEMPTED TO READ IN TO MANY OBSTACLE CELLS     VARP.246
1125          IF(NOB.LE.1200) GO TO 1310                                        VE3.19
1127      134 WRITE(VDO,20)                                                     VARP.248
1133          ERF=1                                                             VARP.249
1134          GO TO 99999                                                       VARP.250
1135     1310 CONTINUE                                                          VARP.251
1135          WRITE(VDO,21)                                                     VARP.252
1141          DO 1320 I=1,NOB                                                   VARP.253
1145          IJK=OBLOC(I)-1                                                    VARP.254
1147          JJ=1+IJK/I2K2                                                     VARP.255
1151          II=1+(IJK-(I2K2*(JJ-1)))/K2NC                                     VARP.256
1157          KK=1+(IJK-(K2NC*(II-1))-(I2K2*(JJ-1)))/NWPC                       VARP.257
1171     1320 WRITE(VDO,24) I,OBTYP(I),OBLOC(I),II,JJ,KK                        VARP.258
        C ***** THIS SECTION , 135 , READS AND PRINTS GRAPH OPTIONS FOR SUB VRIOVARP.259
1216      135 IF( NGOP.LT.1 ) GO TO 140                                         VARP.260
1221          IF( NGOP.GT.6 ) NGOP=6                                            VARP.261
1224          READ(VDI,5) (IGOP(I),I=1,NGOP)                                    VARP.262
1233          WRITE(VDO,6) (IGOP(I),I=1,NGOP)                                   VARP.263
        C ***** THIS SECTION, 140 , READS AND PRINTS CONTOUR PLOT VALUES        VARP.264
1242      140 LT=0                                                              VARP.265
1243          IF( NOVP.LT.1 ) GO TO 160                                         VARP.266
1246          READ(VDI,7) (VNTP(M),NCVTP(M),M=1,NOVP)                           VARP.267
1264          DO 149 M=1,NOVP                                                   VARP.268
1266          WRITE(VDO,9) VNTP(M),NCVTP(M)                                     VARP.269
1277          LT=NCVTP(M)                                                       VARP.270
1301          IF( VNTP(M).GT.6 ) GO TO 146                                      VARP.271
1305          IF( LT.LT.101 ) GO TO 148                                         VARP.272
        C ***** *** ERROR 002 - ATTEMPTED TO READ IN MORE THAN 100 CONTUR PLT VAVARP.273
1307      145 WRITE(VDO,8)                                                      VARP.274
1313          ERF=1                                                             VARP.275
1314          GO TO 99999                                                       VARP.276
        C ***** *** ERROR 009 - NUMBER OF CONTOURS VARIABLES TO BE PLOTTED EXCEDVARP.277
1315      146 WRITE(VDO,15)                                                     VARP.278
1321          ERF=1                                                             VARP.279
1322          GO TO 99999                                                       VARP.280
1323      148 CONTINUE                                                          VARP.281
1323      149 CONTINUE                                                          VARP.282
1326      160 CONTINUE                                                          VARP.283
        C ***** THIS SECTION , 180 , READS IN GRAPHICS QUANTITIES.              VARP.284
1326          IF( PRFLM.LT.1 ) GO TO 200                                        VARP.285
        C ***** THIS SECTION READS IN PLANES TO PLOT.                           VARP.286
```

```
1331        READ(VDI,16) IPMM,(IPM(M),M=1,IPMM)                          VARP.287
1341        READ(VDI,16) JPMM,(JPM(M),M=1,JPMM)                          VARP.288
1352        READ(VDI,16) KPMM,(KPM(M),M=1,KPMM)                          VARP.289
1363        READ(VDI,18) IPPP,JPPP,KPPP,IPROJ                            VARP.290
1377        WRITE(VDO,36) IPMM,(IPM(M),M=1,IPMM)                         VARP.291
1410        WRITE(VDO,37) JPMM,(JPM(M),M=1,JPMM)                         VARP.292
1421        WRITE(VDO,38) KPMM,(KPM(M),M=1,KPMM)                         VARP.293
1432        WRITE(VDO,40) IPPP,JPPP,KPPP,IPROJ                           VARP.294
      C ***** THIS SECTION READS IN EYE,MESH PROJECTIONIAL COORDINATES.  VARP.295
1446        IF( IPROJ.LT.1 ) GO TO 200                                   VARP.296
1451        READ(VDI,17)  NPV,NEC,(PEC(I),I=1,NEC)                       VARP.297
1463        READ(VDI,17) NPV,NMC,(PMC(I),I=1,NMC)                        VARP.298
1476        WRITE(VDO,39) NPV,NEC,(PEC(I),I=1,NEC)                       VARP.299
1511        WRITE(VDO,41) NPV,NMC,(PMC(I),I=1,NMC)                       VARP.300
1524        DO 189 I=1,NEC                                               VARP.301
1526        PEC(I)=PEC(I)+EM6                                            VARP.302
1530        IF( PEC(I).LT.0.0 ) PEC(I)=PEC(I)-2.*EM6                     VARP.303
1535    189 CONTINUE                                                     VARP.304
      C ***** THIS SECTION, 200 , INITIALIZES ALL ARRAYS TO ZERO         VARP.305
1540    200 L1=1                                                         VARP.306
1541        L2=IJKMX                                                     VARP.307
1543    210 DO 219 L=L1,L2                                               VARP.308
1550        A(L)=0.0                                                     VARP.309
1551    219 CONTINUE                                                     VARP.310
1552    230 L2=LPARTC                                                    VARP.311
1554        DO 239 L=L1,L2                                               VARP.312
1561        PTM(L)=0.0                                                   VARP.313
1562    239 CONTINUE                                                     VARP.314
      C ***** THIS SECTION , 300 , INITIALIZES VELOCITIES AND DX,DY,DZ.  VARP.315
1563    300 JJ=1-I2K2                                                    VARP.316
1565        DO 329 J=1,JBP2                                              VARP.317
1570        JJ=JJ+I2K2                                                   VARP.318
1571        KK=-K2NC                                                     VARP.319
1572        DY(J)=DYC                                                    VARP.320
1574        RDY(J)=1./DY(J)                                              VARP.321
1576        DO 329 I=1,IBP2                                              VARP.322
1600        KK=KK+K2NC                                                   VARP.323
1601        LWPC=-NWPC                                                   VARP.324
1602        DX(I)=DXC                                                    VARP.325
1604        DX(I)=DXC                                                    VARP.326
1605        RDX(I)=1./DX(I)                                              VARP.327
1606        DO 329 K=1,KBP2                                              VARP.328
1610        LWPC=LWPC+NWPC                                               VARP.329
1612        IJK=JJ+KK+LWPC                                               VARP.330
1614        DZ(1)=DZC                                                    VARP.331
1616        DZ(K)=DZC                                                    VARP.332
1617        RDZ(K)= 1./DZ(K)                                             VARP.333
1620        U(IJK)=UI                                                    VARP.334
1623        V(IJK)=VI                                                    VARP.335
1624        W(IJK)=WI                                                    VARP.336
1626        IF(I.EQ.1.OR.J.EQ.1.OR.K.EQ.1) GO TO 329                     VARP.337
1637        IF( I.EQ.IBP2.OR.J.EQ.JBP2.OR.K.EQ.KBP2) GO TO 329           VARP.338
1651        VOL(IJK)= DX(I)*DY(J)*DZ(K)                                  VARP.339
1656    329 CONTINUE                                                     VARP.340
      C ***** THIS SECTION,330-339, INITIALIZES VOLUME SEGMENTS.         VARP.341
1666        D8= 1./8.                                                    VARP.342
```

```
1667          JJ=1-I2K2                                                VARP.343
1671          DO 339 J=1,JBP1                                          VARP.344
1673          JJ= JJ+I2K2                                              VARP.345
1674          KK= -K2NC                                                VARP.346
1676          DO 339 I=1,IBP1                                          VARP.347
1677          KK= KK+K2NC                                              VARP.348
1700          LWPC= -NWPC                                              VARP.349
1702          DO 339 K=1,KBP1                                          VARP.350
1713          LWPC= LWPC+NWPC                                          VARP.351
1715          IJK= JJ+KK+LWPC                                          VARP.352
1717          IPJK= IJK+K2NC                                           VARP.353
1720          IJPK= IJK+I2K2                                           VARP.354
1722          IJKP= IJK+NWPC                                           VARP.355
1724          IPJKP= IJKP+K2NC                                         VARP.356
1725          IJPKP= IJPK+NWPC                                         VARP.357
1725          IPJPK= IPJK +I2K2                                        VARP.358
1726          IPJPKP= IPJPK+NWPC                                       VARP.359
1727          VOL(IJK)= D8*(VOL(IJK)+ VOL(IPJK)+ VOL(IJPK)+ VOL(IJKP)+ VOL(IPJKPVARP.360
             1 ) + VOL(IJPKP)+ VOL(IPJPK)+ VOL(IPJPKP))                VARP.361
1745      339 CONTINUE                                                VARP.362
        C ***** THIS SECTION , 350-379 , INITIALIZES MESH COORDINATES. VARP.363
1762          XT=-DX                                                   VARP.364
1763          DO 359 I=1,IBP2                                          VARP.365
1771          XT=XT+DX(I)                                              VARP.366
1772      359 X(I)=XT                                                  VARP.367
1774          ZT=-DZ                                                   VARP.368
1775          DO 369 K=1,KBP2                                          VARP.369
2003          ZT=ZT+DZ(K)                                              VARP.370
2004      369 Z(K)=ZT                                                  VARP.371
2006          YT=-DY                                                   VARP.372
2007          DO 379 J=1,JBP2                                          VARP.373
2015          YT=YT+DY(J)                                              VARP.374
2016      379 Y(J)=YT                                                  VARP.375
2020          XML=X(IBP1)                                              VARP.376
2021          YML=Y(JBP1)                                              VARP.377
2023          ZML=Z(KBP1)                                              VARP.378
        C ***** THIS SECTION, 380-419, CALCULATES FINAL CELL VOLUMES   VARP.379
        C       CELL VOLUME IS CENTERED IN THE UPPER RIGHT FORE CORNER OF VARP.380
        C       THE CELL.                                              VARP.381
2026          PRINT 49                                                 VARP.382
           49 FORMAT( 5H    I,4H   J,4H   K,6H  IJK ,12HFINAL VOLUME /) VARP.383
2031          D16= 1./16.                                              VARP.384
2032          IF(NOB.LT.1) GO TO 400                                   VARP.385
2035          DO 399 L=1,NOB                                           VARP.386
2040          IOB= OBTYP(L).AND.FM                                     VARP.387
2042          IOB= IOB+1                                               VARP.388
2043          IJK= OBLOC(L)-1                                          VARP.389
2044          JJ= 1+IJK/I2K2                                           VARP.390
2047          II= 1+ (IJK-(I2K2*(JJ-1))) /K2NC                         VARP.391
2055          KK= 1+ (IJK-(K2NC *(II-1))-(I2K2*(JJ-1))) /NWPC          VARP.392
2067          IJK= IJK+1                                               VARP.393
2070          VIJKH= D16*DX(II)*DY(JJ)*DZ(KK)                          VARP.394
2075          DBV= 2.*VIJKH                                            VARP.395
2077          ILJK= IJK -K2NC                                          VARP.396
2101          ILJAK= ILJK-I2K2                                         VARP.397
2102          IJAK= IJK -I2K2                                          VARP.398
```

```
2103          IJKB= IJK -NAPC                                          VARP.399
2105          ILJKB= IJKB-K2NC                                         VARP.400
2105          ILJAKB= ILJKB-I2K2                                       VARP.401
2106          IJAKB= IJKB-I2K2                                         VARP.402
2110          GO TO (380,381,382,383,384,385,386,387,388,389,390,391,392), IUB  VARP.403
2131     380  IW1= IJK                                                 VARP.404
2132          IW2= ILJK                                                VARP.405
2134          IW3= ILJAK                                               VARP.406
2135          IW4= IJAK                                                VARP.407
2137          IW5= IJKB                                                VARP.408
2140          IW6= ILJKB                                               VARP.409
2142          IW7= ILJAKB                                              VARP.410
2143          IW8= IJAKB                                               VARP.411
2145          GO TO 394                                                VARP.412
2146     381  IH1= IJAKB                                               VARP.413
2147          IH2= ILJAK                                               VARP.414
2151          IH3= IJKB                                                VARP.415
2152          IH4= ILJK                                                VARP.416
2154          IW1= ILJAKB                                              VARP.417
2155          IW2= ILJKB                                               VARP.418
2157          GO TO 393                                                VARP.419
2160     382  IH1= ILJAKB                                              VARP.420
2161          IH2= IJAK                                                VARP.421
2163          IH3= ILJKB                                               VARP.422
2164          IH4= IJK                                                 VARP.423
2166          IW1= IJAKB                                               VARP.424
2167          IW2= IJKB                                                VARP.425
2171          GO TO 393                                                VARP.426
2172     383  IH1= ILJAK                                               VARP.427
2173          IH2= ILJKB                                               VARP.428
2175          IH3= IJAK                                                VARP.429
2176          IH4= IJKB                                                VARP.430
2200          IW1= ILJAKB                                              VARP.431
2201          IW2= IJAKB                                               VARP.432
2203          GO TO 393                                                VARP.433
2204     384  IH1= ILJAKB                                              VARP.434
2205          IH2= ILJK                                                VARP.435
2207          IH3= IJAKB                                               VARP.436
2210          IH4= IJK                                                 VARP.437
2212          IW1= ILJKB                                               VARP.438
2213          IW2= IJKB                                                VARP.439
2215          GO TO 393                                                VARP.440
2216     385  IH1= ILJAKB                                              VARP.441
2217          IH2= IJAK                                                VARP.442
2221          IH3= ILJKB                                               VARP.443
2222          IH4= IJK                                                 VARP.444
2224          IW1= ILJAK                                               VARP.445
2225          IW2= ILJK                                                VARP.446
2227          GO TO 393                                                VARP.447
2230     386  IH1= IJAKB                                              VARP.448
2231          IH2= ILJAK                                               VARP.449
2233          IH3= IJKB                                                VARP.450
2234          IH4= ILJK                                                VARP.451
2236          IW1= IJAK                                                VARP.452
2237          IW2= IJK                                                 VARP.453
2241          GO TO 393                                                VARP.454
```

```
2242    387 IH1= ILJK                                          VARP.455
2243        IH2= ILJAKB                                        VARP.456
2245        IH3= IJK                                           VARP.457
2246        IH4= IJAKB                                         VARP.458
2250        IW1= ILJAK                                         VARP.459
2251        IW2= IJAK                                          VARP.460
2253        GO TO 393                                          VARP.461
2254    388 IH1= IJAK                                          VARP.462
2255        IH2= IJKB                                          VARP.463
2257        IH3= ILJAK                                         VARP.464
2260        IH4= ILJKB                                         VARP.465
2262        IW1= IJK                                           VARP.466
2263        IW2= ILJK                                          VARP.467
2265        GO TO 393                                          VARP.468
2266    389 IH1= IJAKB                                         VARP.469
2267        IH2= ILJKB                                         VARP.470
2271        IH3= IJAK                                          VARP.471
2272        IH4= ILJK                                          VARP.472
2274        IW1= ILJAKB                                        VARP.473
2275        IW2= ILJAK                                         VARP.474
2277        GO TO 393                                          VARP.475
2300    390 IH1= ILJAKB                                        VARP.476
2301        IH2= IJKB                                          VARP.477
2303        IH3= ILJAK                                         VARP.478
2304        IH4= IJK                                           VARP.479
2306        IW1= ILJKB                                         VARP.480
2307        IW2= ILJK                                          VARP.481
2311        GO TO 393                                          VARP.482
2312    391 IH1= ILJAKB                                        VARP.483
2313        IH2= IJKB                                          VARP.484
2315        IH3= ILJAK                                         VARP.485
2316        IH4= IJK                                           VARP.486
2320        IW1= IJAKB                                         VARP.487
2321        IW2= IJAK                                          VARP.488
2323        GO TO 393                                          VARP.489
2324    392 IH1= IJAKB                                         VARP.490
2325        IH2= IJAK                                          VARP.491
2327        IH3= ILJKB                                         VARP.492
2330        IH4= ILJK                                          VARP.493
2332        IW1= IJKB                                          VARP.494
2333        IW2= IJK                                           VARP.495
2342    393 VOL(IH1)= VOL(IH1)-      VIJKH                     VARP.496
2344        VOL(IH2)= VOL(IH2)-      VIJKH                     VARP.497
2345        VOL(IH3)= VOL(IH3)-      VIJKH                     VARP.498
2346        VOL(IH4)= VOL(IH4)-      VIJKH                     VARP.499
2352    394 VOL(IW1)= VOL(IW1)-D8V                             VARP.500
2353        VOL(IW2)= VOL(IW2)-D8V                             VARP.501
2354        IF (IOB.GT.1) GO TO 399                            VARP.502
2366        VOL(IW3)= VOL(IW3)-D8V                             VARP.503
2370        VOL(IW4)= VOL(IW4)-D8V                             VARP.504
2371        VOL(IW5)= VOL(IW5)-D8V                             VARP.505
2372        VOL(IW6)= VOL(IW6)-D8V                             VARP.506
2374        VOL(IW7)= VOL(IW7)-D8V                             VARP.507
2375        VOL(IW8)= VOL(IW8)-D8V                             VARP.508
2376    399 CONTINUE                                           VARP.509
2401    400 CONTINUE                                           VARP.510
```

```
VRS
   2401      905 CONTINUE                                                    VARP.524
   2401          CALL FILMCO                                                 VARP.525
   2402    99999 RETURN                                                      VARP.526
   2403          END                                                         VARP.527
```

SUBPROGRAM LENGTH

03222

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

1	- 002407	2	- 002416	3	- 002421	4	- 002424	5	- 002433	6	- 002436		
7	- 002442	8	- 002445	9	- 002457	10	- 002464	11	- 002467	12	- 002477		
13	- 002502	14	- 002504	15	- 002510	16	- 002522	17	- 002525	18	- 002531		
20	- 002534	21	- 002543	22	- 002556	23	- 002562	24	- 002577	25	- 002604		
26	- 002607	30	- 002614	31	- 002636	32	- 002654	33	- 002671	34	- 002703		
35	- 002711	36	- 002727	37	- 002733	38	- 002737	39	- 002743	40	- 002751		
41	- 002761	49	- 003010	100	- 000002	120	- 000332	124	- 000364	125	- 000450		
127	- 000463	130	- 000471	132	- 000733	134	- 001130	135	- 001217	140	- 001243		
145	- 001310	146	- 001316	148	- 001324	149	- 001324	160	- 001327	189	- 001536		
200	- 001541	210	- 001544	230	- 001553	300	- 001564	329	- 001657	380	- 002132		
381	- 002147	382	- 002161	383	- 002173	384	- 002205	385	- 002217	386	- 002231		
387	- 002243	388	- 002255	389	- 002267	390	- 002301	391	- 002313	392	- 002325		
393	- 002336	394	- 002350	399	- 002377	400	- 002402	905	- 002402	1000	- 000710		
1010	- 000612	1020	- 000617	1030	- 000624	1040	- 000631	1050	- 000636	1060	- 000643		
1070	- 000650	1080	- 000655	1090	- 000662	1100	- 000667	1110	- 000674	1120	- 000701		
1200	- 000706	1310	- 001136	1330	- 001034	99999	- 002403						

BLOCK NAMES AND LENGTHS

*LCM&	- 0127710										
VRCOM	- 106240/01	CONCEN	- 000132/02	AMIX	- 001130/03	FL	- 000035/04	FLS	- 000011/05	VRCON	- 006352/06
MODSET	- 000310/07	SAFLM	- 000447/08	ZLAST	- 000001/11						

VARIABLE ASSIGNMENTS

A	- 000000/01	ALX	- 000000/06	ALY	- 000001/06	ALZ	- 000002/06	BKGS	- 000036/02	BUFL	- 000004/06
B0	- 000003/06	C	- 000000/03	CEP	- 000005/06	CON	- 0072460/LC	CPRT	- 003075	CQ	- 000004/01
CQF	- 000005/01	DROU	- 000435/06	D1	- 000006/06	DX	- 000007/06	DXC	- 000127/06	DY	- 000130/06
DYC	- 000250/06	DZ	- 000251/06	DZC	- 000371/06	D16	- 003174	D8	- 003166	D8V	- 003176
EM0	- 000372/06	EM6	- 000373/06	ERF	- 000374/06	F	- 003072	FACS	- 000050/02	FB	- 000024/04
FFX3	- 000021/08	FFY3	- 000134/08	FI	- 000000/05	FM	- 000034/04	FU	- 000000/04	GPX	- 000375/06
GPY	- 000376/06	GPZ	- 000377/06	GX	- 000400/06	GY	- 000401/06	GZ	- 000402/06	I	- 003141
IBB	- 003140	IBP1	- 000403/06	IBP2	- 000404/06	IBR	- 000405/06	IFLB	- 000016/08	IGOP	- 000411/06
IH1	- 003216	IH2	- 003217	IH3	- 003220	IH4	- 003221	II	- 003157	IJAK	- 003201
IJAKB	- 003205	IJK	- 003145	IJKB	- 003202	IJKL	- 003155	IJKM	- 003146	IJKMX	- 000417/06
IJKNC	- 000420/06	IJKP	- 003147	IJK2	- 000421/06	IJMK	- 003150	IJPK	- 003151	IJPKP	- 003170
IJP1K	- 000422/06	IKP2	- 000427/06	IL	- 003131	ILJAK	- 003200	ILJAKB	- 003204	ILJK	- 003177
ILJKB	- 003203	IMJK	- 003152	IM1K2	- 003126	IOB	- 003144	IPEDTT	- 000062/02	IPEL	- 000074/02
IPES	- 000106/02	IPJK	- 003153	IPJKP	- 003167	IPJPK	- 003171	IPJPKP	- 003172	IPM	- 000302/08
IPMM	- 000326/08	IPPP	- 000327/08	IPROJ	- 000330/08	IR	- 003132	ISELG	- 000432/06	ITCYC	- 000247/08
IW1	- 003206	IW2	- 003207	IW3	- 003210	IW4	- 003211	IW5	- 003212	IW6	- 003213
IW7	- 003214	IW8	- 003215	I1K2	- 003125	I2JK2	- 000426/06	I2K2	- 000441/06	J	- 003142
JA	- 003133	JBP1	- 000442/06	JBP2	- 000443/06	JBR	- 000444/06	JF	- 003134	JJ	- 003156
JNC	- 000450/06	JPM	- 000331/08	JPMM	- 000355/08	JPPP	- 000356/08	J2NC	- 000453/06	K	- 003143

```
      SUBROUTINE VRM                                                    VARP.724
      INTEGER F,ERF,VDI,VDO,BUFL,PYML,PXML,CPRT,VNTP,VIP,PF,ID,ID1,ID2, INREAL.2
     1 PRFLM,COF,SHIFT,OBLOC,OBTYP,FI,FO,FB,FM,OBTYPL                   INREAL.3
      REAL NU                                                           INREAL.4
C                                                                       INREAL.5
      DIMENSION U(1),V(1),W(1),P(1),CQ(1),COF(1)                        DIMEN.2
     1 ,VOL(1)                                                          DIMEN.3
      DIMENSION SPEC(10),TSP(10)                                        VF3.1
      DIMENSION TA(1),QCON(1)                                           DIMEN.6
C                                                                       DIMEN.7
      COMMON/VRCOM/A(36000)                                             VF3.2
      COMMON/CONCEN/ NSPS(10),PMAXWS(10),PMINWS(10),BKGS(10),FACS(10)   VF3.3
     1 ,IPEDIT(10),IPEL(10),IPES(10),PMIN(10)                           VF3.4
      COMMON /AMIX/ C(600)                                              COMM.5
      COMMON/FL/FO(20),FB(8),FM                                         COMM.6
      COMMON/FLS/FI(9)                                                  COMM.7
      COMMON/VRCON/ALX,ALY,ALZ,BO,BUFL,CEP,DT,DX(80),DXC,DY(80),DYC,    VF3.5
     1 DZ(80),DZC,EM0,EM6,ERF,GPX,GPY,GPZ,GX,GY,GZ,IBP1,IBP2,IBR,IBRED, VF3.6
     2 IBRIN,IBRST,IGOP(6),IJKMX,IJKNC,IJK2,IJP1K,IPL,IPR,ITER,I2JK2,   COMM.10
     * IKP2,IPARTF,IPLTF,ISELG,RIBJBKB,LPR,DROU,                        COMM.11
     *IJKCON,ICON,JCON,                                                 COMM.12
     3 I2K2,JBP1,JBP2,JBR,JBRED,JBRIN,JBRST,JNC,JPA,JPF,J2NC,KBP1,KBP2, COMM.13
     4 KBR,KBRED,KBRIN,KBRST,KFP,KNC,KPB,KPT,KWA,KWB,KWF,KWL,KWR,KWT,   COMM.14
     5 K2NC,LABEL(8),LNPART,LNVRCM,LPARTC,NCIP,NCVIP(6),NCYC,NGOP,NOPI, COMM.15
     *NOCONC,                                                           COMM.16
     * OBLOC(1200),OBTYP(1200),NOB,NU,                                  VF3.7
     6 NOVP,NOVS,NPMX,NPRT,NTPE,NWPC,NWPP,PDX,PDY,PDZ,PI,PRFLM,ROT,     COMM.18
     7 RDX(80),RDY(80),RDZ(80),RRAN(20),SIGMA,TD,TD1,TD2,TFIN,TIMFT,TIP,VF3.8
     8 TPL,TPLT,TPR,TPRT,TTO,TWIP,TWTD,UI,UPI,VDI,VDO,VI,VNTP(6),VPI,   COMM.20
     9 VTP,WI,WPI,WRAN(20),X(80),XPCEN,Y(80),YPCEN,Z(80),ZPCEN          VF3.9
      COMMON /MODSET/ SCELM(200)                                        COMM.22
      COMMON /SAELM/ XCONV,ZCONV,FX1,FX2,FX3,FX4,FY1,FY2,FY3,FY4,       COMM.23
     1 FXL,FYB,PXL,PYB,IFLB(3),FFX3(75),FFY3(75)                        COMM.24
     2 ,ITCYC(3),XFC1(4),XFC2(4),YFC1(4),YFC2(4),PBTIM(2),XML,YML,ZML   COMM.25
     3 ,IFMID,IFTEST,IPLOPT,IPM(20),IPMM,IPPP,IPROJ,JPM(20),JPMM,JPPP,  COMM.26
     4 KPM(20),KPMM,KPPP,NEC,NMC,NPS,NPV,PEC(10),PMC(10),YCONV          COMM.27
     5 ,IFLAB,IFLIN,IFPN1,IFOP1,NCH,XFSP,XT,YFSP,YT                     COMM.28
      COMMON /ZLAST/ ZLAS                                               COMM.29
C                                                                       COMM.30
      EQUIVALENCE (A(1),U),(A(2),V),(A(3),W),(A(4),P),(A(5),CQ),        EQUIV.2
     1 (A(6),COF), (A(7),VOL)                                           EQUIV.3
      EQUIVALENCE (C(1),TA,QCON)                                        EQUIV.7
C                                                                       EQUIV.8
    1 FORMAT(1H ,46H *** ENTERED SUBROUTINE VRM - MAIN PROGRAM ***)     VARP.730
    2 FORMAT(1H ,72H *** ERROR 008 - PRESSURES FAILED TO CONVERGE WITHINVARP.731
     1 1000 ITERATIONS ***)                                             VARP.732
    3 FORMAT(1H ,8H PMAX = ,1PE12.5)                                    VARP.733
    4 FORMAT(1H ,6HTIMET=,1PE12.5,2X,5HITER=,I5)                        VARP.734
    5 FORMAT(1H ,17H *** GRIND TIME =,1PE12.5,3X,7H AT T =,E12.5)       VARP.735
      WRITE(VDO,1)                                                      VARP.736
C ***** THIS INITIALIZES PMAX FOR PRESSURE ITERATION                    VARP.737
  900 PMAX=EM6                                                          VARP.738
C ***** THE FOLLOWING CONSTANTS DEFINE THE LIMITS OF THE PRINT LOOPS IN VARP.739
      IBRST=2                                                           VARP.740
      IBRED=IBP1                                                        VARP.741
      IBRIN=1                                                           VARP.742
```

```
12          KBRST=2                                                     VARP.743
13          IUMAX= 0                                                    VE3.20
14          KBRED=KBP1                                                  VARP.744
15          KBRIN=1                                                     VARP.745
16          JBRST=2                                                     VARP.746
17          JBRED=JBP1                                                  VARP.747
21          JBRIN=1                                                     VARP.748
      C ***** THIS SECTION, 1000 , CALCULATES CONSTANTS                 VARP.749
22    1000  CALL SECOND(S1)                                             VARP.750
24          ASSIGN 2000 TO KBC                                          VARP.751
25          IBCF=2000                                                   VARP.752
      C     **** ISELG IS FLAG FOR NO SLIP BOUNDRY ON FLOOR AND OBSTACLES, VARP.753
      C     ISELG=0 FREE SLIP,    ISELG=1 NO SLIP                       VARP.754
26          ITER=0                                                      VARP.755
      C ***** IF IPARTF=1, THEN HYDRO WILL BE BYPASSED.                 VARP.756
27          IPARTF= 0                                                   VE3.21
30    1040  IF( IBR.EQ.0 ) GO TO 7500                                   VARP.761
      C ***** THIS SECTION,1100 TO 1900 , CALCULATES BOUNDARY CONDITIONS VARP.762
      C ***** THIS SECTION , 1100-1289 , CAL WALL RIGHT + LEFT BNDRY COND. VARP.763
31    1100  JJ=1-I2K2                                                   VARP.764
33          DO 1289 J=1,JBP2                                            VARP.765
35          JJ=JJ+I2K2                                                  VARP.766
36          LWPC=-NWPC                                                  VARP.767
40          DO 1289 K=1,KBP2                                            VARP.768
41          LWPC=LWPC+NWPC                                              VARP.769
42          IMJK=JJ+LWPC                                                VARP.770
43          V(IMJK)=V(IMJK+K2NC)                                        VARP.771
47          W(IMJK)=W(IMJK+K2NC)                                        VARP.772
51          IPJK=IKP2+IMJK                                              VARP.773
52          V(IPJK+K2NC)=V(IPJK)                                        VARP.774
55          W(IPJK+K2NC)=W(IPJK)                                        VARP.775
60          GO TO( 1140,1150,1160 ),KWL                                 VARP.776
      C ***** THIS SECTION, 1140 , CALCULATES RIGID WALL LEFT BNDRY COND VARP.777
70    1140  U(IMJK)=0.0                                                 VARP.778
      C  ******** INFLOW BOUNDRY CONDITION                              VARP.779
71          U(IMJK)= 1.0                                                VE3.22
73          IF( ISELG.EQ.0) GO TO 1180                                  VE3.23
75          W(IMJK)= -W(IMJK+K2NC)                                      VE3.24
77          V(IMJK)= -V(IMJK+K2NC)                                      VE3.25
101         GO TO 1180                                                  VARP.781
      C ***** THIS SECTION, 1150 , CALCULATES CONTINUATIVE BNDRY COND   VARP.782
102   1150  IF( ITER.GT.0 ) GO TO 1180                                  VARP.783
105         U(IMJK)=U(IMJK+K2NC)                                        VARP.784
110         GO TO 1180                                                  VARP.785
      C ***** THIS SECTION, 1160 , CALCULATES PERIODIC WALL LEFT BNDRY COND VARP.786
110   1160  U(IMJK)=U(IPJK)                                             VARP.787
113   1180  CONTINUE                                                    VARP.788
      C ***** THIS SECTION, 1200 , CALCULATES WALL RIGHT BOUNDARY CONDITIONS VARP.789
113         GO TO( 1240,1250,1260 ),KWR                                 VARP.790
      C ***** THIS SECTION, 1240 , CALCULATES RIGID WALL RIGHT BNDRY COND VARP.791
122   1240  U(IPJK)=0.0                                                 VARP.792
124         IF( ISELG.EQ.0) GO TO 1280                                  VE3.26
126         W(IPJK+K2NC)= -W(IPJK)                                      VE3.27
130         V(IPJK+K2NC)= -V(IPJK)                                      VE3.28
133         GO TO 1280                                                  VARP.793
      C ***** THIS SECTION, 1250 , CALCULATES CONTINUATIVE WALL RIGHT BNDRY COVARP.794
```

```
133  1250  IF( ITER.GT.0 ) GO TO 1280                                  VARP.795
136        U(IPJK)=U(IPJK-K2NC)                                        VARP.796
141        GO TO 1280                                                  VARP.797
      C ***** THIS SECTION, 1260 , CALCULATES PERIODIC WALL RIGHT BNDRY COND  VARP.798
141  1260  U(IPJK+K2NC)=U(INJK+K2NC)                                   VARP.799
145  1280  CONTINUE                                                    VARP.800
145  1289  CONTINUE                                                    VARP.801
      C ***** THIS SECTION, 1300 TO 1490 , CAL WALL TOP + BOTTOM BNDRY COND  VARP.802
      C ***** THIS SECTION, 1300 , CALCULATES WALL TOP BOUNDARY CONDITIONS  VARP.803
152  1300  JJ=1-I2K2                                                   VARP.804
154        DO 1489 J=1,JBP2                                            VARP.805
156        JJ=JJ+I2K2                                                  VARP.806
157        KK=-K2NC                                                    VARP.807
161        DO 1489 I=1,IBP2                                            VARP.808
162        KK=KK+K2NC                                                  VARP.809
163        IJKM=JJ+KK                                                  VARP.810
164        U(IJKM)=U(IJKM+NWPC)                                        VARP.811
170        V(IJKM)=V(IJKM+NWPC)                                        VARP.812
172        IJKP=IJKM+KNC                                               VARP.813
173        U(IJKP+NWPC)=U(IJKP)                                        VARP.814
176        V(IJKP+NWPC)=V(IJKP)                                        VARP.815
201        GO TO( 1340,1350,1360 ),KWT                                 VARP.816
      C ***** THIS SECTION, 1340 , CALCULATES RIGID WALL TOP BNDRY COND     VARP.817
210  1340  W(IJKP)=0.0                                                 VARP.818
212        IF( ISFLG.EQ.0) GO TO 1380                                  VE3.29
214        U(IJKP+NWPC)= -U(IJKP)                                      VE3.30
216        V(IJKP+NWPC)= -V(IJKP)                                      VE3.31
221        GO TO 1380                                                  VARP.819
      C ***** THIS SECTION, 1350 , CALCULATES CONTINUATIVE WALL TOP BNDRY COND VARP.820
221  1350  IF( ITER.GT.0 ) GO TO 1380                                  VARP.821
224        W(IJKP)=W(IJKP-NWPC)                                        VARP.822
227        GO TO 1380                                                  VARP.823
      C ***** THIS SECTION, 1360 , CALCULATES PERIODIC WALL TOP BNDRY COND  VARP.824
227  1360  W(IJKP+NWPC)=W(IJKM+NWPC)                                   VARP.825
233  1380  CONTINUE                                                    VARP.826
      C ***** THIS SECTION, 1400 , CALCULATES WALL BOTTOM BOUNDARY CONDITIONS VARP.827
233  1400  GO TO( 1440,1450,1460 ),KWB                                 VARP.828
      C ***** THIS SECTION, 1440 , CALCULATES RIGID WALL BOTTOM BNDRY COND  VARP.829
242  1440  W(IJKM)=0.0                                                 VARP.830
244        IF( ISFLG.EQ.0)  GO TO 1480                                 VARP.831
246        U(IJKM)= -U(IJKM+NWPC)                                      VARP.832
250        V(IJKM)= -V(IJKM+NWPC)                                      VARP.833
252        GO TO 1480                                                  VARP.834
      C ***** THIS SECTION, 1450 , CALCULATES CONTINUATIVE WALL BOTTOM BNDRY C VARP.835
253  1450  IF( ITER.GT.0 ) GO TO 1480                                  VARP.836
256        W(IJKM)=W(IJKM+NWPC)                                        VARP.837
261        GO TO 1480                                                  VARP.838
      C ***** THIS SECTION, 1460 , CALCULATES PERIODIC WALL BOTTOM BNDRY COND VARP.839
261  1460  W(IJKM)=W(IJKP)                                             VARP.840
264  1480  CONTINUE                                                    VARP.841
264  1489  CONTINUE                                                    VARP.842
      C ***** THIS SECTION ,1500-1690 , CALCULATES FRONT + AFT BNDRY COND.  VARP.843
      C ***** THIS SECTION , 1500 , CALCULATES WALL FRONT BNDRY COND.   VARP.844
271  1500  KK=1-K2NC                                                   VARP.845
273        DO 1689 I=1,IBP2                                            VARP.846
275        KK=KK+K2NC                                                  VARP.847
```

```
276          LWPC=-NWPC                                                    VARP.848
300          DO 1689 K=1,KBP2                                              VARP.849
301          LWPC=LWPC+NWPC                                                VARP.850
302          IJMK=KK+LWPC                                                  VARP.851
303          W(IJMK)=W(IJMK+I2K2)                                          VARP.852
307          U(IJMK)=U(IJMK+I2K2)                                          VARP.853
311          IJPK=IJMK+I2JK2                                               VARP.854
312          W(IJPK+I2K2)=W(IJPK)                                          VARP.855
315          U(IJPK+I2K2)=U(IJPK)                                          VARP.856
320          GO TO( 1540,1550,1560 ),KWF                                   VARP.857
       C ***** THIS SECTION , 1540 , CALCULATES RIGID WALL FRONT BNDRY COND.  VARP.858
327    1540  V(IJPK)=0.0                                                   VARP.859
331          IF( ISFLG.EQ.0) GO TO 1580                                    VE3.32
333          U(IJPK+I2K2)= -U(IJPK)                                        VE3.33
335          W(IJPK+I2K2)= -W(IJPK)                                        VE3.34
340          GO TO 1580                                                    VARP.860
       C ***** THIS SECTION , 1550 , CALCULATES CONTINUATIVE WALL FRONT BNDRY CVARP.861
340    1550  IF( ITER.GT.0 ) GO TO 1580                                    VARP.862
343          V(IJPK)=V(IJPK-I2K2)                                          VARP.863
346          GO TO 1580                                                    VARP.864
       C ***** THIS SECTION , 1560 , CALCULATES PERIODIC WALL FRONT BNDRY COND.VARP.865
346    1560  V(IJPK+NWPC)=V(IJMK+NWPC)                                    VARP.866
352    1580  CONTINUE                                                      VARP.867
       C ***** THIS SECTION , 1600 , CALCULATES WALL AFT BNDRY COND.       VARP.868
352    1600  GO TO( 1640,1650,1660 ),KWA                                   VARP.869
       C ***** THIS SECTION , 1640 , CALCULATES RIGID WALL AFT BNDRY COND. VARP.870
361    1640  V(IJMK)=0.0                                                   VARP.871
363          IF( ISFLG.EQ.0) GO TO 1680                                    VE3.35
365          U(IJMK)= -U(IJMK+I2K2)                                        VE3.36
367          W(IJMK)= -W(IJMK+I2K2)                                        VE3.37
371          GO TO 1680                                                    VARP.872
       C ***** THIS SECTION , 1650 , CALCULATES CONTINUATIVE WALL AFT BNDRY CONVARP.873
372    1650  IF( ITER.GT.0 ) GO TO 1680                                    VARP.874
375          V(IJMK)=V(IJMK+I2K2)                                          VARP.875
400          GO TO 1680                                                    VARP.876
       C ***** THIS SECTION , 1660 , CALCULATES PERIODIC WALL AFT BNDRY COND. VARP.877
400    1660  V(IJMK)=V(IJPK)                                              VARP.878
403    1680  CONTINUE                                                      VARP.879
403    1689  CONTINUE                                                      VARP.880
       C *****  THIS SECTION, 1700, CALCULATES OBSTACLE BOUNDRY CONDITIONS VARP.881
410    1700  IF(NOB.LE.0) GO TO 1990                                       VARP.882
412          DO 1720 I=1,NOB                                               VARP.883
414          IJK=OBLOC(I)                                                  VARP.884
415           IMJK=IJK-K2NC                                                VARP.885
416           IJMK=IJK-I2K2                                                VARP.886
420           IJKM=IJK-NWPC                                                VARP.887
421          IOB= OBTYP(I).AND.FM                                          VARP.888
423          IOB=IOB+1                                                     VARP.889
425          GO TO (1713,1701,1702,1703,1704,1705,1706,1707,1708,1709,     VARP.890
           1        1710,1711,1712), IOB                                   VARP.891
445    1701  CONTINUE                                                      VARP.892
445          U(IMJK)=0.0                                                   VARP.893
447          V(IJK)=0.0                                                    VARP.894
450          V(IJMK)=0.0                                                   VARP.895
452          W(IJKM)=0.0                                                   VARP.896
453          GO TO 1720                                                    VARP.897
```

```
454  1702  CONTINUE                                          VARP.898
455        U(IJK)=0.0                                        VARP.899
456        V(IJK)=0.0                                        VARP.900
457        V(IJMK)=0.0                                       VARP.901
460        W(IJKM)=0.0                                       VARP.902
462        GO TO 1720                                        VARP.903
462  1703  CONTINUE                                          VARP.904
462        U(IJK)=0.0                                        VARP.905
464        U(IMJK)=0.0                                       VARP.906
465        V(IJMK)=0.0                                       VARP.907
467        W(IJKM)=0.0                                       VARP.908
470        GO TO 1720                                        VARP.909
471  1704  CONTINUE                                          VARP.910
472        U(IJK)=0.0                                        VARP.911
473        U(IMJK)=0.0                                       VARP.912
475        V(IJK)=0.0                                        VARP.913
475        W(IJKM)=0.0                                       VARP.914
477        GO TO 1720                                        VARP.915
477  1705  CONTINUE                                          VARP.916
500        U(IMJK)=0.0                                       VARP.917
502        V(IJK)=0.0                                        VARP.918
502        V(IJMK)=0.0                                       VARP.919
504        W(IJK)=0.0                                        VARP.920
504        GO TO 1720                                        VARP.921
505  1706  CONTINUE                                          VARP.922
506        U(IJK)=0.0                                        VARP.923
507        V(IJK)=0.0                                        VARP.924
510        V(IJMK)=0.0                                       VARP.925
511        W(IJK)=0.0                                        VARP.926
512        GO TO 1720                                        VARP.927
512  1707  CONTINUE                                          VARP.928
513        U(IJK)=0.0                                        VARP.929
514        U(IMJK)=0.0                                       VARP.930
516        V(IJMK)=0.0                                       VARP.931
517        W(IJK)=0.0                                        VARP.932
520        GO TO 1720                                        VARP.933
520  1708  CONTINUE                                          VARP.934
521        U(IJK)=0.0                                        VARP.935
522        U(IMJK)=0.0                                       VARP.936
524        V(IJK)=0.0                                        VARP.937
524        W(IJK)=0.0                                        VARP.938
525        GO TO 1720                                        VARP.939
525  1709  CONTINUE                                          VARP.940
525        U(IMJK)=0.0                                       VARP.941
527        V(IJMK)=0.0                                       VARP.942
530        W(IJK)=0.0                                        VARP.943
532        W(IJKM)=0.0                                       VARP.944
533        GO TO 1720                                        VARP.945
534  1710  CONTINUE                                          VARP.946
535        U(IMJK)=0.0                                       VARP.947
537        V(IJK)=0.0                                        VARP.948
537        W(IJK)=0.0                                        VARP.949
540        W(IJKM)=0.0                                       VARP.950
541        GO TO 1720                                        VARP.951
542  1711  CONTINUE                                          VARP.952
543        U(IJK)=0.0                                        VARP.953
```

```
544          V(IJMK)=0.0                                                    VARP.954
546          W(IJK)=0.0                                                     VARP.955
546          W(IJKM)=0.0                                                    VARP.956
550          GO TO 1720                                                     VARP.957
550    1712  CONTINUE                                                       VARP.958
551          U(IJK)=0.0                                                     VARP.959
552          V(IJK)=0.0                                                     VARP.960
553          W(IJK)=0.0                                                     VARP.961
553          W(IJKM)=0.0                                                    VARP.962
555          GO TO 1720                                                     VARP.963
555    1713  CONTINUE                                                       VARP.964
556          U(IJK)=0.0                                                     VARP.965
557          U(IMJK)=0.0                                                    VARP.966
561          V(IJK)=0.0                                                     VARP.967
561          V(IJMK)=0.0                                                    VARP.968
563          W(IJK)=0.0                                                     VARP.969
563          W(IJKM)=0.0                                                    VARP.970
565    1720  CONTINUE                                                       VARP.971
       C  *****  BYPASS NO SLIP BOUNDARIES ONLY FOR PRESSURE ITERATIONS OR  VARP.972
       C        ISELG EQUALS 0                                              VARP.973
570          IF( IBCF.EQ.4100.OR. ISFLG.EQ.0) GO TO 1770                    VARP.974
576          DO 1760 I=1,NOB                                                VARP.975
600          IJK= OBLOC(I)                                                  VARP.976
601          IMJK= IJK-K2NC                                                 VARP.977
602          IJMK= IJK-I2K2                                                 VARP.978
604          IJKM= IJK-NWPC                                                 VARP.979
605          OBTYPL=OBTYP(I)                                                VARP.980
607          IOB=OBTYPL .AND. FM                                            VARP.981
610          IOB= IOB+1                                                     VARP.982
612          IPJK= IJK+K2NC                                                 VARP.983
613          IJPK= IJK+I2K2                                                 VARP.984
614          IJKP= IJK+NWPC                                                 VARP.985
       C  *****  TEST FOR DIAGONAL OBSTACLE CELLS                           VARP.986
616          GO TO (1721,1745,1746,1747,1748,1749,1750,1751,1752,1753,1754, VARP.987
            U      1755,1756) , IOB                                         VARP.988
637    1721  KT= OBTYPL .AND. FB(1)                                         VARP.989
641          IR= OBTYPL .AND. FB(2)                                         VARP.990
642          IL= OBTYPL .AND. FB(3)                                         VARP.991
644          JA= OBTYPL .AND. FB(4)                                         VARP.992
645          JF= OBTYPL .AND. FB(5)                                         VARP.993
647          KB= OBTYPL .AND. FB(6)                                         VARP.994
650          IF(KT.EQ.0) GO TO 1724                                         VARP.995
       C ** TOP FACE EXPOSED      KT                                        VARP.996
654    1722  U(IJK)= -U(IJKP)                                               VARP.997
656          V(IJK)= -V(IJKP)                                               VARP.998
       C *** RIGHT FACE EXPOSED    IR                                       VARP.999
657    1724  IF(IR.EQ.0) GO TO 1726                                         VARP.1000
662          W(IJK)= -W(IPJK)                                               VARP.1001
664          V(IJK)= -V(IPJK)                                               VARP.1002
       C *** LEFT FACE EXPOSED    IL                                        VARP.1003
665    1726  IF(IL.EQ.0) GO TO 1728                                         VARP.1004
670          W(IJK)= -W(IMJK)                                               VARP.1005
672          V(IJK)= -V(IMJK)                                               VARP.1006
       C *** AFT FACE EXPOSED    JA                                         VARP.1007
       C    ***   BYPASS JA AND JF FOR 2-D CALCULATIONS                     VARP.1008
673    1728  IF(JBR.EQ.1) GO TO 1732                                        VES.38
```

```
675          IF(JA.EQ.0) GO TO 1730                                  VF3.39
700          U(IJK)= -U(IJMK)                                        VARP.1010
702          W(IJK)= -W(IJMK)                                        VARP.1011
       C *** FRONT FACE EXPOSED   JF                                 VARP.1012
703     1730 IF(JF.EQ.0) GO TO 1732                                  VARP.1013
706          U(IJK)= -U(IJPK)                                        VARP.1014
710          W(IJK)= -W(IJPK)                                        VARP.1015
711     1732 IF(KB.EQ.0) GO TO 1733                                  VARP.1016
714          U(IJK)= -U(IJKM)                                        VARP.1017
716          V(IJK)= -V(IJKM)                                        VARP.1018
       C *****   SET NORMAL FACE VELOCITIES TO ZERO                  VARP.1019
       C KT                                                          VARP.1020
717     1733 IF (KT.EQ.0) GO TO 1734                                 VARP.1021
720          W(IJK)=0.0                                              VARP.1022
       C IR                                                          VARP.1023
722     1734 IF (IR.EQ.0) GO TO 1736                                 VARP.1024
723          U(IJK)=0.0                                              VARP.1025
       C IL                                                          VARP.1026
725     1736 IF (IL.EQ.0) GO TO 1738                                 VARP.1027
726          U(IMJK)=0.0                                             VARP.1028
       C JA                                                          VARP.1029
730     1738 IF (JA.EQ.0) GO TO 1740                                 VARP.1030
731          V(IJMK)=0.0                                             VARP.1031
       C JF                                                          VARP.1032
733     1740 IF (JF.EQ.0) GO TO 1742                                 VARP.1033
734          V(IJK)= 0.0                                             VARP.1034
       C KB                                                          VARP.1035
736     1742 IF (KB.EQ.0) GO TO 1760                                 VARP.1036
737          W(IJKM)= 0.0                                            VARP.1037
741          GO TO 1760                                              VARP.1038
       C *****   DIAGONAL CELLS                                      VARP.1039
741     1745 U(IMJK)= 0.0                                            VARP.1040
743          W(IJKM)= 0.0                                            VARP.1041
744          GO TO 1759                                              VARP.1042
745     1746 U(IJK)= 0.0                                             VARP.1043
747          W(IJKM)= 0.0                                            VARP.1044
750          GO TO 1759                                              VARP.1045
751     1747 V(IJMK)= 0.0                                            VARP.1046
753          W(IJKM)= 0.0                                            VARP.1047
754          GO TO 1759                                              VARP.1048
755     1748 V(IJK)= 0.0                                             VARP.1049
757          W(IJKM)= 0.0                                            VARP.1050
760          GO TO 1759                                              VARP.1051
761     1749 W(IJK)= 0.0                                             VARP.1052
763          U(IMJK)= 0.0                                            VARP.1053
764          GO TO 1759                                              VARP.1054
766     1750 W(IJK)= 0.0                                             VARP.1055
767          U(IJK)= 0.0                                             VARP.1056
770          GO TO 1759                                              VARP.1057
770     1751 W(IJK)= 0.0                                             VARP.1058
772          V(IJMK)= 0.0                                            VARP.1059
773          GO TO 1759                                              VARP.1060
775     1752 W(IJK)= 0.0                                             VARP.1061
776          V(IJK)= 0.0                                             VARP.1062
777          GO TO 1759                                              VARP.1063
777     1753 U(IMJK)= 0.0                                            VARP.1064
```

```
1001          V(IJMK)= 0.0                                          VARP.1065
1002          GO TO 1759                                            VARP.1066
1003    1754  V(IJK)= 0.0                                           VARP.1067
1005          U(IMJK)= 0.0                                          VARP.1068
1006          GO TO 1759                                            VARP.1069
1007    1755  U(IJK)= 0.0                                           VARP.1070
1011          V(IJMK)= 0.0                                          VARP.1071
1012          GO TO 1759                                            VARP.1072
1014    1756  V(IJK)= 0.0                                           VARP.1073
1015          U(IJK)= 0.0                                           VARP.1074
1016          GO TO 1759                                            VARP.1075
1016    1759  CONTINUE                                              VARP.1076
1016    1760  CONTINUE                                              VARP.1077
1021    1770  CONTINUE                                              VARP.1078
1021    1990  GO TO KBC,( 2000,4100,5500 )                          VARP.1079
        C ***** THIS SECTION,2000 TO 2990 , CALCULATES TILDE VELOCITIES  VARP.1080
1024    2000  KFP=0                                                 VARP.1081
1025          IF( NCYC.LI.NPRT ) GO TO 2010                         VARP.1082
1030          GO TO 2030                                            VARP.1083
1030    2010  ASSIGN 2030 TO KPRT                                   VARP.1084
1031          IF( NPRT.LE.1 ) KFP=1                                 VARP.1085
1035    2020  GO TO 6020                                            VARP.1086
1036    2030  ASSIGN 7000 TO KPRT                                   VARP.1087
        C ***** THIS SECTION, 2100-2109 , STORES U(I-1,K) TO U( STORAGE ARRY )  VARP.1088
1037    2100  JJ=1                                                  VARP.1089
1040          KK=-K2NC                                              VARP.1090
1042          DO 2109 I=1,IBP2                                      VARP.1091
1043          KK=KK+K2NC                                            VARP.1092
1044          LWPC=-NWPC                                            VARP.1093
1046          DO 2109 K=1,KBP2                                      VARP.1094
1055          LWPC=LWPC+NWPC                                        VARP.1095
1056          IJK=JJ+KK+LWPC                                        VARP.1096
1060          IJKS=IJK+IJKNC                                        VARP.1097
1062          U(IJKS)=U(IJK)                                        VARP.1098
1063          V(IJKS)=V(IJK)                                        VARP.1099
1065          W(IJKS)=W(IJK)                                        VARP.1100
1066    2109  CONTINUE                                              VARP.1101
        C ***** NOW BEGIN MAJOR TILDE VELOCITIES DO LOOPS           VARP.1102
1073    2200  JJ=1                                                  VARP.1103
1074          DO 2889 J=2,JBP1                                      VARP.1104
1077          JJ=JJ+I2K2                                            VARP.1105
1100          KK=0                                                  VARP.1106
1101          HRDY=.5*RDY(J)                                        VARP.1107
1103          RDYA=1./(DY(J)+DY(J+1))                               VARP.1108
1106          RDYP= 1./DY(J+1)                                      VARP.1109
        C      THIS SECTION TRANSFERS LEFT COLUMN TO STORAGE ARRAY  VARP.1110
1106          LWPC=-NWPC                                            VARP.1111
1110          DO 2201 K=1,KBP2                                      VARP.1112
1120          LWPC=LWPC+NWPC                                        VARP.1113
1121          IJK=JJ+LWPC                                           VARP.1114
1123          IJKS=IJKNC+LWPC+1                                     VARP.1115
1124          U(IJKS)=U(IJK)                                        VARP.1116
1126          V(IJKS)=V(IJK)                                        VARP.1117
1130          W(IJKS)=W(IJK)                                        VARP.1118
1131    2201  CONTINUE                                              VARP.1119
        C      THIS SECTION TRANSFERS BOTTOM CELL TO STORAGE ARRAY  VARP.1120
```

```
1134          DO 2203 I=2,IBP2                                          VARP.1121
1143          KK= KK+K2NC                                               VARP.1122
1144          IJK=JJ+KK                                                 VARP.1123
1146          IJKS= IJKNC+KK+1                                          VARP.1124
1147          U(IJKS)=U(IJK)                                            VARP.1125
1151          V(IJKS)=V(IJK)                                            VARP.1126
1153          W(IJKS)=W(IJK)                                            VARP.1127
1154     2203 CONTINUE                                                  VARP.1128
1157          KK=0                                                      VARP.1129
1157     2210 DO 2889 I=2,IBP1                                          VARP.1130
1161          KK=KK+K2NC                                                VARP.1131
1163          LWPC=0                                                    VARP.1132
1164          HRDX=.5*RDX(I)                                            VARP.1133
1166          RDXA=1./(DX(I)+DX(I+1))                                   VARP.1134
1171          RDXP= 1./DX(I+1)                                          VARP.1135
1172     2220 DO 2889 K=2,KBP1                                          VARP.1136
        C ***** THE FOLLOWING CALCULATES CELL QUANTITY INDEXES          VARP.1137
1174          LWPC=LWPC+NWPC                                            VARP.1138
1176          HRDZ=.5*RDZ(K)                                            VARP.1139
1200          RDZA=1./(DZ(K)+DZ(K+1))                                   VARP.1140
1203          IJK=JJ+KK+LWPC                                            VARP.1141
1205          IPJK=IJK+K2NC                                             VARP.1142
1207          IJKS=IJKNC+KK+LWPC+1                                      VARP.1143
1211          IMJK=IJKS-K2NC                                            VARP.1144
1212          IJKP=IJK+NWPC                                             VARP.1145
1214          IJKM=IJKS-NWPC                                            VARP.1146
1216          IJPK=IJK+T2K2                                             VARP.1147
1217          IJMK=IJKS                                                 VARP.1148
1220          UR=U(IPJK)                                                VARP.1149
1222          UC=U(IJK)                                                 VARP.1150
1224          UL=U(IMJK)                                                VARP.1151
1226          VF=V(IJPK)                                                VARP.1152
1230          VC=V(IJK)                                                 VARP.1153
1232          VA=V(IJMK)                                                VARP.1154
1234          WT=W(IJKP)                                                VARP.1155
1236          WC=W(IJK)                                                 VARP.1156
1240          WB=W(IJKM)                                                VARP.1157
1242          PC=P(IJK)                                                 VARP.1158
        C ***** THIS SECTION , 2300-2370 , CALCULATES U TILDE FLUXES    VARP.1159
1254     2300 URA=.5*(UC+UR)                                            VARP.1160
1256          ULA=.5*(UL+UC)                                            VARP.1161
1261          VFA= .5*(VC + V(IPJK))                                    VARP.1162
1264          VAA= .5*(VA+V(IJMK+K2NC))                                 VARP.1163
1270          WTA= .5*(WC + W(IPJK))                                    VARP.1164
1273          WBA= .5*(WB + W(IPJK-NWPC))                               VARP.1165
1300          RDZP= 1./DZ(K+1)                                          VARP.1166
        C ***** CELL CENTERED PARTIALS                                  VARP.1167
1303          DUXR= (UR-UC) * RDXP                                      VARP.1168
1306          DUXL= (UC-UL) *RDX(I)                                     VARP.1169
1310          DVYF=(VF-VC) * RDYP                                       VARP.1170
1313          DVYA= (VC-VA) * RDY(J)                                    VARP.1171
1316          DWZT= (WT-WC) * RDZP                                      VARP.1172
1321          DWZB= (WC-WB) * RDZ(K)                                    VARP.1173
1323          DUYF= (U(IJPK)-UC) * RDYP                                 VARP.1174
1326          DUYA= (UC-U(IJMK)) * RDY(J)                               VARP.1175
1331          DUZT= (U(IJKP)-UC) * RDZP                                 VARP.1176
```

```
1335        DUZB= (UC-U(IJKM)) * RDZ(K)                                   VARP.1177
1340        DVXR= (V(IPJK)-VC) * RDXP                                     VARP.1178
1343        DVXL= (VC-V(IMJK)) * RDX(I)                                   VARP.1179
1346        DVZT= (V(IJKP)-VC) * RDZP                                     VARP.1180
1352        DVZB= (VC-V(IJKM)) * RDZ(K)                                   VARP.1181
1355        DWXR= (W(IPJK)-WC) * RDXP                                     VARP.1182
1360        DWXL= (WC-W(IMJK)) * RDX(I)                                   VARP.1183
1363        DWYF= (W(IJPK)-WC) * RDYP                                     VARP.1184
1366        DWYA= (WC-W(IJMK)) * RDY(J)                                   VARP.1185
1370        D2UX= 2.*RDXA* (DUXR-DUXL)                                    VARP.1186
1374        D2VY= 2.*RDYA* (DVYF-DVYA)                                    VARP.1187
1377        D2WZ= 2.*RDZA* (DWZT-DWZB)                                    VARP.1188
1402        D2UY= 2.*RDYA* (DUYF-DUYA)                                    VARP.1189
1405        D2UZ= 2.*RDZA* (DUZT-DUZB)                                    VARP.1190
1410        D2VX= 2.*RDXA* (DVXR-DVXL)                                    VARP.1191
1413        D2VZ= 2.*RDZA* (DVZT-DVZB)                                    VARP.1192
1416        D2WX= 2.*RDXA* (DWXR-DWXL)                                    VARP.1193
1421        D2WY= 2.*RDYA* (DWYF-DWYA)                                    VARP.1194
      C ***** TURBULENCE                                                   VARP.1195
1424        SIGNUI= SIGNUJ= SIGNUK= NU                                    VE3.40
1434        TBU= SIGNUI*D2UX + SIGNUJ*D2UY + SIGNUK*D2UZ                  VARP.1197
1441        TBV= SIGNUI*D2VX + SIGNUJ*D2VY + SIGNUK*D2VZ                  VARP.1198
1447        TBW= SIGNUI*D2WX + SIGNUJ*D2WY + SIGNUK*D2WZ                  VARP.1199
1454        FUX=HRDXA*( URA*(UC+UR) + ALX*ABS(URA)*(UC-UR)               VARP.1200
           1          - ULA*(UL+UC) - ALX*ABS(ULA)*(UL-UC) )              VARP.1201
1475        FUZ=HRDZ*( WTA*(UC+U(IJKP)) + ALZ*ABS(WTA)*(UC-U(IJKP))      VARP.1202
           1          - WBA*(U(IJKM)+UC) - ALZ*ABS(WBA)*(U(IJKM)-UC) )    VARP.1203
1515        FUY=HRDY*( VFA*(UC+U(IJPK)) + ALY*ABS(VFA)*(UC-U(IJPK))      VARP.1204
           1          - VAA*(U(IJMK)+UC) - ALY*ABS(VAA)*(U(IJMK)-UC) )    VARP.1205
      C ***** THIS SECTION , 2400-2470 , CALCULATES W TILDE FLUXES        VARP.1206
1547  2400  UTA=.5*(UC+U(IJKP))                                           VARP.1207
1552        UBA=.5*(UL + U(IMJK+NWPC))                                    VARP.1208
1556        VTA=.5*(VC+V(IJKP))                                           VARP.1209
1561        VATA=.5*(V(IJMK)+V(IJMK+NWPC))                                VARP.1210
1565        WTA=.5*(WC+WT)                                                VARP.1211
1567        WBA=.5*(WB+WC)                                                VARP.1212
1570        FWX=HRDX*( UTA*(WC+W(IJK+K2NC)) + ALX*ABS(UTA)*(WC-W(IJK+K2NC))  VARP.1213
           1          -UBA*(W(IMJK)+WC)-ALX*ABS(UBA)*(W(IMJK)-WC))        VARP.1214
1612        FWY=HRDY*( VTA*(WC+W(IJPK)) + ALY*ABS(VTA)*(WC-W(IJPK))      VARP.1215
           1          - VATA*(W(IJMK)+WC) - ALY*ABS(VATA)*(W(IJMK)-WC) )  VARP.1216
1632        FWZ=RDZA*( WTA*(WC+WT) + ALZ*ABS(WTA)*(WC-WT)                VARP.1217
           1          - WBA*(WB+WC) - ALZ*ABS(WBA)*(WB-WC) )              VARP.1218
      C ***** THIS SECTION , 2500 , CALCULATES V TILDE FLUXES.            VARP.1219
1662  2500  UFA=.5*(UC+U(IJPK))                                           VARP.1220
1664        ULFA=0.5*(UL + U(IJPK-K2NC))                                  VARP.1221
1671        VFA=.5*(VC+VF)                                                VARP.1222
1672        VAA=.5*(VA+VC)                                                VARP.1223
1674        WFA=.5*(WC+W(IJPK))                                           VARP.1224
1677        WBFA=0.5*(WB + W(IJPK-NWPC))                                  VARP.1225
1703        FVX=HRDX*( UFA*(VC+V(IPJK)) + ALX*ABS(UFA)*(VC-V(IPJK))      VARP.1226
           1          - ULFA*(V(IMJK)+VC) - ALX*ABS(ULFA)*(V(IMJK)-VC) )  VARP.1227
1723        FVY=RDYA*( VFA*(VC+VF) + ALY*ABS(VFA)*(VC-VF)                VARP.1228
           1          - VAA*(VA+VC) - ALY*ABS(VAA)*(VA-VC) )              VARP.1229
1741        FVZ=HRDZ*( WFA*(VC+V(IJKP)) + ALZ*ABS(WFA)*(VC-V(IJKP))      VARP.1230
           1          - WBFA*(V(IJKM)+VC) - ALZ*ABS(WBFA)*(V(IJKM)-VC) )  VARP.1231
      C ***** THIS SECTION , 2600 , CALCULATES FULL TILDE EQUATIONS       VARP.1232
```

```
1763  2600  U(IJK)=UC + DT*(2.*RDXA*(PC-P(IPJK)) + GX - FUX - FUY - FUZ)      VARP.1233
           1      +IBU*DT                                                    VARP.1234
1777         W(IJK)=WC + DT*(2.*RDZA*(PC-P(IJKP)) + GZ - FWX - FWY - FWZ)    VARP.1235
           1 +IBW*DT                                                         VARP.1236
2012         V(IJK)=VC + DT*(2.*RDYA*(PC-P(IJPK)) + GY - FVX - FVY - FVZ)    VARP.1237
           1 +IBV*DT                                                         VARP.1238
2026  2700  U(IJKS)=UC                                                       VARP.1239
2030         W(IJKS)=WC                                                      VARP.1240
2031         V(IJKS)=VC                                                      VARP.1241
2033  2800  CONTINUE                                                         VARP.1242
2033  2889  CONTINUE                                                         VARP.1243
2043         PMAX=EM6                                                        VARP.1244
      C ***** THIS SECTION, 4000-4600 , CALCULATES PRESSURES BY ITERATION    VARP.1245
2044  4000  IEC=0                                                            VARP.1246
2045         ASSIGN 4100 TO KBC                                              VARP.1247
2046         IBCF=4100                                                       VARP.1248
2047         GO TO 1100                                                      VARP.1249
      C ***** THIS BEGINS ACTUAL PRESSURE ITERATION AFTER RETURN FROM BNDRY CNVARP.1250
2050  4100  JJ=1                                                            VARP.1251
2051         DO 4499 J=2,JBP1                                                VARP.1252
2053         JJ=JJ+I2K2                                                      VARP.1253
2055         KK=0                                                            VARP.1254
2056         RDYC=RDY(J)                                                     VARP.1255
2057         RDYP=2./(DY(J)+DY(J+1))                                         VARP.1256
2062         RDYM=2./(DY(J-1)+DY(J))                                         VARP.1257
2065  4110  DO 4499 I=2,IBP1                                                 VARP.1258
2067         RDXC=RDX(I)                                                     VARP.1259
2072         RDXP=2./(DX(I)+DX(I+1))                                         VARP.1260
2074         RDXM=2./(DX(I-1)+DX(I))                                         VARP.1261
2077  4120  KK=KK+K2NC                                                       VARP.1262
2101         LWPC=0                                                          VARP.1263
2102  4130  DO 4499 K=2,KBP1                                                 VARP.1264
2106  4140  LWPC=LWPC+NWPC                                                   VARP.1265
2107         RDZC=RDZ(K)                                                     VARP.1266
2110         RDZP=2./(DZ(K)+DZ(K+1))                                         VARP.1267
2113         RDZM=2./(DZ(K-1)+DZ(K))                                         VARP.1268
2116         IJK=JJ+KK+LWPC                                                  VARP.1269
2120         IMJK=IJK-K2NC                                                   VARP.1270
2122         IJMK=IJK-I2K2                                                   VARP.1271
2123         IJKM=IJK-NWPC                                                   VARP.1272
2125         D=RDXC*(U(IJK)-U(IMJK)) + RDYC*(V(IJK)-V(IJMK))                 VARP.1273
           1+ RDZC*(W(IJK)-W(IJKM))                                          VARP.1274
2137         BETA=BO/(RDXC*(RDXP+RDXM) + RDYC*(RDYP+RDYM) + RDZC*(RDZP+RDZM) ) VARP.1275
2151         DTP=-BETA*D                                                     VARP.1276
2152         U(IJK)=U(IJK)+RDXP*DTP                                          VARP.1277
2156         U(IMJK)=U(IMJK)-RDXM*DTP                                        VARP.1278
2161         W(IJK)=W(IJK)+RDZP*DTP                                          VARP.1279
2164         W(IJKM)=W(IJKM)-RDZM*DTP                                        VARP.1280
2167         V(IJK)=V(IJK)+RDYP*DTP                                          VARP.1281
2172         V(IJMK)=V(IJMK)-RDYM*DTP                                        VARP.1282
2176         P(IJK)=P(IJK)+RD1*DTP                                           VARP.1283
2201         PT=P(IJK)                                                       VARP.1284
2201         PABS=ABS(PT)                                                    VARP.1285
2203         PMAX=AMAX1( PMAX,PABS )                                         VARP.1286
2207         XT=RDT*DTP/PMAX                                                 VARP.1287
2211         PCONV=ABS(XT)                                                   VARP.1288
```

```
      C ***** THIS CHECKS FOR PRESSURE CONVERGENCE - PRESSURE CONVERGENCE      VARP.1289
2212        IF( PCONV.GT.CEP ) IFC=1                                           VARP.1290
2216  4499  CONTINUE                                                           VARP.1291
2226  4500  ITER=ITER+1                                                        VARP.1292
2230        IF( ITER.LT.1001) GO TO 4510                                       VARP.1293
      C ***** *** ERROR 008 - PRESSURES FAILED TO CONVERGE WITHIN 1000 ITERATIVARP.1294
2232        WRITE(VDO,2)                                                       VARP.1295
2235        ERF=1                                                              VARP.1296
2236        WRITE(VDO,3) PMAX                                                  VARP.1297
2244        ASSIGN 99999 TO KPRT                                               VARP.1298
2245        GO TO 6020                                                         VARP.1299
      C ***** THIS CHECKS TO SEE IF ANY PRESSURES FAILED TO CONVERGED          VARP.1300
2246  4510  IF( IFC.EQ.1 ) GO TO 4000                                          VARP.1301
2250        ASSIGN 5500 TO KRC                                                 VARP.1302
2251        IBCF=5500                                                          VARP.1303
2252        TIMET= TIMET+DT                                                    VARP.1304
2254        NCYC= NCYC+1                                                       VARP.1305
2256        IF(TIMET.LT.TPLT) GO TO 1100                                       VARP.1306
2260        IPLTF=1                                                            VARP.1307
2261        CALL VRIO                                                          VARP.1308
2262        TPLT= TPLT+TPL                                                     VARP.1309
2264        GO TO 1100                                                         VARP.1310
      C ***** THIS SECTION , 5500 , CALLS THE VARIABLE REST. PARTICLE SECTION.VARP.1311
2265  5500  IF( NPMX.LT.1 ) GO TO 5900                                         VARP.1312
2270        CALL VRPS                                                          VARP.1313
2271        IF(IPARTF.LT.1) GO TO 5900                                         VARP.1314
      C  INCREMENT WHEN HYDRO IS BYPASSED                                      VARP.1315
2274        TIMET= TIMET + DT                                                  VARP.1316
2275        NCYC= NCYC+1                                                       VARP.1317
2277        IF( TIMET.LT.TPLT) GO TO 5900                                      VARP.1318
2302        IPLTF= 1                                                           VARP.1319
2303        CALL VRIO                                                          VARP.1320
2304        TPLT= TPLT+TPL                                                     VARP.1321
      C ***** THIS SECTION , 5900 , UPDATES TIME AND NUMBER OF CYCLES          VARP.1322
2306  5900  CALL SECOND(S2)                                                    VARP.1323
2310        GRIND=(S2-S1)*RTBJBKB                                              VARP.1324
2313        WRITE(6,5) GRIND,TIMET                                             VARP.1325
2323        WRITE(VDO,4) TIMET,ITER                                            VARP.1326
2333        IF( IPARTF.GT.0) GO TO 7000                                        VARP.1327
      C ***** THIS SECTION , 6000 , CALLS THE VARIABLE RESISTANCE INPUT/OUTPUTVARP.1328
2336  6000  IF( TIMET.GE.TPRT ) GO TO 6015                                     VARP.1329
2341        GO TO 7000                                                         VARP.1330
2341  6015  TPRT=TPRT+TPR                                                      VARP.1331
2343        IPLTF=0                                                            VARP.1332
2344  6020  CALL VRIO                                                          VARP.1333
2345        GO TO KPRT,( 1100,2030,4000,7000,99999 )                           VARP.1334
      C ***** THIS SECTION , 7000-7200 , HANDLES MAG TAPE WRITES               VARP.1335
2350  7000  ASSIGN 1000 TO KTD                                                 VARP.1336
2351        IF( TIMET.LT.TFIN ) GO TO 7020                                     VARP.1337
2354        ASSIGN 8000 TO KTD                                                 VARP.1338
2355        IF( TWTD.LT.1.E+5 ) GO TO 7100                                     VARP.1339
2357        RETURN                                                             VARP.1340
2360  7020  IF( TIMET.LT.TWTD ) GO TO 1000                                     VARP.1341
2363  7030  TWTD=TWTD+TTD                                                      VARP.1342
2365  7100  TD=TD+1                                                            VARP.1343
2367  7200  CALL VRTAP                                                         VARP.1344
```

```
VRM
 2370          IF( ERF.EQ.1 ) GO TO 99999                                  VARP.1345
 2372          GO TO KTD,( 1000,1100,8000 )                                VARP.1346
          C ***** THIS SECTION , 7500 , HANDLES MAG TAPE RESTART           VARP.1347
 2375     7500 ASSIGN 1000 TO KTD                                          VARP.1348
 2376          GO TO 7200                                                  VARP.1349
 2377     8000 CONTINUE                                                    VARP.1350
 2377    99999 RETURN                                                      VARP.1351
 2400          END                                                         VARP.1352


SUBPROGRAM LENGTH

03071

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1      - 002404   2      - 002413   3      - 002425   4      - 002431   5      - 002436   900    - 000006
1000   - 000023   1040   - 000031   1100   - 000032   1140   - 000070   1150   - 000103   1160   - 000111
1180   - 000114   1240   - 000123   1250   - 000134   1260   - 000142   1280   - 000146   1289   - 000146
1300   - 000153   1340   - 000211   1350   - 000222   1360   - 000230   1380   - 000234   1400   - 000234
1440   - 000243   1450   - 000254   1460   - 000262   1480   - 000265   1489   - 000265   1500   - 000272
1540   - 000330   1550   - 000341   1560   - 000347   1580   - 000353   1600   - 000353   1640   - 000362
1650   - 000373   1660   - 000401   1680   - 000404   1689   - 000404   1700   - 000411   1701   - 000446
1702   - 000455   1703   - 000463   1704   - 000472   1705   - 000500   1706   - 000506   1707   - 000513
1708   - 000521   1709   - 000526   1710   - 000535   1711   - 000543   1712   - 000551   1713   - 000556
1720   - 000566   1721   - 000640   1722   - 000653   1724   - 000660   1726   - 000666   1728   - 000674
1730   - 000704   1732   - 000712   1733   - 000720   1734   - 000723   1736   - 000726   1738   - 000731
1740   - 000734   1742   - 000737   1745   - 000742   1746   - 000746   1747   - 000752   1748   - 000756
1749   - 000762   1750   - 000766   1751   - 000771   1752   - 000775   1753   - 001000   1754   - 001004
1755   - 001010   1756   - 001014   1759   - 001017   1760   - 001017   1770   - 001022   1990   - 001022
2000   - 001025   2010   - 001031   2020   - 001036   2030   - 001037   2100   - 001040   2200   - 001074
2210   - 001160   2220   - 001173   2300   - 001245   2400   - 001537   2500   - 001651   2600   - 001763
2700   - 002026   2800   - 002034   2889   - 002034   4000   - 002045   4100   - 002051   4110   - 002066
4120   - 002100   4130   - 002103   4140   - 002105   4499   - 002217   4500   - 002227   4510   - 002247
5500   - 002266   5900   - 002307   6000   - 002337   6015   - 002342   6020   - 002345   7000   - 002351
7020   - 002361   7030   - 002364   7100   - 002366   7200   - 002370   7500   - 002376   8000   - 002400
99999  - 002400

BLOCK NAMES AND LENGTHS
VRCOM  - 106240/01  CONCEN - 000132/02  AMIX   - 001130/03  FL     - 000035/04  FLS    - 000011/05  VRCON  - 006352/06
MODSET - 000310/07  SAFLM  - 000447/08  ZLAST  - 000001/11

VARIABLE ASSIGNMENTS
A      - 000000/01  ALX    - 000000/06  ALY    - 000001/06  ALZ    - 000002/06  BETA   - 003061    BKGS   - 000056/02
BUFL   - 000004/06  B0     - 000003/06  C      - 000000/03  CEP    - 000005/06  CPRT   - 002653    CU     - 000004/01
CQF    - 000005/01  D      - 003060     D1     - 000006/06  DTP    - 003062     DUXL   - 002770    DUXR   - 002767
DUYA   - 002776     DUYF   - 002775     DUZB   - 003000     DUZT   - 002777     DVXL   - 003002    DVXR   - 003001
DVYA   - 002772     DVYF   - 002771     DVZB   - 003004     DVZT   - 003003     DWXL   - 003006    DWXR   - 003005
DWYA   - 003010     DWYF   - 003007     DWZB   - 002774     DWZT   - 002773     DX     - 000007/06  DY     - 000130/06
DZ     - 000251/06  D2UX   - 003011     D2UY   - 003014     D2UZ   - 003015     D2VX   - 003016    D2VY   - 003012
D2VZ   - 003017     D2WX   - 003020     D2WY   - 003021     D2WZ   - 003013     EM6    - 000373/06  ERF    - 000374/06
F      - 002650     FACS   - 000050/02  FB     - 000024/04  FFX3   - 000021/08  FFY3   - 000134/08  FI     - 000000/05
FM     - 000034/04  FO     - 000000/04  FUX    - 003030     FUY    - 003032     FUZ    - 003031    FVX    - 003046
FVY    - 003047     FVZ    - 003050     FWX    - 003037     FWY    - 003040     FWZ    - 003041    GRIND  - 003067
```

FILMCO

```
      SUBROUTINE FILMCO                                                 VARP.528
      INTEGER E,ERF,VDI,VDO,BUFL,PYML,PXML,CPRT,VNTP,VTP,PF,TO,TO1,TO2, INREAL.2
     1 PRFLM,CQF,SHIFT,OBLOC,OBTYP,FI,FO,FB,FM,OBTYPL                   INREAL.3
      REAL NU                                                           INREAL.4
C                                                                       INREAL.5
      DIMENSION U(1),V(1),W(1),P(1),CQ(1),CQF(1)                        DIMEN.2
     1 ,VOL(1)                                                          DIMEN.3
      DIMENSION SPEC(10),TSP(10)                                        VF3.1
      DIMENSION TA(1),QCON(1)                                           DIMEN.6
C                                                                       DIMEN.7
      COMMON/VRCOM/A(36000)                                             VF3.2
      COMMON/CONCEN/ NSPS(10),PMAXWS(10),PMINWS(10),BKGS(10),FACS(10)   VF3.3
     1 ,IPEDIT(10),IPFL(10),IPES(10),PMIN(10)                           VF3.4
      COMMON /AMIX/ C(600)                                              COMM.5
      COMMON/FL/FO(20),FB(8),FM                                         COMM.6
      COMMON/FLS/FI(9)                                                  COMM.7
      COMMON/VRCON/ALX,ALY,ALZ,BO,BUFL,CEP,DT,DX(80),DXC,DY(80),DYC,    VF3.5
     1 DZ(80),DZC,EMO,EM6,ERF,GPX,GPY,GPZ,GX,GY,GZ,IBP1,IBP2,IBR,IBRED, VF3.6
     2 IBRIN,IBRST,IGOP(6),IJKMX,IJKNC,IJK2,IJP1K,IPL,IPR,ITER,I2JK2,   COMM.10
     * IKP2,IPARIF,IPLIF,ISELG,RIBJBKB,LPR,DROU,                        COMM.11
     *IJKCON,ICON,JCON,                                                 COMM.12
     3 I2K2,JBP1,JBP2,JBR,JBRED,JBRIN,JBRST,JNC,JPA,JPF,J2NC,KBP1,KBP2, COMM.13
     4 KBR,KBRED,KBRIN,KBRST,KFP,KNC,KPB,KPT,KWA,KWB,KWF,KWL,KWR,KWT,   COMM.14
     5 K2NC,LABEL(8),LNPART,LNVRCM,LPARTC,NCTP,NCVTP(6),NCYC,NGOP,NOPI, COMM.15
     *NOCONC,                                                           COMM.16
     * OBLOC(1200),OBTYP(1200),NOB,NU,                                  VF3.7
     6 NOVP,NOVS,NPMX,NPRT,NTPE,NWPC,NWPP,PDX,PDY,PDZ,PI,PRFLM,RDT,     COMM.18
     7 RDX(80),RDY(80),RDZ(80),RRAN(20),SIGMA,TD,TD1,TD2,TFIN,TIMET,TIP,VF3.8
     8 TPL,TPLT,TPR,TPRT,TTD,TWTP,TWTD,UI,UPI,VDI,VDO,VI,VNTP(6),VPI,   COMM.20
     9 VTP,WI,WPI,WRAN(20),X(80),XPCEN,Y(80),YPCEN,Z(80),ZPCEN          VF3.9
      COMMON /NODSET/ SCFLM(200)                                        COMM.22
      COMMON /SAFLM/ XCONV,ZCONV,FX1,FX2,FX3,FX4,FY1,FY2,FY3,FY4,       COMM.23
     1 FXL,FYB,PXL,PYB,IFLB(3),FFX3(75),FFY3(75)                        COMM.24
     2 ,ITCYC(3),XFC1(4),XFC2(4),YFC1(4),YFC2(4),PBTIM(2),XML,YML,ZML   COMM.25
     3 ,IFMID,IFTEST,IPLOPT,IPM(20),IPMM,IPPP,IPROJ,JPM(20),JPMM,JPPP,  COMM.26
     4 KPM(20),KPMM,KPPP,NEC,NMC,NPS,NPV,PEC(10),PMC(10),YCONV          COMM.27
     5 ,IFLAB,IFLIN,IFPNT,IFOPT,NCH,XFSP,XT,YFSP,YT                     COMM.28
      COMMON /ZLAST/ ZLAS                                               COMM.29
C                                                                       COMM.30
      EQUIVALENCE (A(1),U),(A(2),V),(A(3),W),(A(4),P),(A(5),CQ),        EQUIV.2
     1 (A(6),CQF), (A(7),VOL)                                           EQUIV.3
      EQUIVALENCE (C(1),TA,QCON)                                        EQUIV.7
C                                                                       EQUIV.8
  100 IF( PRFLM.LT.1 ) RETURN                                           VARP.533
    4 DM1=1./AMAX1(XML,ZML)                                             VARP.534
   11 DM2=1./AMAX1(XML,YML)                                             VARP.535
   14 DM3=1./AMAX1(YML,ZML)                                             VARP.536
   17 XFSP=0.363                                                        VARP.537
   21 YFSP=0.030                                                        VARP.538
   23 I1=J1=K1=2                                                        VARP.539
   26 I2=IBP2                                                           VARP.540
   27 J2=JBP2                                                           VARP.541
   30 K2=KBP2                                                           VARP.542
   32 I3=J3=K3=1                                                        VARP.543
   35 RETURN                                                            VARP.544
C ***** THIS ENTRY , 200 , COMPUTES FILM PLANE COORDINATES,             VARP.545
```

```
36          ENTRY FPLAN                                                  VARP.546
43      200 GO TO( 220,230,240 ),IPLOPT                                  VARP.547
        C ***** THIS SECTION , 220 , COMPUTES CONSTANTS FOR PLANE I X K. VARP.548
52      220 DM=DM1                                                       VARP.549
53          XM=XML                                                       VARP.550
55          YM=ZML                                                       VARP.551
57          FXL=PXL=X(1)                                                 VARP.552
61          FYB=PYB=Z(1)                                                 VARP.553
64          GO TO 260                                                    VARP.554
        C ***** THIS SECTION , 230 , COMPUTES CONSTANTS FOR PLANE I X J. VARP.555
64      230 DM=DM2                                                       VARP.556
65          XM=XML                                                       VARP.557
67          YM=YML                                                       VARP.558
71          FXL=PXL=X(1)                                                 VARP.559
73          FYB=PYB=Y(1)                                                 VARP.560
76          GO TO 260                                                    VARP.561
        C ***** THIS SECTION , 240 , COMPUTES CONSTANTS FOR PLANE J X K. VARP.562
76      240 DM=DM3                                                       VARP.563
77          XM=YML                                                       VARP.564
101         YM=ZML                                                       VARP.565
103         FXL=PXL=Y(1)                                                 VARP.566
105         FYB=PYB=Z(1)                                                 VARP.567
111     260 XFEP=XM*DM+XFSP-0.060                                        VARP.568
114         YFEP=YM*DM+YFSP-0.060                                        VARP.569
117         XCONV=(XFEP-XFSP)/XM                                         VARP.570
121         YCONV=(YFEP-YFSP)/YM                                         VARP.571
123         XFC1(1)=XFSP                                                 VARP.572
124         XFC1(2)=XFEP                                                 VARP.573
125         XFC1(3)=XFEP                                                 VARP.574
126         XFC1(4)=XFSP                                                 VARP.575
127         YFC1(1)=YFSP                                                 VARP.576
131         YFC1(2)=YFSP                                                 VARP.577
131         YFC1(3)=YFEP                                                 VARP.578
133         YFC1(4)=YFEP                                                 VARP.579
133         XFC2(1)=XFEP                                                 VARP.580
134         XFC2(2)=XFEP                                                 VARP.581
135         XFC2(3)=XFSP                                                 VARP.582
136         XFC2(4)=XFSP                                                 VARP.583
137         YFC2(1)=YFSP                                                 VARP.584
140         YFC2(2)=YFEP                                                 VARP.585
141         YFC2(3)=YFEP                                                 VARP.586
142         YFC2(4)=YFSP                                                 VARP.587
144         RETURN                                                       VARP.588
        C ***** THIS ENTRY FCTEST TEST ON FILM COORDINATES OUT OF RANGE. VARP.589
144         ENTRY FCTEST                                                 VARP.590
151         IF( FX1.LT.XFSP ) FX1=XFSP                                   VARP.591
154         IF( FX1.GT.XFEP ) FX1=XFEP                                   VARP.592
157         IF( FY1.LT.YFSP ) FY1=YFSP                                   VARP.593
162         IF( FY1.GT.YFEP ) FY1=YFEP                                   VARP.594
165         IF( IFTEST.LT.2 ) RETURN                                     VARP.595
170         IF( FX2.LT.XFSP ) FX2=XFSP                                   VARP.596
173         IF( FX2.GT.XFEP ) FX2=XFEP                                   VARP.597
176         IF( FY2.LT.YFSP ) FY2=YFSP                                   VARP.598
201         IF( FY2.GT.YFEP ) FY2=YFEP                                   VARP.599
204         IF( IFTEST.LT.3 ) RETURN                                     VARP.600
207         IF( FX3.LT.XFSP ) FX3=XFSP                                   VARP.601
```

```
FILMCO
 212        IF( FX3.GT.XFEP ) FX3=XFEP                                      VARP.602
 215        IF( FY3.LT.YFSP ) FY3=YFSP                                      VARP.603
 220        IF( FY3.GT.YFEP ) FY3=YFEP                                      VARP.604
 223        IF( FX4.LT.XFSP ) FX4=XFSP                                      VARP.605
 226        IF( FX4.GT.XFEP ) FX4=XFEP                                      VARP.606
 231        IF( FY4.LT.YFSP ) FY4=YFSP                                      VARP.607
 234        IF( FY4.GT.YFEP ) FY4=YFEP                                      VARP.608
 237        RETURN                                                          VARP.609
      C ***** THIS ENTRY FLMID LABELS SAI,DATE, TIME+CYCLE, AND DRAWS FRAME. VARP.610
 240        ENTRY FLMID                                                     VARP.611
 245        IFOPT=1                                                         VARP.612
 246        CALL FLMOPT                                                     VARP.613
 247        IFLAB=1                                                         VARP.614
 250        CALL FLMLAB                                                     VARP.615
 251        IFOPT=2                                                         VARP.616
 252        CALL FLMOPT                                                     VARP.617
 253        XT=0.02                                                         VARP.618
 255        ENCODE(30,9000,ITCYC) TIMET,NCYC                                VARP.619
 266        NCH=30                                                          VARP.620
 267        IFLAB=3                                                         VARP.621
 270        CALL FLMLAB                                                     VARP.622
 271        IFLAB=2                                                         VARP.623
 272        IF( IFMID.GT.0 ) RETURN                                         VARP.624
 274        IFLIN=1                                                         VARP.625
 275        CALL FLMLIN                                                     VARP.626
 276        RETURN                                                          VARP.627
      C ***** FORMATS ***** FORMATS ***** FORMATS ***** FORMATS ***** FORMATS.VARP.628
      9000  FORMAT(6HTIME =,1PE11.4,8H CYCLE =,I5)                          VARP.629
 277        END                                                             VARP.630
```

SUBPROGRAM LENGTH

00403

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

100	- 000002	200	- 000044	220	- 000053	230	- 000065	240	- 000077	260	- 000111
9000	- 000310										

BLOCK NAMES AND LENGTHS

VRCOM	- 106240/01	CONCFN	- 000132/02	AMIX	- 001130/03	FL	- 000035/04	FLS	- 000011/05	VRCON	- 006352/06
MODSET	- 000310/07	SAFLM	- 000447/08	ZLAST	- 000001/11						

VARIABLE ASSIGNMENTS

A	- 000000/01	BKGS	- 000036/02	BUFL	- 000004/06	C	- 000000/03	CPRT	- 000332	CQ	- 000004/01
CQF	- 000005/01	DM	- 000376	DM1	- 000362	DM2	- 000363	DM3	- 000364	DX	- 000007/06
DY	- 000130/06	DZ	- 000251/06	ERF	- 000374/06	F	- 000327	FACS	- 000050/02	FB	- 000024/04
FCTEST	- 000326	FFX3	- 000021/08	FFY3	- 000134/08	F1	- 000000/05	FLMID	- 000326	FM	- 000034/04
FQ	- 000000/04	FPLAN	- 000326	FXL	- 000012/08	FX1	- 000002/08	FX2	- 000003/08	FX3	- 000004/08
FX4	- 000005/08	FYB	- 000013/08	FY1	- 000006/08	FY2	- 000007/08	FY3	- 000010/08	FY4	- 000011/08
IBP2	- 000404/06	IFLAB	- 000436/08	IFLB	- 000016/08	IFLIN	- 000437/08	IFMID	- 000277/08	IFOPT	- 000441/08
IFTEST	- 000300/08	IGOP	- 000411/06	IPEDIT	- 000062/02	IPEL	- 000074/02	IPES	- 000106/02	IPLOPT	- 000301/08
IPM	- 000302/08	ITCYC	- 000247/08	I1	- 000365	I2	- 000370	I3	- 000373	JBP2	- 000443/06

FLMCAL

```
      SUBROUTINE FLMCAL                                                 VARP.631
      INTEGER F,ERF,VDI,VDU,BUFL,PYML,PXML,CPRT,VNTP,VTP,PF,TD,TD1,TD2, INREAL.2
     1 PRFLM,CQF,SHIFT,OBLOC,OBTYP,FI,FO,FB,FM,OBTYPL                   INREAL.3
      REAL NU                                                           INREAL.4
C                                                                       INREAL.5
      DIMENSION U(1),V(1),W(1),P(1),CQ(1),CQF(1)                        DIMEN.2
     1 ,VOL(1)                                                          DIMEN.3
      DIMENSION SPEC(10),TSP(10)                                        VF3.1
      DIMENSION TA(1),QCON(1)                                           DIMEN.6
C                                                                       DIMEN.7
      COMMON/VRCOM/A(36000)                                             VF3.2
      COMMON/CONCEN/ NSPS(10),PMAXWS(10),PMINWS(10),BKGS(10),FACS(10)   VF3.3
     1 ,IPEDIT(10),IPEL(10),IPES(10),PMIN(10)                           VF3.4
      COMMON /AMIX/ C(600)                                              COMM.5
      COMMON/FL/FO(20),FB(8),FM                                         COMM.6
      COMMON/FLS/FI(9)                                                  COMM.7
      COMMON/VRCON/ALX,ALY,ALZ,BO,BUFL,CEP,DT,DX(80),DXC,DY(80),DYC,    VF3.5
     1 DZ(80),DZC,EMO,EM6,ERF,GPX,GPY,GPZ,GX,GY,GZ,IBP1,IBP2,IBR,IBRED, VF3.6
     2 IBRIN,IBRST,IGOP(6),IJKMX,IJKNC,IJK2,IJPIK,IPL,IPR,ITER,I2JK2,   COMM.10
     * IKP2,IPARTE,IPLIE,ISELG,RIBJBKB,LPR,DROU,                        COMM.11
     *IJKCON,ICON,JCON,                                                 COMM.12
     3 I2K2,JBP1,JBP2,JBR,JBRED,JBRIN,JBRST,JNC,JPA,JPF,J2NC,KBP1,KBP2, COMM.13
     4 KBR,KBRED,KBRIN,KBRST,KFP,KNC,KPB,KPT,KWA,KWB,KWF,KWL,KWR,KWT,   COMM.14
     5 K2NC,LAPFL(8),LNPART,LNVRCM,LPARTC,NCTP,NCVTP(6),NCYC,NGOP,NOPI, COMM.15
     *NOCONC,                                                           COMM.16
     * OBLOC(1200),OBTYP(1200),NOB,NU,                                  VF3.7
     6 NOVP,NOVS,NPMX,NPRT,NTPE,NWPC,NWPP,PDX,PDY,PDZ,PI,PRFLM,RD1,     COMM.18
     7 RDX(80),RDY(80),RDZ(80),RRAN(20),SIGMA,TD,TD1,TD2,TFIN,TIMET,TIP,VF3.8
     8 TPL,TPLT,TPR,TPRT,TTD,TWTP,TWTD,UI,UPI,VDI,VDU,VI,VNTP(6),VPI,   COMM.20
     9 VTP,WI,WPI,WRAN(20),X(80),XPCEN,Y(80),YPCEN,Z(80),ZPCEN          VF3.9
      COMMON /MODSET/ SCFLM(200)                                        COMM.22
      COMMON /SAFLM/ XCONV,ZCONV,FX1,FX2,FX3,FX4,FY1,FY2,FY3,FY4,       COMM.23
     1 FXL,FYB,PXL,PYB,IFLB(3),FFX3(75),FFY3(75)                        COMM.24
     2 ,ITCYC(3),XFC1(4),XFC2(4),YFC1(4),YFC2(4),PBTIM(2),XML,YML,ZML   COMM.25
     3 ,IFMID,IFTEST,IPLOPT,IPM(20),IPMM,IPPP,IPROJ,JPM(20),JPMM,JPPP,  COMM.26
     4 KPM(20),KPMM,KPPP,NEC,NMC,NPS,NPV,PEC(10),PMC(10),YCONV          COMM.27
     5 ,IFLAB,IFLIN,IFPNT,IFOPT,NCH,XFSP,XT,YFSP,YT                     COMM.28
      COMMON //LAST/ ZLAS                                               COMM.29
C                                                                       COMM.30
      EQUIVALENCE (A(1),U),(A(2),V),(A(3),W),(A(4),P),(A(5),CQ),        EQUIV.2
     1 (A(6),CQF), (A(7),VOL)                                           EQUIV.3
      EQUIVALENCE (C(1),TA,QCON)                                        EQUIV.7
C                                                                       EQUIV.8
      DATA ISAI/10H   SAI    /                                          VARP.636
C ***** THIS ENTRY ADVANCES TO A NEW FILM FRAME.                        VARP.637
      ENTRY FLMADV                                                      VARP.638
C ***** BKY SC 4060 RECORDS THE DISPLAY AND FILM ADVANCE.               VARP.639
  6   120 CALL PAGEG(SCFLM,0,1,1)                                       VARP.640
 11   190 RETURN                                                        VARP.641
C ***** THIS ENTRY TERMINATES THE FILM PLOT PACKAGE.                    VARP.642
 12       ENTRY FLMFIN                                                  VARP.643
C ***** BKY SC 4060 TERMINATES GRAPHIC OUTPUT.                          VARP.644
 17   220 CALL EXITG(SCFLM)                                             VARP.645
 21   290 RETURN                                                        VARP.646
C ***** THIS ENTRY INITIALIZES THE FILM PLOT PACKAGE.                   VARP.647
 22       ENTRY FLMINIT                                                 VARP.648
```

```
27          IFTAP=10                                                VARP.649
      C ***** BKY SC 4060 INITIALIZE THE GRAPHICS SYSTEM.           VARP.650
30      320 CALL MODESG(SCFLM,IFTAP)                                VARP.651
32          CALL OBJCTG(SCFLM,0.,0.,1.333,1.)                       VARP.652
36          CALL SUBJEG(SCFLM,0.,0.,1.333,1.)                       VARP.653
42      390 RETURN                                                  VARP.654
      C ***** THIS ENTRY LABELS THE FILM FRAME.                     VARP.655
43          ENTRY FLMLAB                                            VARP.656
      C ***** BKY SC 4060 CHARACTER DISPLAY.                        VARP.657
50      420 GO TO( 422,424,426 ),IFLAB                              VARP.658
57      422 YT=0.70                                                 VARP.659
60          XT=0.12                                                 VARP.660
62          CALL LEGNDG(SCFLM,XT,YT,10,ISAI)                        VARP.661
66          YT=YT-0.04                                              VARP.662
70          CALL LEGNDG(SCFLM,XT,YT,10,PBTIM(1))                    VARP.663
74          YT=0.60                                                 VARP.664
76          GO TO 490                                               VARP.665
76      424 CALL LEGNDG(SCFLM,XT,YT,NCH,IFLB)                       VARP.666
102         GO TO 490                                               VARP.667
103     426 CALL LEGNDG(SCFLM,XT,YT,NCH,ITCYC)                      VARP.668
107     490 RETURN                                                  VARP.669
      C ***** THIS ENTRY DRAWS LINE SEGMENTS.                       VARP.670
110         ENTRY FLMLIN                                            VARP.671
      C ***** BKY SC 4060 DRAWS NONCONTIGUOUS LINE SEGMENTS.        VARP.672
115     520 GO TO( 522,524,526,528,530,532 ),IFLIN                  VARP.673
127     522 CALL SEGMTG(SCFLM,4,XFC1,YFC1,XFC2,YFC2)                VARP.674
133         CALL SEGMTG(SCFLM,4,XFC1,YFC1,XFC2,YFC2)                VARP.675
137         GO TO 590                                               VARP.676
140     524 CALL SEGMTG(SCFLM,1,FX1,FY1,FX2,FY2)                    VARP.677
144         GO TO 590                                               VARP.678
145     526 CALL SEGMTG(SCFLM,1,FX2,FY2,FX3,FY3)                    VARP.679
151         GO TO 590                                               VARP.680
152     528 CALL SEGMTG(SCFLM,1,FX3,FY3,FX4,FY4)                    VARP.681
156         GO TO 590                                               VARP.682
157     530 CALL SEGMTG(SCFLM,1,FX4,FY4,FX1,FY1)                    VARP.683
163         GO TO 590                                               VARP.684
164     532 CALL SEGMTG(SCFLM,1,FX2,FY2,FX1,FY1)                    VARP.685
170     590 RETURN                                                  VARP.686
      C ***** THIS ENTRY PLOTS AN ARRAY OF POINTS.                  VARP.687
171         ENTRY FLMPNT                                            VARP.688
      C ***** BKY SC 4060 PLOTS POINTS.                             VARP.689
176     620 GO TO( 622,624 ),IFPNT                                  VARP.690
204     622 CALL POINTG(SCFLM,1,FX1,FY1)                            VARP.691
207         GO TO 690                                               VARP.692
210     624 CALL POINTG(SCFLM,1,FX2,FY2)                            VARP.693
213     690 RETURN                                                  VARP.694
      C ***** THIS ENTRY COMPUTES GRAPHICS SYSTEM OPTIONS.          VARP.695
214         ENTRY FLMOPT                                            VARP.696
      C ***** BKY SC 4060 SYSTEMS OPTIONS.                          VARP.697
221     720 GO TO( 721,722,723,724,725,726,727,728 ),IFOPT          VARP.698
      C ***** BKY SC 4060 LARGE CHARACTER SIZE.                     VARP.699
235     721 CALL SETSMG(SCFLM,45,1.5)                               VARP.700
240         GO TO 790                                               VARP.701
      C ***** BKY SC 4060 NORMAL CHARACTER SIZE.                    VARP.702
241     722 CALL SETSMG(SCFLM,45,1.0)                               VARP.703
244         GO TO 790                                               VARP.704
```

```
      C ***** BKY SC 4060 RESETS NORMAL PLOT POINT.                     VARP.705
  245   723 CALL SETSMG(SCFLM,84,0.0)                                   VARP.706
  250       GO TO 790                                                   VARP.707
      C ***** BKY SC 4060 RESETS NORMAL LINE WIDTH.                     VARP.708
  251   724 CALL SETSMG(SCFLM,30,1.0)                                   VARP.709
  254       GO TO 790                                                   VARP.710
      C ***** BKY SC 4060 RESETS NORMAL LINE DENSITY.                   VARP.711
  255   725 CALL SETSMG(SCFLM,94,0.0)                                   VARP.712
  260       GO TO 790                                                   VARP.713
      C ***** BKY SC 4060 RESETS UPPER CHARACTER CASE.                  VARP.714
  261   726 CALL SETSMG(SCFLM,55,0.0)                                   VARP.715
  264       GO TO 790                                                   VARP.716
      C ***** BKY SC 4060 RESETS L=LOW FOR PLOT POINT.                  VARP.717
  265   727 CALL SETSMG(SCFLM,84,1HL)                                   VARP.718
  270       GO TO 790                                                   VARP.719
      C ***** BKY SC 4060 RESETS H=HIGH FOR PLOT POINT.                 VARP.720
  271   728 CALL SETSMG(SCFLM,84,1HH)                                   VARP.721
  274   790 RETURN                                                      VARP.722
  275       END                                                         VARP.723
```

SUBPROGRAM LENGTH

00375

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

120	- 000007	190	- 000012	220	- 000020	290	- 000022	320	- 000031	390	- 000043
420	- 000051	422	- 000060	424	- 000077	426	- 000104	490	- 000110	520	- 000116
522	- 000130	524	- 000141	526	- 000146	528	- 000153	530	- 000160	532	- 000165
590	- 000171	620	- 000177	622	- 000205	624	- 000211	690	- 000214	720	- 000222
721	- 000236	722	- 000242	723	- 000246	724	- 000252	725	- 000256	726	- 000262
727	- 000266	728	- 000272	790	- 000275						

BLOCK NAMES AND LENGTHS

VRCOM	- 106240/01	CONCEN	- 000132/02	AMIX	- 001150/03	FL	- 000035/04	FLS	- 000011/05	VRCON	- 006352/06
MODSET	- 000310/07	SAFLM	- 000447/08	ZLAST	- 000001/11						

VARIABLE ASSIGNMENTS

A	- 000000/01	BKGS	- 000036/02	BUFL	- 000004/06	C	- 000000/03	CPRT	- 000343	CU	- 000004/01
CUF	- 000005/01	DX	- 000007/06	DY	- 000130/06	DZ	- 000251/06	FRF	- 000374/06	F	- 000340
FACS	- 000050/02	FB	- 000024/04	FFX3	- 000021/08	FFY3	- 000134/08	FI	- 000000/05	FLMADV	- 000337
FLMFIN	- 000337	FLMINIT	- 000337	FLMLAB	- 000337	FLMLIN	- 000337	FLMOPT	- 000337	FLMPNT	- 000337
FM	- 000034/04	FO	- 000000/04	FX1	- 000002/08	FX2	- 000003/08	FX3	- 000004/08	FX4	- 000005/08
FY1	- 000006/08	FY2	- 000007/08	FY3	- 000010/08	FY4	- 000011/08	IFLAB	- 000436/08	IFLB	- 000016/08
IFLIN	- 000437/08	IFOPT	- 000441/08	IFPNT	- 000440/08	IFTAP	- 000374	IGOP	- 000411/06	IPEDIT	- 000062/02
IPEL	- 000074/02	IPES	- 000106/02	IPM	- 000302/08	ISAI	- 000373	ITCYC	- 000247/08	JPM	- 000331/08
KPM	- 000357/08	LABEL	- 000475/06	NCH	- 000442/08	NCVTP	- 000511/06	NSPS	- 000000/02	NU	- 005264/06
OBLOC	- 000523/06	OBTYP	- 003003/06	OBTYPL	- 000346	P	- 000003/01	PBTIM	- 000272/08	PEC	- 000411/08
PF	- 000344	PMAXWS	- 000012/02	PMC	- 000423/08	PMIN	- 000120/02	PMINWS	- 000024/02	PRFLM	- 005300/06
PXML	- 000342	PYML	- 000341	QCON	- 000000/03	RDX	- 005302/06	RDY	- 005422/06	RDZ	- 005542/06
RRAN	- 005662/06	SCFLM	- 000000/07	SHIFT	- 000345	SPEC	- 000347	TA	- 000000/03	TD	- 005707/06
TD1	- 005710/06	TD2	- 005711/06	TSP	- 000361	U	- 000000/01	V	- 000001/01	VDT	- 005726/06
VDO	- 005727/06	VNTP	- 005731/06	VOL	- 000006/01	VTP	- 005740/06	W	- 000002/01	WRAN	- 005743/06

VRPS

```
      SUBROUTINE VRPS                                                   VARP.1353
      LARGE XPC(1),YPC(1),ZPC(1), PM(1),ABC(29996),CHN(15000)           VF3.43
      INTEGER F,ERF,VDI,VDU,BUFL,PYML,PXML,CPRT,VNTP,VIP,PF,TD,TD1,TD2, INREAL.2
     1 PRFLM,CQF,SHIFT,OBLOC,OBTYP,FI,FO,FB,FM,OBTYPL                   INREAL.3
      REAL NU                                                           INREAL.4
C                                                                       INREAL.5
      DIMENSION U(1),V(1),W(1),P(1),CQ(1),CQF(1)                        DIMEN.2
     1 ,VOL(1)                                                          DIMEN.3
      DIMENSION SPEC(10),TSP(10)                                        VF3.1
      DIMENSION TA(1),QCON(1)                                           DIMEN.6
C                                                                       DIMEN.7
      COMMON/VRCOM/A(36000)                                             VF3.2
      COMMON/CONCEN/ NSPS(10),PMAXWS(10),PMINWS(10),BKGS(10),FACS(10)   VF3.3
     1 ,IPEDIT(10),IPEL(10),IPES(10),PMIN(10)                           VF3.4
      COMMON /ABIX/ C(600)                                              COMM.5
      COMMON/FL/FO(20),FB(8),FM                                         COMM.6
      COMMON/FLS/FI(9)                                                  COMM.7
      COMMON/VRCON/ALX,ALY,ALZ,B0,BUFL,CEP,DT,DX(80),DXC,DY(80),DYC,    VF3.5
     1 DZ(80),DZC,EM0,EM6,ERF,GPX,GPY,GPZ,GX,GY,GZ,IBP1,IBP2,IBR,IBRED, VF3.6
     2 IBRIN,IBRST,IGOP(6),IJKMX,IJKNC,IJK2,IJP1K,IPL,IPR,ITER,I2JK2,   COMM.10
     * IKP2,IPARTF,IPLTF,ISFLG,RIBJAKB,LPR,DROU,                         COMM.11
     *IJKCON,ICON,JCON,                                                 COMM.12
     3 I2K2,JBP1,JBP2,JBR,JBRED,JBRIN,JBRST,JNC,JPA,JPF,J2NC,KBP1,KBP2, COMM.13
     4 KBR,KBRED,KBRIN,KBRST,KFP,KNC,KPB,KPI,KWA,KWB,KWF,KWL,KWR,KWT,   COMM.14
     5 K2NC,LABEL(8),LNPART,LNVRCM,LPARTC,NCTP,NCVTP(6),NCYC,NGOP,NOPI, COMM.15
     *NOCONC,                                                           COMM.16
     * OBLOC(1200),OBTYP(1200),NOB,NU,                                  VF3.7
     6 NOVP,NOVS,NPMX,NPRI,NTPE,NWPC,NWPP,PDX,PDY,PDZ,PI,PRFLM,RDT,     COMM.18
     7 RDX(80),RDY(80),RDZ(80),RRAN(20),SIGMA,TD,TD1,TD2,IFIN,TIMET,TIP,VF3.8
     8 TPL,TPLT,TPR,TPRT,TTD,TWTP,TWTD,UI,UPI,VDI,VDU,VI,VNTP(6),VPI,   COMM.20
     9 VTP,WI,WPI,WRAN(20),X(80),XPCEN,Y(80),YPCEN,Z(80),ZPCEN          VF3.9
      COMMON /MODSET/ SCFLM(200)                                        COMM.22
      COMMON /SAFLN/ XCONV,ZCONV,FX1,FX2,FX3,FX4,FY1,FY2,FY3,FY4,       COMM.23
     1 FXL,FYB,PXL,PYB,IFLR(3),FFX3(75),FFY3(75)                        COMM.24
     2 ,IICYC(3),XFC1(4),XFC2(4),YFC1(4),YFC2(4),PBTIN(2),XML,YML,ZML   COMM.25
     3 ,IFMID,IFTEST,IPLOPT,IPM(20),IPMM,IPPP,IPROJ,JPM(20),JPMM,JPPP,  COMM.26
     4 KPM(20),KPMM,KPPP,NFC,NMC,NPS,NPV,PEC(10),PMC(10),YCONV          COMM.27
     5 ,IFLAB,IFLIN,IFPNT,IFOPT,NCH,XFSP,XT,YFSP,YT                     COMM.28
      COMMON /ZLAST/ ZLAS                                               COMM.29
C                                                                       COMM.30
      EQUIVALENCE (A(1),U),(A(2),V),(A(3),W),(A(4),P),(A(5),CQ),        EQUIV.2
     1 (A(6),CQF), (A(7),VOL)                                           EQUIV.3
      EQUIVALENCE (C(1),TA,QCON)                                        EQUIV.7
C                                                                       EQUIV.8
    1 FORMAT(1H ,68H *** ERROR 006 - ATTEMPTED TO INSERT MORE PARTICLES VARP.1359
     1THAN ALLOWED ***)                                                 VARP.1360
    2 FORMAT(1H ,37H TOTAL NUMBER OF PARTICLES IN SYSTEM ,I8,4X,        VARP.1361
     17H TIME =,1PE12.5)                                                VARP.1362
    3 FORMAT(6(I4),3(F8.3),I4)                                          VARP.1363
    4 FORMAT(8(F8.3))                                                   VARP.1364
    5 FORMAT(1H ,4X,4HIPL=,I4/5X,4HIPR=,I4/5X,4HJPA=,I4/5X,4HJPF=,I4/   VARP.1365
     1 5X,4HKPI=,I4/5X,4HKPB=,I4/3X,6HXPCEN=,1PE12.5/3X,6HYPCEN=,E12.5/ VARP.1366
     2 3X,6HZPCEN=,E12.5/4X,5HNOPI=,I4)                                 VARP.1367
    6 FORMAT(1H ,4X,4HPDX=,1PE12.5/5X,4HPDY=,E12.5/5X,4HPDZ=,E12.5/     VARP.1368
     1 4X,5HTWIP=,E12.5/5X,4HUPI=,E12.5/5X,4HVPI=,E12.5/5X,4HWPI=,E12.5/VARP.1369
     2 3X,6HSIGMA=,E12.5)                                               VARP.1370
```

```
                7 FORMAT(1H ,2X,8H TIME = ,1PE12.5,8H CYCLE =,I6/3X,1HL,11X,2HXC,   VARP.1371
                1 15X,2HYC,15X,2HZC,15X,2HPM )                                      VARP.1372
                8 FORMAT(1H ,1X,I5,6(5X,1PE12.5))                                   VARP.1373
                9 FORMAT(1H ,28H TIME IN PARTICLE SECTION = ,1PE12.5)               VARP.1374
              100 NPF=0                                                             VARP.1375
        2         CALL SECOND(TP1)                                                  VARP.1376
        4         PRINT 13                                                          VARP.1377
               13 FORMAT( 1H ,*ENTERED PARTICLE SECTION*)                           VARP.1378
       10         IF( IPFLG.GT.0.OR. RRAN(20).GT.0.0) GO TO 210                     VARP.1379
      C ***** THIS SECTION , 200-299 , INITIALIZES PARTICLES AT T=0.0               VARP.1380
       20      200 READ(VDI,3) IPL,IPR,JPA,JPF,KPT,KPB,XPCEN,YPCEN,ZPCEN,NOPI       VARP.1381
       50         IF( IPL.EQ.0 ) GO TO 9500                                         VARP.1382
       51         READ(VDI,4) PDX,PDY,PDZ,TWIP,UPI,VPI,WPI,SIGMA                    VARP.1383
       75         WRITE(VDO,5) IPL,IPR,JPA,JPF,KPT,KPB,XPCEN,YPCEN,ZPCEN,NOPI       VARP.1384
      125         WRITE(VDO,6) PDX,PDY,PDZ,TWIP,UPI,VPI,WPI,SIGMA                   VARP.1385
      C        SINGLE SPECIES RIGHT NOW                                             VARP.1386
      151         NSPES=1                                                           VE3.44
      152         WAIR= 1.2                                                         VE3.45
      153         FACS(1)= 29./46.*1.E3/WAIR                                        VE3.46
      155         FACS(2)= 29./30.*1.E3/WAIR                                        VE3.47
      156         FACS(3)= 29./48.*1.E3/WAIR                                        VE3.48
      C           THREE SPECIES PLUME DATA                                          VE3.49
      160         PMAXWS(1)= 2.0                                                    VE3.50
      161         PMAXWS(2)= 2.E-3                                                  VE3.51
      162         PMAXWS(3)= 5.E-4                                                  VE3.52
      164         PMIN(1)=1.0                                                       VE3.53
      165         PMIN(2)=.11E-2                                                    VE3.54
      167         PMIN(3)=-1.E-8                                                    VE3.55
      170         BKGS(1)=0.0                                                       VE3.56
      171         BKGS(2)=1.E-4                                                     VE3.57
      172         BKGS(3)=.5                                                        VE3.58
      174         TEM6=EM6+EM6                                                      VE3.59
      175         NOCONC=1                                                          VARP.1393
      176         IJKCON= IJK2 *NSPES                                               VARP.1394
      201         JCON= IBP2*KBP2                                                   VARP.1395
      203         ICON= KBP2                                                        VARP.1396
      204         NREAL= IJK2-JCON                                                  VARP.1397
      205         IPFLG=0                                                           VARP.1398
      206         WSUM=-0.1                                                         VARP.1399
      210         DO 209 L=1,20                                                     VARP.1400
      216         WSUM=WSUM+0.1                                                     VARP.1401
      217     209 WRAN(L)=WSUM                                                      VARP.1402
      220         RRAN(1)=0.00000                                                   VARP.1403
      221         RRAN(2)=0.11246                                                   VARP.1404
      222         RRAN(3)=0.22270                                                   VARP.1405
      224         RRAN(4)=0.32863                                                   VARP.1406
      225         RRAN(5)=0.42939                                                   VARP.1407
      227         RRAN(6)=0.52050                                                   VARP.1408
      230         RRAN(7)=0.60386                                                   VARP.1409
      232         RRAN(8)=0.67780                                                   VARP.1410
      233         RRAN(9)=0.74219                                                   VARP.1411
      235         RRAN(10)=0.79691                                                  VARP.1412
      236         RRAN(11)=0.84270                                                  VARP.1413
      240         RRAN(12)=0.88021                                                  VARP.1414
      241         RRAN(13)=0.91031                                                  VARP.1415
      243         RRAN(14)=0.93401                                                  VARP.1416
```

```
244          RRAN(15)=0.95229                                          VARP.1417
246          RRAN(16)=0.96611                                          VARP.1418
247          RRAN(17)=0.97635                                          VARP.1419
251          RRAN(18)=0.98379                                          VARP.1420
252          RRAN(19)=0.98909                                          VARP.1421
254          RRAN(20)=0.99279                                          VARP.1422
256      210 IF( TIMEI.LT.TWIP ) RETURN                                VARP.1423
261          IF( IPFLG.GT.0 ) GO TO 500                                VARP.1424
264          TIP=TWIP                                                  VARP.1425
264          IF( NOPT.GT.NPMX ) NOPI=NPMX                              VARP.1426
270          LC=-NPP+1                                                 VE3.60
272          NPS=NOPI                                                  VARP.1428
273      230 DO 299 L=1,NPS                                            VARP.1429
275          LC=LC+NPP                                                 VE3.61
       C ***** ASSIGN PARTICLES EITHER TO A POINT OR AREA              VARP.1432
276          IF(XPCEN.GT.0.0) GO TO 250                                VARP.1433
301          XPC(LC)= FLOAT(IPL) + FLOAT(IPR-IPL+1) * RANF(DUM)        VARP.1434
310          YPC(LC)= FLOAT(JPA) + FLOAT(JPF-JPA+1) * RANF(DUM)        VARP.1435
320          ZPC(LC)= FLOAT(KPB) + FLOAT(KPT-KPB+1) * RANF(DUM)        VARP.1436
330          GO TO 298                                                 VARP.1437
331      250 CONTINUE                                                  VARP.1438
332          XPC(LC)=XPCEN                                             VARP.1439
333          YPC(LC)=YPCEN                                             VARP.1440
335          ZPC(LC)=ZPCEN                                             VARP.1441
340      298 CONTINUE                                                  VARP.1442
       C *****  ASSIGN UNIT MASS TO PARTICLES                           VARP.1443
340          DO 297 N=1,NSPES                                          VE3.62
345          LLC=LC+N-1                                                VE3.63
347          PM(LLC)=PMIN(N)                                           VE3.64
350      297 CONTINUE                                                  VE3.65
352      299 CONTINUE                                                  VARP.1445
354          IPFLG=1                                                   VARP.1447
355          GO TO 9500                                                VARP.1448
       C ***** THIS SECTION , 500 , PREASSIGNS LOGIC FLOW TYPES OF PART EQUATIOVARP.1449
356      500 GO TO( 510,520                                  ),NTPE    VARP.1450
       C ***** THIS SECTION , 510 , PREASSIGNS FLOW FOR MARKER PARTICLE EQUATIOVARP.1451
364      510 ASSIGN 2000 TO KPE                                        VARP.1452
365          ASSIGN 8000 TO KMPE                                       VARP.1453
       C *****  SECOND ORDER PARTICLE MOVEMENT ANYONE<, SET KMPO TO 2400 VARP.1454
366          ASSIGN 2400 TO KMPO                                       VE3.66
367          GO TO 600                                                 VARP.1456
       C ***** THIS SECTION , 520 , PREASSIGNS FLOW FOR TURBULENT DISPERSION EQVARP.1457
370      520 ASSIGN 2000 TO KPE                                        VARP.1458
371          ASSIGN 3000 TO KMPE                                       VARP.1459
372          ASSIGN 8000 TO KIDE                                       VARP.1460
373          ASSIGN 2400 TO KMPO                                       VE3.67
       C ***** THIS SECTION,600, ZEROS THE CONCENTRATION ARRAY, CONC.  VARP.1462
374      600 IF( NOCONC.LT.1) GO TO 1000                               VARP.1463
377          DO 610 I=1,IJKCON                                         VARP.1464
403      610 CON(I)= 0.0                                               VARP.1465
405          GO TO 1000                                                VARP.1466
       C ***** BEGIN MAJOR LOOP FOR CALCULATION OF PARTICLE TRANSPORT MODEL VARP.1467
405     1000 CONTINUE                                                  VE3.68
405          NPF=0                                                     VARP.1472
406          IF(NPS.LE.0) GO TO 9000                                   VE3.69
410          NPL=0                                                     VE3.70
```

```
411          LF=1-NWPP+NPL                                          VARP.1473
413          LC=1-NWPP+NPL                                          VARP.1474
       C ***** THIS SECTION , 1050 , PREASSIGNS DRG(),PM() TO CONSTANTS  VARP.1475
414  1050    DRGC=0.0                                               VARP.1476
415          PMK= 0.0                                               VARP.1477
416          DDPM=0.0                                               VARP.1478
416          DDT= DT                                                VF3.71
420  1100    DO 8999 L=1,NPS                                        VARP.1479
422          DT= DDT                                                VF3.72
423          IF( L.LE. NPS-NDPI) GO TO 1120                         VF3.73
427          DT= DDT* (NPS-L) /NDPI                                 VF3.74
433  1120    LC=LC+NWPP                                             VARP.1480
435  1200    GO TO KPE,( 2000,3000                          )       VARP.1481
       C ***** THIS SECTION , 2000-2600 , CALCULATES SIMPLE MARKER PARTICLES  VARP.1482
440  2000    CONTINUE                                               VARP.1483
444          XC=XPC(LC)                                             VARP.1484
445          IX=IFIX(XC+EM6)                                        VARP.1485
446          YC=YPC(LC)                                             VARP.1486
450          IY=IFIX(YC+EM6)                                        VARP.1487
451          ZC=ZPC(LC)                                             VARP.1488
452          IZ=IFIX(ZC+EM6)                                        VARP.1489
454          XW=XC-FLOAT(IX)                                        VARP.1490
456          YL=YC-FLOAT(IY)                                        VARP.1491
460          ZH=ZC-FLOAT(IZ)                                        VARP.1492
463          IX1=IX+SIGN(1.,XW-.5)                                  VARP.1493
470          XW1=ABS(XW-.5)                                         VARP.1494
472          IY1=IY+SIGN(1.,YL-.5)                                  VARP.1495
477          YL1=ABS(YL-.5)                                         VARP.1496
501          IZ1=IZ+SIGN(1.,ZH-.5)                                  VARP.1497
507          ZH1=ABS(ZH-.5)                                         VARP.1498
511          I=(IX-1)*K2NC                                          VARP.1499
514          J=(IY-1)*I2K2                                          VARP.1500
516          K=(IZ-1)*NWPC+1                                        VARP.1501
522          I1=(IX1-1)*K2NC                                        VARP.1502
524          J1=(IY1-1)*I2K2                                        VARP.1503
530          K1=(IZ1-1)*NWPC+1                                      VARP.1504
533          IM=I-K2NC                                              VARP.1505
535          JM=J-I2K2                                              VARP.1506
537          KM=K-NWPC                                              VARP.1507
541          IJK=I+J+K                                              VARP.1508
543          XW1M=1.-XW1                                            VARP.1509
545          YL1M=1.-YL1                                            VARP.1510
546          ZH1M=1.-ZH1                                            VARP.1511
       C ***** THIS SECTION , 2100 , CALCULATES U-FLUID VELOCITY   VARP.1512
550  2100    UR=(YL1M*(ZH1*U(I+J+K1)+ZH1M*U(IJK))                  VARP.1513
            *    +YL1*(ZH1*U(I+J1+K1)+ZH1M*U(I+J1+K)))             VARP.1514
567          UL=(YL1M*(ZH1*U(IM+J+K1)+ZH1M*U(IM+J+K))               VARP.1515
            *    +YL1*(ZH1*U(IM+J1+K1)+ZH1M*U(IM+J1+K)))           VARP.1516
       C ***** THIS SECTION , 2200 , CALCULATES W-FLUID VELOCITY   VARP.1517
607  2200    WT=(XW1M*(YL1*W(I+J1+K)+YL1M*W(IJK))                  VARP.1518
            *    +XW1*(YL1*W(I1+J1+K)+YL1M*W(I1+J+K)))             VARP.1519
626          WB= (XW1M* (YL1*W(I+J1+KM)+ YL1M*W(I+J+KM))           VARP.1520
            *    +XW1*(YL1*W(I1+J1+KM)+YL1M*W(I1+J+KM)))           VARP.1521
       C ***** THIS SECTION , 2300 , CALCULATES V-FLUID VELOCITY.  VARP.1522
650  2300    VF=(ZH1M*(XW1*V(I1+J+K)+XW1M*V(IJK))                  VARP.1523
            *    +ZH1*(XW1*V(I1+J+K1)+XW1M*V(I+J+K1)))             VARP.1524
```

```
667          VA=(ZH1M*(XW1*V(I1+JM+K)+XW1M*V(I+JM+K))                    VARP.1525
            *   +ZH1*(XW1*V(I1+JM+K1)+XW1M*V(I+JM+K1)))                  VARP.1526
       C *****  THIS SECTION COMPUTES FIRST ORDER PARTICLE ADVECTION     VARP.1527
707          UFL=X**UR + (1.0-XW)*UL                                     VARP.1528
714          WFL=ZH*WT + (1.0-ZH)*WB                                     VARP.1529
720          VFL=YL*VF + (1.0-YL)*VA                                     VARP.1530
725          GO TO KMPD, (2400,2600      )                               VARP.1531
       C ****  THIS SECTION, 2600, COMPUTES SECOND ORDER PARTICLE ADVECTION  VARP.1532
730  2400    UFL=UFL*(1.0+0.5*DT*RDX(IX)*(UR-UL))                        VARP.1533
736          WFL=WFL*(1.0+0.5*DT*RDZ(IZ)*(WT-WB))                        VARP.1534
745          VFL=VFL*(1.0+0.5*DT*RDY(IY)*(VF-VA))                        VARP.1535
       C ***** THIS SECTION , 2600 , COLLECTS TRANSPORT TERMS            VARP.1536
753  2600    UPAR=UFL                                                    VARP.1537
754          WPAR=WFL                                                    VARP.1538
756          VPAR=VFL                                                    VARP.1539
760          GO TO KMPF,( 3000,8000                          )           VARP.1540
       C ***** THIS SECTION , 3000-3600 , CALCULATES TURBULENT DISPERSION VARP.1541
763  3000    CONTINUE                                                    VARP.1542
763          DO 3099 M=1,3                                               VARP.1543
765          IF(M-2) 3001,3002,3003                                      VE3.75
767  3001    SIGMA=10.0                                                  VE3.76
771          GO TO 3010                                                  VE3.77
771  3002    SIGMA=10.0                                                  VE3.78
773          GO TO 3010                                                  VE3.79
773  3003    SIGMA=5.0                                                   VE3.80
775  3010    ZETA=RANF(DUM)                                              VARP.1544
1000         ZETSIN=1.-2.*ZETA                                           VARP.1545
1003         ZETA=ABS(ZETSIN)                                            VARP.1546
1004         ZETSIN=SIGN(1.,ZETSIN)                                      VARP.1547
1006         LT=10                                                       VARP.1548
1007         LTM=5                                                       VARP.1549
1010         LTP=15                                                      VARP.1550
1011         LTMX=20                                                     VARP.1551
1012         IF( ZETA.GT.RRAN(LT) ) GO TO 3020                           VARP.1552
1016         IF( ZETA.GT.RRAN(LTM) ) GO TO 3015                          VARP.1553
1021         L1=LTM                                                      VARP.1554
1021  3012   IF( ZETA.GE.RRAN(L1) ) GO TO 3030                           VARP.1555
1024         L1=L1-1                                                     VARP.1556
1025         GO TO 3012                                                  VARP.1557
1026  3015   L1=LT                                                       VARP.1558
1030  3017   IF( ZETA.GE.RRAN(L1) ) GO TO 3030                           VARP.1559
1033         L1=L1-1                                                     VARP.1560
1034         GO TO 3017                                                  VARP.1561
1035  3020   IF( ZETA.GT.RRAN(LTP) ) GO TO 3025                          VARP.1562
1041         L1=LTP                                                      VARP.1563
1041  3022   IF( ZETA.GE.RRAN(L1) ) GO TO 3030                           VARP.1564
1044         L1=L1-1                                                     VARP.1565
1045         GO TO 3022                                                  VARP.1566
1046  3025   L1=LTMX                                                     VARP.1567
1050  3027   IF( ZETA.GE.RRAN(L1) ) GO TO 3030                           VARP.1568
1053         L1=L1-1                                                     VARP.1569
1054         GO TO 3027                                                  VARP.1570
1055  3030   IF( L1.EQ.LTMX .AND. ZETA.GT.RRAN(L1) ) GO TO 3010          VARP.1571
1065         W1=WRAN(L1) + (WRAN(L1+1)-WRAN(L1))*(ZETA-RRAN(L1))         VARP.1572
            1 /(RRAN(L1+1)-RRAN(L1))                                     VARP.1573
1074         W2=SQRT(4.*SIGMA*RDT)*W1*ZETSIN                             VARP.1574
```

```
1103          GO TO(3040,3050,3060),M                                  VARP.1575
1112   3040   URAND=K2                                                 VARP.1576
1114          GO TO 3099                                               VARP.1577
1114   3050   VRAND=K2                                                 VARP.1578
1116          GO TO 3099                                               VARP.1579
1116   3060   WRAND=K2                                                 VARP.1580
1120   3099   CONTINUE                                                 VARP.1581
1122          UPAR=UFL+URAND                                           VARP.1582
1124          VPAR=VFL+VRAND                                           VARP.1583
1126          WPAR=WFL+WRAND                                           VARP.1586
1130          GO TO KIDE,( 8000                                  )     VARP.1587
        C ***** THIS SECTION , 8000 , CALCULATES THE TRANSPORT CONTRIBUTION VARP.1588
        C     REFLECTIVE BOUNDARIES FOR OBSTACLES                      VARP.1589
1133   8000   XWN=XW+UPAR*DT*RDX(IX)                                   VARP.1590
1137          XC=FLOAT(IX)+XWN                                         VARP.1591
1141          IF(XWN.GT.0.0) GO TO 8010                                VARP.1592
1143          IF(U(IJK-K2NC).EQ.0) GO TO 8005                          VARP.1593
1145          XC=FLOAT(IX)+XWN*DX(IX)*RDX(IX-1)                        VARP.1594
1151          GO TO 8020                                               VARP.1595
1151   8005   XC=FLOAT(IX)-XWN                                         VARP.1596
1154          GO TO 8020                                               VARP.1597
1154   8010   IF(XWN.LT.1.0) GO TO 8020                                VARP.1598
1157          IF(U(IJK).EQ.0) GO TO 8015                               VARP.1599
1160          XC=FLOAT(IX)+1.0+(XWN-1.)*DX(IX)*RDX(IX+1)               VARP.1600
1165          GO TO 8020                                               VARP.1601
1165   8015   XC=FLOAT(IX)+2.0-XWN                                     VARP.1602
1171   8020   CONTINUE                                                 VARP.1603
1171          YLN=YL + VPAR*DT*RDY(IY)                                 VARP.1604
1175          YC=FLOAT(IY) + YLN                                       VARP.1605
1177          IF(YLN.GT.0) GO TO 8030                                  VARP.1606
1201          IF(V(IJK -I2K2).EQ.0) GO TO 8025                         VARP.1607
1203          YC= FLOAT(IY)+YLN * DY(IY)*RDY(IY-1)                     VARP.1608
1207          GO TO 8040                                               VARP.1609
1207   8025   YC= FLOAT(IY)-YLN                                        VARP.1610
1212          GO TO 8040                                               VARP.1611
1212   8030   IF(YLN.LT.1) GO TO 8040                                  VARP.1612
1215          IF(V(IJK).EQ.0) GO TO 8035                               VARP.1613
1217          YC= FLOAT(IY)+1.0+ (YLN-1.) *DY(IY)*RDY(IY+1)            VARP.1614
1224          GO TO 8040                                               VARP.1615
1225   8035   YC= FLOAT(IY)+2.0-YLN                                    VARP.1616
1231   8040   CONTINUE                                                 VARP.1617
1231          ZHN=ZH + WPAR*DT*RDZ(IZ)                                 VARP.1618
1235          ZC=FLOAT(IZ) + ZHN                                       VARP.1619
1237          IF( ZHN.GT.0) GO TO 8050                                 VARP.1620
1241          IF( W(IJK-NWPC). EQ. 0) GO TO 8045                       VARP.1621
1243          ZC= FLOAT(IZ)+ZHN * DZ(IZ)*RDZ(IZ-1)                     VARP.1622
1247          GO TO 8060                                               VARP.1623
1247   8045   ZC= FLOAT(IZ)- ZHN                                       VARP.1624
1252          GO TO 8060                                               VARP.1625
1252   8050   IF( ZHN.LT.1) GO TO 8060                                 VARP.1626
1255          IF( W(IJK).EQ.0) GO TO 8055                              VARP.1627
1257          ZC= FLOAT(IZ)+1.0 + (ZHN-1.0) *DZ(IZ)*RDZ(IZ+1)          VARP.1628
1264          GO TO 8060                                               VARP.1629
1265   8055   ZC= FLOAT(IZ)+2.0-ZHN                                    VARP.1630
1271   8060   CONTINUE                                                 VARP.1631
1271          IX= IFIX(XC)                                             VF3.81
```

```
1272          IY= IFIX(YC)                                          VE3.82
1274          IZ= IFIX(ZC)                                          VE3.83
1275          IF(IX.LT.2.OR.IX.GT.IBP1) GO TO 8990                  VARP.1632
1305          IF(IY.LT.2.OR.IY.GT.JBP1) GO TO 8990                  VARP.1633
1314          IF(IZ.GT.KBP1) GO TO 8990                             VARP.1634
1317          IF(IZ.LT.2) ZPC(LC)= 2.+ (2.-ZC)                      VARP.1635
1324          NPF= NPF+1                                            VARP.1639
1326          LF= LF+NNPP                                           VARP.1640
1327          XPC(LF)=XC                                            VARP.1641
1331          YPC(LF)=YC                                            VARP.1642
1333          ZPC(LF)=ZC                                            VARP.1643
1336          DO 8061 N=1,NSPES                                     VE3.84
1342          LLF=LF+N-1                                            VE3.85
1344          LLC=LC+N-1                                            VE3.86
1345          PM(LLF)=PM(LLC)                                       VE3.87
1350    8061  CONTINUE                                              VE3.88
1351          IF(NOCONC.LT.1) GO TO 8990                            VARP.1645
        C ***** THIS SECTION COMPUTES MASSES AT GRID NODES.         VARP.1646
1353          I=IX                                                  VARP.1648
1354          J=IY                                                  VARP.1649
1355          K=IZ                                                  VARP.1650
1357          DO 8989 N=1,NSPES                                     VE3.89
1375          IJK=((I-1)*ICON+(J-1)*JCON+K)+(IJK2 )*(N-1)           VARP.1665
1400          LLC=LC+N-1                                            VE3.90
1401          PMK=PM(LLC)                                           VE3.91
1404          F1= 1.0                                               VE3.92
1405          CON(IJK)=F1*PMK+CON(IJK)                              VARP.1673
1410    8989  CONTINUE                                              VE3.93
1413    8990  CONTINUE                                              VARP.1681
1413    6999  CONTINUE                                              VARP.1682
1416          DT= DDT                                               VE3.94
1417          NPS=NPF                                               VE3.95
1420    9000  CONTINUE                                              VARP.1684
        C ***** THIS SECTION, 4000, DIVIDES NODE MASSES BY VOLUMES TO COMPUTE  VARP.1685
        C       CONCENTRATIONS.                                     VARP.1686
1420          IF(NOCONC.LT.1) GO TO 4999                            VARP.1687
1423          NCON=-IJK2                                            VARP.1688
1424          DO 4000 N=1,NSPES                                     VE3.96
1427          BKG= BKGS(N)                                          VE3.97
1430          FAC= FACS(N)                                          VE3.98
1431          NCON=NCON+IJK2                                        VARP.1690
1433          DO 4000 L=1,MREAL                                     VARP.1691
1435          LCON=L+NCON                                           VARP.1692
1436          LVOL=(L-1)*N*PC+1                                     VARP.1693
1441          IF( VOL(LVOL).EQ.0) GO TO 4000                        VARP.1694
1445          VOL(LVOL)= 1.0                                        VE3.99
1447          CON(LCON)= (CON(LCON)/VOL(LVOL)) * FAC+BKG            VE3.100
1453    4000  CONTINUE                                              VARP.1696
1460          IF(MOD(NCYC+1,100).NE.0) GO TO 8100                   VE3.101
1465          DO 8099 N=1,NSPES                                     VE3.102
1467          DO 8099 J=1,JBP1                                      VARP.1697
1470          PRINT 8801,J,N                                        VE3.103
1477          JMJC= (J-1)*JCON                                      VARP.1699
        8801  FORMAT( 1H0,9HFACE NO. ,I2,10X,14HCONCENTRATIONS ,10X,  VE3.104
             1 12HSPECIES NO. ,I2)                                  VE3.105
1503          DO 8099 I=1,IBP1                                      VE3.106
```

```
1508          IJC= (I-1)*ICON+JMJC+(K-1)*IJK2 + 1                    VF3.107
1514          IJL= IJC+NPP                                           VF3.108
1516          PRINT 4002 ,I,(CON(IJK),IJK=IJC,IJL)                   VF3.109
        4002  FORMAT(1H ,I2,1HI,11(1X,1PE11.5))                      VF3.110
1531          C ***** [illegible]                                    VARP.1718
1541          C ***** [illegible]                                    VF3.111
              C ***** THIS SECTION, 4100, CALLS OUTPUT FOR PARTICLES AND  VARP.1719
              C       [illegible]                                    VARP.1720
              C ***** THIS SECTION, 4200, CALLS KEM AND ADDS NEW PARTICLES.  VARP.1721
1541          DO 4200 I=2,[illegible]                                VF3.112
1543          DO 4200 J=2,[illegible]                                VF3.113
1544          DO 4200 K=2,[illegible]                                VF3.114
1545          L=(K-1) + (J-2)*JCON + (I-2)*ICON                      VF3.115
              [illegible]                                            VF3.116
              [illegible]                                            VF3.117
1550          DO 4003 NN=1,NSPES                                     VF3.118
1566          LFIELD= L+LFIELD+IJK2                                  VF3.119
1570          SPEC(NN)= CON(LFIELD)                                  VF3.120
1572          TSP(NN)= SPEC(NN)-BKGS(NN)                             VF3.121
1573          TV= TV+ABS(TSP(NN) /BKGS(NN))                          VF3.122
1576     4003 CONTINUE                                               VF3.123
1600          IF( TV.GT.1.E-5) GO TO 4005                            VF3.124
1603          LFIELD= -IJK2                                          VF3.125
1604          DO 4004 NN=1,NSPES                                     VF3.126
1611          LFIELD= L+LFIELD+IJK2                                  VF3.127
1613          CON(LFIELD)= 1.0                                       VF3.128
1614     4004 CONTINUE                                               VF3.129
1616          GO TO 4299                                             VF3.130
1616     4005 CONTINUE                                               VF3.131
1616          CALL KEM(SPEC,DT,NSPES)                                VF3.132
1621          LFIELD= -IJK2                                          VF3.133
        C      UPDATE CONCENTRATIONS                                 VF3.134
1623          DO 4006 NN=1,NSPES                                     VF3.135
1631          LFIELD= LFIELD+L+IJK2                                  VF3.136
1633          TV= SPEC(NN)- BKGS(NN)                                 VF3.137
1634          CON(LFIELD)= TV/TSP(NN)                                VF3.138
1636     4006 CONTINUE                                               VF3.139
1641     4299 CONTINUE                                               VARP.1811
        C ***** THIS SECTION, 4300, REWEIGHTS PARTICLES AND SETS PARTICLE EDIT  VARP.1812
        C       FLAG                                                 VARP.1813
1651          NPL=NPS                                                VF3.140
1652          LC=1-NWPP                                              VF3.141
1654          IMAX=0                                                 VARP.1818
1654          IMIN=0                                                 VARP.1819
1655          IF(NPS.LE.0) GO TO 4310                                VARP.1824
1656          DO 4300 L=1,NPS                                        VARP.1825
1660          LC=LC+NWPP                                             VARP.1826
1661          XC= XPC(LC)                                            VARP.1827
1663          YC= YPC(LC)                                            VARP.1828
1664          ZC= ZPC(LC)                                            VARP.1829
1666          I= IFIX(XC)                                            VARP.1830
1667          J= IFIX(YC)                                            VARP.1831
1671          K= IFIX(ZC)                                            VARP.1832
1673          DO 4301 N=1,NSPES                                      VF3.142
1710          IJK=((I-1)*ICON+(J-1)*JCON+K)+(IJK2 )*(N-1)            VARP.1847
1713          PMK= CON(IJK)                                          VF3.143
```

```
1715          LLC=LC+N-1                                                  VE3.144
1717          PM(LLC)=PM(LLC)*PMK                                         VE3.145
1721   4301   CONTINUE                                                    VE3.146
1723   4300   CONTINUE                                                    VARP.1863
1725   4310   CONTINUE                                                    VARP.1865
1725   4999   CONTINUE                                                    VARP.1924
1725          IF( TIMET.LT.TWIP ) GO TO 9500                              VARP.1926
1730          IF( TIP.EQ.0.0 ) GO TO 9500                                 VARP.1927
        C ***** THIS SECTION , 9100-9199 , ADDS PARTICLES TO THE SYSTEM   VARP.1928
1731   9100   TWIP=TWIP+TIP                                               VARP.1929
1733          LC=1 + NWPP * (NPS-1)                                       VE3.147
1736          L1=NPS+1                                                    VARP.1930
1737          NPS=NPS+NDPI                                                VARP.1931
1741          IF( NPS.GT.NPMX ) GO TO 9200                                VARP.1932
1744   9110   DO 9199 L=L1,NPS                                            VARP.1933
1746   9120   LC= LC+NWPP                                                 VE3.148
1750          IF(XPCEN.GT.0.0) GO TO 9130                                 VARP.1936
1752          XPC(LC)= FLOAT(IPL) + FLOAT(IPR-IPL+1) * RANF(DUM)          VARP.1937
1761          YPC(LC)= FLOAT(JPA) + FLOAT(JPF-JPA+1) * RANF(DUM)          VARP.1938
1771          ZPC(LC)= FLOAT(KPB) + FLOAT(KPT-KPB+1) * RANF(DUM)          VARP.1939
2001          GO TO 9198                                                  VARP.1940
2002   9130   CONTINUE                                                    VARP.1941
2003          XPC(LC)=XPCEN                                               VARP.1942
2004          YPC(LC)=YPCEN                                               VARP.1943
2006          ZPC(LC)=ZPCEN                                               VARP.1944
2011   9198   CONTINUE                                                    VARP.1945
2011          DO 9199 N=1,NSPES                                           VE3.149
2016          LLC=LC+N-1                                                  VE3.150
2020          PM(LLC)=PMIN(N)                                             VE3.151
2021   9199   CONTINUE                                                    VARP.1948
2025   9197   CONTINUE                                                    VE3.152
2025          GO TO 9500                                                  VARP.1950
        C ***** *** ERROR 006 - ATTEMPTED TO INSERT MORE PARTICLES THAN ALLOWED VARP.1951
2026   9200   WRITE(VDO,1)                                                VARP.1952
2032          ERF=1                                                       VARP.1954
2033   9500   CONTINUE                                                    VE3.153
2033          WRITE(VDO,2) NPS, TIMET                                     VARP.1955
2043          IF( LPR.EQ.1 ) GO TO 9600                                   VARP.1956
2045          VDO=6                                                       VARP.1957
2046          IF( LPR.EQ.2 ) VDO=9                                        VARP.1958
2051          LC= 1 - NWPP                                                VE3.154
2053          WRITE(VDO,7) TIMET,NCYC                                     VARP.1960
2063          LL=10                                                       VE3.155
2064          IF(MOD(NCYC+1,20).NE.0) GO TO 9600                          VE3.156
2071          DO 9599 L=1,LL                                              VE3.157
2073          LC=LC + NWPP                                                VE3.158
2074          XC=XPC(LC)                                                  VARP.1963
2076          IX=IFIX(XC+EM6)                                             VARP.1964
2100          XW=XC - FLOAT(IX)                                           VARP.1965
2102          YC=YPC(LC)                                                  VARP.1966
2104          IY=IFIX(YC+EM6)                                             VARP.1967
2106          YL=YC - FLOAT(IY)                                           VARP.1968
2111          ZC=ZPC(LC)                                                  VARP.1969
2112          IZ=IFIX(ZC+EM6)                                             VARP.1970
2114          ZH=ZC - FLOAT(IZ)                                           VARP.1971
2117          XP=X(IX-1) + XW*DX(IX)                                      VARP.1972
```

```
VRPS
 2122          YP=Y(IY-1) + YL*DY(IY)                                  VARP.1973
 2126          ZP=Z(IZ-1) + ZH*DZ(IZ)                                  VARP.1974
 2131          PMK1=PM(LC)                                             VE3.159
 2133          PMK2=PM(LC+1)                                           VE3.160
 2135          PMK3=PM(LC+2)                                           VE3.161
 2137          WRITE(VDU,8) L,XC,YC,ZC,PMK1,PMK2,PMK3                  VE3.162
 2160          IF( NPS.EQ.0) GO TO 9599                                VARP.1977
 2161          NPL=0                                                   VARP.1978
 2162   9599   CONTINUE                                                VARP.1979
 2165   9600   VDU=6                                                   VARP.1980
 2166          CALL SECOND(TP2)                                        VARP.1981
 2170          TP3=TP2-TP1                                             VARP.1982
 2172          WRITE(6,9) TP3                                          VARP.1983
 2200   99999  RETURN                                                  VARP.1984
 2201          END                                                     VARP.1985


SUBPROGRAM LENGTH

02713

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1      - 002205   2      - 002216   3      - 002227   4      - 002232   5      - 002235   6      - 002257
7      - 002277   8      - 002312   9      - 002316   13     - 002325   100    - 000002   200    - 000021
210    - 000257   230    - 000274   250    - 000332   298    - 000341   500    - 000357   510    - 000365
520    - 000371   600    - 000375   1000   - 000406   1050   - 000415   1100   - 000421   1120   - 000434
1200   - 000436   2000   - 000441   2100   - 000551   2200   - 000610   2300   - 000651   2400   - 000731
2600   - 000754   3000   - 000764   3001   - 000770   3002   - 000772   3003   - 000774   3010   - 000776
3012   - 001022   3015   - 001027   3017   - 001031   3020   - 001036   3022   - 001042   3025   - 001047
3027   - 001051   3030   - 001056   3040   - 001113   3050   - 001115   3060   - 001117   3099   - 001121
4000   - 001454   4005   - 001617   4299   - 001642   4310   - 001726   4999   - 001726   8000   - 001134
8005   - 001152   8010   - 001155   8015   - 001166   8020   - 001172   8025   - 001210   8030   - 001213
8035   - 001226   8040   - 001232   8045   - 001250   8050   - 001253   8055   - 001266   8060   - 001272
8100   - 001542   8801   - 002406   8802   - 002420   8990   - 001414   8999   - 001414   9000   - 001421
9100   - 001732   9110   - 001745   9120   - 001747   9130   - 002003   9197   - 002026   9198   - 002012
9200   - 002027   9500   - 002034   9599   - 002163   9600   - 002166   99999  - 002201

BLOCK NAMES AND LENGTHS
  *LCMR- 0127/10
VRCOM  - 106240/01  CONCEN - 000132/02  AMIX   - 001130/03  FL     - 000035/04  FLS    - 000011/05  VRCON  - 006352/06
MODSET - 000310/07  SAFLM  - 000447/08  ZLAST  - 000001/11

VARIABLE ASSIGNMENTS
A      - 000000/01  ABC    - 0000004/LC BKG    - 002666     BKGS   - 000036/02  BUFL   - 000004/06  C      - 000000/03
CON    - 0072460/LC CPRT   - 002514     CQ     - 000004/01  CQF    - 000005/01  DDT    - 002573     DDPM   - 002572
DRGC   - 002570     DT     - 000006/06  DUM    - 002556     DX     - 000007/06  DY     - 000130/06  DZ     - 000251/06
EM6    - 000373/06  ERF    - 000374/06  F      - 002511     FAC    - 002667     FACS   - 000050/02  FB     - 000024/04
FFX3   - 000021/08  FFY3   - 000134/08  F1     - 000000/05  FM     - 000034/04  FO     - 000000/04  F1     - 002664
I      - 002565     IBP1   - 000403/06  IBP2   - 000404/06  ICON   - 000437/06  IFLB   - 000016/08  IGOP   - 000411/06
IJC    - 002673     IJK    - 002623     IJKCON - 000436/06  IJK2   - 000421/06  IJL    - 002674     IN     - 002620
IMAX   - 002700     IMIN   - 002701     IPEDIT - 000062/02  IPEL   - 000074/02  IPFS   - 000106/02  IPFLG  - 002546
IPL    - 000423/06  IPM    - 000302/08  IPR    - 000424/06  ITCYC  - 000247/08  IX     - 002575     IX1    - 002605
IY     - 002577     IY1    - 002607     IZ     - 002601     IZ1    - 002611     I1     - 002615     I2K2   - 000441/06
```

KEM

```
                SUBROUTINE KEM(SPEC,DT,NSPES)                                   VE3.161
          C                                                                     VE3.162
          C     X= NO2,     Y= NO,    TOX= NO2+O3,    TNO= NO2+NO,    Z=O3      VE3.163
          C                                                                     VE3.164
                DIMENSION SPEC(10),TSP(10)                                      VE3.165
                DATA C1,C3/0.4,10.0/                                            VE3.166
                X= SPEC(1)                                                      VE3.167
          5     Y= SPEC(2)                                                      VE3.168
          7     Z= SPEC(3)                                                      VE3.169
         10     DDT=(DT/60.)*100.                                               VE3.170
         13     IF( Z.LT.0.0) GO TO 600                                         VE3.171
         15        IF( X.LE.0.) X= 1.E-6                                        VE3.172
         17     IF( Y.LE.0.) Y= 1.E-6                                           VE3.173
         22     IF( Z.LE.0.) Z= 1.E-6                                           VE3.174
         26     TOX= X+Z                                                        VE3.175
         27     TNO= X+Y                                                        VE3.176
         30     A= C3                                                           VE3.177
         31     B= -C1-C3* (TOX+TNO)                                            VE3.178
         34     C= C3*TOX*TNO                                                   VE3.179
         35     R2= B*B-4.*A*C                                                  VE3.180
         41     IF( R2) 900,400,100                                             VE3.181
         42 100 R= SQRT(R2)                                                     VE3.182
         44     ALPHA= (2.*A*X +B)/R                                            VE3.183
         51     IF( ABS(ALPHA).LT.1.) CON= ALOG(( 1.+ALPHA)/ (1.-ALPHA))/R      VE3.184
         64     IF( ABS(ALPHA).GT.1.) CON= ALOG(( ALPHA+1.)/(ALPHA-1.)) /R      VE3.185
        100     TAU= 2./R+CON                                                   VE3.186
        103     IF(ABS(TAU).GT.DDT*0.1) GO TO 200                               VE3.187
        112     X1= (-B-R)/(2.*A)                                               VE3.188
        115     GO TO 500                                                       VE3.189
        116 200 CONTINUE                                                        VE3.190
        116     XX= -(DDT-CON)*R*0.5                                            VE3.191
        122     EXP1= EXP(XX)                                                   VE3.192
        124     EXP2= EXP(-XX)                                                  VE3.193
        130     IF( ABS(ALPHA).GT. 1.0)  GO TO 250                              VE3.194
        137     F= (EXP1-EXP2)/ (EXP1+EXP2)                                     VE3.195
        142     GO TO 300                                                       VE3.196
        143 250 F= (EXP1+EXP2)/(EXP1-EXP2)                                      VE3.197
        147 300 X1= (-B+R*F)/( 2.*A)                                            VE3.198
        154     GO TO 500                                                       VE3.199
        154 400 CON= 2./(2.*A*X+B)                                              VE3.200
        160     X1= ((2./(DDT-CON)) -B)/(2.*A)                                  VE3.201
        164 500 Y1= TNO-X1                                                      VE3.202
        166     Z1= TOX-X1                                                      VE3.203
        167     X= X1                                                           VE3.204
        170     Y= Y1                                                           VE3.205
        172     Z= Z1                                                           VE3.206
        173     IF( X.LE.0.) X= 1.E-6                                           VE3.207
        176     IF( Y.LE.0.) Y= 1.E-6                                           VE3.208
        201     IF( Z.LE.0.) Z= 1.E-6                                           VE3.209
        204     GO TO 1000                                                      VE3.210
        205 900 STOP                                                            VE3.211
        207 600 X= X+Z-1.E-6                                                    VE3.212
        212     Z=1.E-6                                                         VE3.213
        212     Y = Y                                                           VE3.214
        214 1000 RETURN                                                         VE3.215
        215     END                                                             VE3.216
```

```
      SUBROUTINE VRIO                                                   VARP.1998
      LARGE XPC(1),YPC(1),ZPC(1)                                        VE3.217
      INTEGER F,ERF,VD1,VDO,BUFL,PYML,PXML,CPRT,VNTP,VIP,PF,TD,TD1,TD2, INREAL.2
     1 PRELM,CQE,SHIET,OBLOC,OBTYP,FI,FO,FB,FM,OBTYPL                   INREAL.3
      REAL NU                                                           INREAL.4
C                                                                       INREAL.5
      DIMENSION U(1),V(1),W(1),P(1),CQ(1),CQF(1)                        DIMEN.2
     1 ,VOL(1)                                                          DIMEN.3
      DIMENSION SPEC(10),TSP(10)                                        VE3.1
      DIMENSION TA(1),QCON(1)                                           DIMEN.6
C                                                                       DIMEN.7
      COMMON/VRCOM/A(36000)                                             VE3.2
      COMMON/CONCEN/ NSPS(10),PMAXWS(10),PMINWS(10),BKGS(10),FACS(10)   VE3.3
     1 ,IPEDII(10),IPEL(10),IPES(10),PMIN(10)                           VE3.4
      COMMON /AMIX/ C(600)                                              COMM.5
      COMMON/FL/FO(20),FB(8),FM                                         COMM.6
      COMMON/FLS/FI(9)                                                  COMM.7
      COMMON/VRCON/ALX,ALY,ALZ,BO,BUFL,CEP,DT,DX(80),DXC,DY(80),DYC,    VE3.5
     1 DZ(80),DZC,EMO,EM6,ERF,GPX,GPY,GPZ,GX,GY,GZ,IBP1,IBP2,IBR,IBRED, VE3.6
     2 IBRIN,IBRSI,IGOP(6),IJKMX,IJKNC,IJK2,IJP1K,IPL,IPR,ITER,I2JK2,   COMM.10
     * IKP2,IPARTF,IPLTF,ISFLG,RIBJBKB,LPR,DROU,                         COMM.11
     *IJKCON,ICON,JCON,                                                 COMM.12
     3 I2K2,JBP1,JBP2,JBR,JBRED,JBRIN,JBRST,JNC,JPA,JPF,J2NC,KBP1,KBP2, COMM.13
     4 KBR,KBRED,KBRIN,KBRST,KFP,KNC,KPB,KPT,KWA,KWB,KWF,KWL,KWR,KWT,   COMM.14
     5 K2NC,LABEL(8),LNPART,LNVRCM,LPARTC,NCTP,NCVTP(6),NCYC,NGOP,NOPI, COMM.15
     *NOCONC,                                                           COMM.16
     * OBLOC(1200),OBTYP(1200),NOB,NU,                                  VE3.7
     6 NOVP,NOVS,NPMX,NPRT,NTPE,NWPC,NWPP,PDX,PDY,PDZ,PI,PRELM,RDT,     COMM.18
     7 RDX(80),RDY(80),RDZ(80),RRAN(20),SIGMA,TD,TD1,TD2,TFIN,TIMET,TIP,VE3.8
     8 TPL,TPLT,TPR,TPRT,TTD,TWIP,TWTD,UI,UPI,VDI,VDO,VI,VNTP(6),VPI,   COMM.20
     9 VTP,WI,WPI,WRAN(20),X(80),XPCEN,Y(80),YPCEN,Z(80),ZPCEN          VE3.9
      COMMON /NODSET/ SCFLM(200)                                        COMM.22
      COMMON /SAFLM/ XCONV,ZCONV,FX1,FX2,FX3,FX4,FY1,FY2,FY3,FY4,       COMM.23
     1 FXL,FYB,PXL,PYB,IFLB(3),FFX3(75),FFY3(75)                        COMM.24
     2 ,ITCYC(3),XFC1(4),XFC2(4),YFC1(4),YFC2(4),PBTIM(2),XML,YML,ZML   COMM.25
     3 ,IFMID,IFIESI,IPLOPT,IPM(20),IPMM,IPPP,IPROJ,JPM(20),JPMM,JPPP,  COMM.26
     4 KPM(20),KPMM,KPPP,NFC,NMC,NPS,NPV,PEC(10),PMC(10),YCONV          COMM.27
     5 ,IFLAB,IFLIN,IFPNI,IFOPI,NCH,XFSP,XT,YFSP,YT                     COMM.28
      COMMON /ZLAST/ ZLAS                                               COMM.29
C                                                                       COMM.30
      EQUIVALENCE (A(1),U),(A(2),V),(A(3),W),(A(4),P),(A(5),CQ),        EQUIV.2
     1 (A(6),CQF), (A(7),VOL)                                           EQUIV.3
      EQUIVALENCE (C(1),TA,QCON)                                        EQUIV.7
C                                                                       EQUIV.8
C ***** THIS SUBROUTINE VRIO HANDLES INPUT/OUTPUT                       VARP.1999
    1 FORMAT(1H1)                                                       VARP.2000
    2 FORMAT(1H1,2X,8H TIMET =,1PE12.5,8H CYCLE =,I6,5X,19H ITERATION NUVARP.2001
     1MBER =,I5,5X,10H DELTA T =,E12.5/3X,1HK,5X,1HI,5X,1HJ,12X,1HU,16X,VARP.2002
     2 1HV,16X,1HW,16X,1HP)                                             VARP.2003
    3 FORMAT(1H1,2X,1HK,5X,1HI,5X,1HJ,12X,1HU,16X,1HV,16X,1HW,16X,1HP)  VARP.2004
    4 FORMAT(1H ,1X,I4,2X,I4,2X,I4,4(5X,1PE12.5))                       VARP.2005
    5 FORMAT(1H1,2X,8H TIMET =,1PE12.5,8H CYCLE =,I6,5X,19H ITERATION NUVARP.2006
     1MBER =,I5,5X,10H DELTA T =,E12.5/3X,3HIJK,12X,2HDX,15X,2HDY,15X,  VARP.2007
     2 2HDZ,16X,1HX,16X,1HY,16X,1HZ)                                    VARP.2008
    6 FORMAT(1H1,3X,3HIJK,12X,2HDX,15X,2HDY,15X,2HDZ,16X,1HX,16X,1HY,   VARP.2009
     1 16X,1HZ)                                                         VARP.2010
```

```
      7 FORMAT(1H ,2X,I5,6(5X,1PE12.5))                                 VARP.2011
        ENTRY VRIO1                                                     VARP.2012
        IF( LPR.EQ.1 .OR. IPARTF.GT.0 .OR. IPLTF.GT.0 ) GO TO 300       VARP.2013
        IF( LPR.EQ.2 ) VDU=9                                            VARP.2014
C ***** THIS SECTION, 100-199 , PRINTS DATA ARRAYS ON THE LINE PRINTER  VARP.2015
  100 LINS=0                                                            VARP.2016
      LINT=57                                                           VARP.2017
      LINC=LINT                                                         VARP.2018
      ASSIGN 110 TO KP1                                                 VARP.2019
      KS=0                                                              VARP.2020
      GO TO 107                                                         VARP.2021
  103 LINS=LINS+1                                                       VARP.2022
      IF( LINS.LT.LINC ) GO TO KP1,( 110,190 )                          VARP.2023
      LINS=0                                                            VARP.2024
      LINC=LINT                                                         VARP.2025
      GO TO 107                                                         VARP.2026
  106 WRITE(VDU,2) TIMET,NCYC,ITER,DT                                   VARP.2027
      GO TO KP1,( 110,190 )                                             VARP.2028
  107 KS=KS+1                                                           VARP.2029
      IF( KS.EQ.1 ) GO TO 106                                           VARP.2030
      WRITE(VDU,3)                                                      VARP.2031
      GO TO KP1,( 110,190 )                                             VARP.2032
  110 JJ=1+(JBRST-2)*I2K2                                               VARP.2033
      DO 199 J=JBRST,JBRED,JBRIN                                        VARP.2034
      JJ=JJ+I2K2                                                        VARP.2035
      KK=(IBRST-2)*K2NC                                                 VARP.2036
      DO 199 I=IBRST,IBRED,IBRIN                                        VARP.2037
      KK=KK+K2NC                                                        VARP.2038
      LWPC=(KBRST-2)*NWPC                                               VARP.2039
  120 DO 199 K=KBRST,KBRED,KBRIN                                        VARP.2040
      LWPC=LWPC+NWPC                                                    VARP.2041
      IJK=JJ+KK+LWPC                                                    VARP.2042
      ASSIGN 190 TO KP1                                                 VARP.2043
      WRITE(VDU,4) K,I,J,U(IJK),V(IJK),W(IJK),P(IJK)                    VARP.2044
      GO TO 103                                                         VARP.2045
  190 CONTINUE                                                          VARP.2046
  199 CONTINUE                                                          VARP.2047
  200 IF( NCYC.GT.0 ) GO TO 300                                         VARP.2048
      INMAX=MAX0(IBP2,JBP2,KBP2)                                        VARP.2049
      LINS=0                                                            VARP.2050
      LINT=57                                                           VARP.2051
      LINC=LINT                                                         VARP.2052
      ASSIGN 210 TO KP1                                                 VARP.2053
      KS=0                                                              VARP.2054
      GO TO 207                                                         VARP.2055
  203 LINS=LINS+1                                                       VARP.2056
      IF( LINS.LT.LINC ) GO TO KP1,(210,290)                            VARP.2057
      LINS=0                                                            VARP.2058
      LINC=LINT                                                         VARP.2059
  206 WRITE(VDU,5) TIMET,NCYC,ITER,DT                                   VARP.2060
      GO TO KP1,(210,290)                                               VARP.2061
  207 KS=KS+1                                                           VARP.2062
      IF( KS.EQ.1 ) GO TO 206                                           VARP.2063
      WRITE(VDU,6)                                                      VARP.2064
      GO TO KP1,(210,290)                                               VARP.2065
  210 DO 299 IN=1,INMAX                                                 VARP.2066
```

```
244          ASSIGN 290 TO KP1                                          VARP.2067
245          WRITE(VDO,7) IN,DX(IN),DY(IN),DZ(IN),X(IN),Y(IN),Z(IN)     VARP.2068
274          GO TO 203                                                  VARP.2069
275      290 CONTINUE                                                   VARP.2070
275      299 CONTINUE                                                   VARP.2071
300      300 VDO=6                                                      VARP.2072
301          IF( PRELM.LT.1 ) RETURN                                    VARP.2073
304          CALL SECOND(T1)                                            VARP.2074
306      800 NOP=0                                                      VARP.2075
307          IG1=IGOP(1)                                                VARP.2076
310          IG2=IGOP(2)                                                VARP.2077
312          IG3=IGOP(3)                                                VARP.2078
313          IG4=IGOP(4)                                                VARP.2079
315          IG5=IGOP(5)                                                VARP.2080
317      900 NOP=NOP + 1                                                VARP.2081
321          YT=.6                                                      VARP.2082
322          IOPT=IGOP(NOP)                                             VARP.2083
324          GO TO( 1000,2000,3000,4000,                    99999),IOPT VARP.2084
       C ***** THIS SECTION , 10000-1999 , PLOTS PARTICLE COORDINATES.  VARP.2085
       C ***** THIS SECTION , 10000-1999 , PLOTS PARTICLE COORDINATES.  VARP.2086
334   1000   IFTEST=1                                                   VARP.2087
335          IFMID=0                                                    VARP.2088
336          IFPNT=1                                                    VARP.2089
336          NPS=NPS                                                    VE3.218
340          M=0                                                        VARP.2090
340          IF( IPROJ.EQ.0 ) GO TO 1020                                VARP.2091
342   1020   CONTINUE                                                   VARP.2092
       C ***** THIS SECTION , 1030 , SETS CONSTANTS FOR IXK PLANE.      VARP.2093
342          GO TO ( 1040,1030 ),JPPP                                   VARP.2094
350   1030   IPLOPT=1                                                   VARP.2095
351          ENCODE(30,9110,IFLB)                                       VARP.2096
357          ASSIGN 1040 TO KPART                                       VARP.2097
360          GO TO 1060                                                 VARP.2098
       C ***** THIS SECTION , 1040 , SETS CONSTANTS FOR IXJ PLANE.      VARP.2099
361   1040   GO TO ( 1050,1042 ),KPPP                                   VARP.2100
367   1042   IPLOPT=2                                                   VARP.2101
370          ENCODE(30,9120,IFLB)                                       VARP.2102
376          ASSIGN 1050 TO KPART                                       VARP.2103
377          GO TO 1060                                                 VARP.2104
       C ***** THIS SECTION , 1050 , SETS CONSTANTS FOR JXK PLANE.      VARP.2105
400   1050   IF( IPPP.EQ.1 ) GO TO 900                                  VARP.2106
402   1052   IPLOPT=3                                                   VARP.2107
403          ENCODE(30,9130,IFLB)                                       VARP.2108
411          ASSIGN 900 TO KPART                                        VARP.2109
412   1060   CALL FPLAN                                                 VARP.2110
413          CALL FLMID                                                 VARP.2111
414          M=M+1                                                      VARP.2112
416          XT=0.02                                                    VARP.2113
417          YT=0.52                                                    VARP.2114
421          CALL FLMLAB                                                VARP.2115
422          XT=0.06                                                    VARP.2116
423          YT=0.56                                                    VARP.2117
425          ENCODE(15,9010,IFLB)                                       VARP.2118
433          NCH=15                                                     VARP.2119
434          CALL FLMLAB                                                VARP.2120
435          IFOPT=3                                                    VARP.2121
```

```
436          CALL FLMOPT                                                VARP.2122
437          LC=1-NNPP                                                  VARP.2123
441    1100  DO 1199 L=1,NPS                                            VARP.2124
443          LC=LC+NNPP                                                 VARP.2125
444          IX=IFIX(XPC(LC)+EM6)                                       VARP.2126
447          IY=IFIX(YPC(LC)+EM6)                                       VARP.2127
451          IZ=IFIX(ZPC(LC)+EM6)                                       VARP.2128
454          XP=X(IX-1) + DX(IX)*( XPC(LC)-FLOAT(IX) )                  VARP.2129
461          YP=Y(IY-1) + DY(IY)*( YPC(LC)-FLOAT(IY) )                  VARP.2130
466          ZP=Z(IZ-1) + DZ(IZ)*( ZPC(LC)-FLOAT(IZ) )                  VARP.2131
474          GO TO ( 1120,1140,1160 ),IPLOPT                            VARP.2132
       C ***** THIS SECTION , 1120 , COMPUTES FILM COORDINATES FOR IXK PLANE.  VARP.2133
503    1120  FPX=XP                                                     VARP.2134
504          FPY=ZP                                                     VARP.2135
506          GO TO 1180                                                 VARP.2136
       C ***** THIS SECTION , 1140 , COMPUTES FILM COORDINATES FOR IXJ PLANE.  VARP.2137
507    1140  FPX=XP                                                     VARP.2138
510          FPY=YP                                                     VARP.2139
512          GO TO 1180                                                 VARP.2140
       C ***** THIS SECTION , 1160 , COMPUTES FILM COORDINATES FOR JXK PLANE.  VARP.2141
513    1160  FPX=YP                                                     VARP.2142
514          FPY=ZP                                                     VARP.2143
516    1180  FX1=XFSP + (FPX-PXL)*XCONV                                 VARP.2144
522          FY1=YFSP + (FPY-PYB)*YCONV                                 VARP.2145
526          CALL FCTEST                                                VARP.2146
527    1199  CALL FLMPNT                                                VARP.2147
533          IF( M.LE.3 ) CALL FLMADV                                   VARP.2148
536          GO TO KPART,( 900,1040,1050 )                              VARP.2149
       C ***** THIS SECTION , 2000-2999 , PLOTS MESH ZONES ON FILM.     VARP.2150
       C ***** THIS SECTION , 2000-2999 , PLOTS MESH ZONES ON FILM.     VARP.2151
541    2000  IF(TIMET.GT.TPL) GO TO 900                                 VF3.219
545          IEMID=1                                                    VARP.2153
545          IFTEST=4                                                   VARP.2154
546          M=0                                                        VARP.2155
547          I1=2                                                       VARP.2156
550          J1=2                                                       VARP.2157
551          K1=2                                                       VARP.2158
552          IF( IPROJ.EQ.0 ) GO TO 2020                                VARP.2159
       C ***** THIS SECTION , 2020 , SETS CONSTANTS FOR I X K PLANE.    VARP.2160
553    2020  GO TO( 2030,2022 ),JPPP                                    VARP.2161
561    2022  IPLOPT=1                                                   VARP.2162
562          ASSIGN 2030 TO KZONE                                       VARP.2163
563          I2=IBP1                                                    VARP.2164
564          J2=2                                                       VARP.2165
565          K2=KBP1                                                    VARP.2166
567          ENCODE(30,9110,IFLB)                                       VARP.2167
575          GO TO 2100                                                 VARP.2168
       C ***** THIS SECTION , 2030 , SETS CONSTANTS FOR I X J PLANE.    VARP.2169
576    2030  GO TO( 2040,2032 ),KPPP                                    VARP.2170
604    2032  IPLOPT=2                                                   VARP.2171
605          ASSIGN 2040 TO KZONE                                       VARP.2172
606          I2=IBP1                                                    VARP.2173
607          J2=JBP1                                                    VARP.2174
611          K2=2                                                       VARP.2175
612          ENCODE(30,9120,IFLB)                                       VARP.2176
620          GO TO 2100                                                 VARP.2177
```

```
      C ***** THIS SECTION , 2040 , SETS CONSTANTS FOR J X K PLANE.      VARP.2178
621  2040  IF( IPPP.EQ.1 ) GO TO 900                                     VARP.2179
623        IPLOPT=3                                                      VARP.2180
624        I2=2                                                          VARP.2181
625        J2=JBP1                                                       VARP.2182
626        K2=KBP1                                                       VARP.2183
630        ASSIGN 900 TO KZONE                                           VARP.2184
631        ENCODE(30,9130,IFLB)                                          VARP.2185
637  2100  CALL FPLAN                                                    VARP.2186
640        CALL FLMID                                                    VARP.2187
641        M=M+1                                                         VARP.2188
643        XT=0.02                                                       VARP.2189
644        YT=0.52                                                       VARP.2190
646        CALL FLMLAB                                                   VARP.2191
647        XT=0.16                                                       VARP.2192
650        YT=0.56                                                       VARP.2193
652        ENCODE(6,9020,IFLB)                                           VARP.2194
660        NCH=6                                                         VARP.2195
661        CALL FLMLAB                                                   VARP.2196
662        IFOPT=4                                                       VARP.2197
663        CALL FLMOPT                                                   VARP.2198
664        IFOPT=5                                                       VARP.2199
665        CALL FLMOPT                                                   VARP.2200
666        DO 2299 J=J1,J2                                               VARP.2201
670        DO 2299 I=I1,I2                                               VARP.2202
672        DO 2299 K=K1,K2                                               VARP.2203
674        GO TO( 2120,2140,2160 ),IPLOPT                                VARP.2204
      C ***** THIS SECTION , 2120 , COMPUTES FILM COORD FOR I X K PLANE. VARP.2205
705  2120  X1=X(I)                                                       VARP.2206
706        X2=X(I)                                                       VARP.2207
707        X3=X(I-1)                                                     VARP.2208
711        X4=X(I-1)                                                     VARP.2209
711        IX=I                                                          VARP.2210
712        IXP1=I2                                                       VARP.2211
714        Y1=Z(K-1)                                                     VARP.2212
715        Y2=Z(K)                                                       VARP.2213
717        Y3=Z(K)                                                       VARP.2214
717        Y4=Z(K-1)                                                     VARP.2215
720        IY=K                                                          VARP.2216
722        IYP1=K2                                                       VARP.2217
724        GO TO 2180                                                    VARP.2218
      C ***** THIS SECTION , 2140 , COMPUTES FILM COORD FOR I X J PLANE. VARP.2219
726  2140  X1=X(I)                                                       VARP.2220
727        X2=X(I)                                                       VARP.2221
730        X3=X(I-1)                                                     VARP.2222
732        X4=X(I-1)                                                     VARP.2223
732        IX=I                                                          VARP.2224
733        IXP1=I2                                                       VARP.2225
735        Y1=Y(J-1)                                                     VARP.2226
736        Y2=Y(J)                                                       VARP.2227
740        Y3=Y(J)                                                       VARP.2228
740        Y4=Y(J-1)                                                     VARP.2229
741        IY=J                                                          VARP.2230
743        IYP1=J2                                                       VARP.2231
745        GO TO 2180                                                    VARP.2232
      C ***** THIS SECTION , 2160 , COMPUTES FILM COORD FOR J X K PLANE. VARP.2233
```

```
747   2160  X1=Y(J)                                                    VARP.2234
750         X2=Y(J)                                                    VARP.2235
751         X3=Y(J-1)                                                  VARP.2236
753         X4=Y(J-1)                                                  VARP.2237
753         IX=J                                                       VARP.2238
754         IXP1=J2                                                    VARP.2239
756         Y1=Z(K-1)                                                  VARP.2240
757         Y2=Z(K)                                                    VARP.2241
761         Y3=Z(K)                                                    VARP.2242
761         Y4=Z(K-1)                                                  VARP.2243
762         IY=K                                                       VARP.2244
764         IYP1=K2                                                    VARP.2245
766   2180  FX1=XFSP + (X1-FXL)*XCONV                                  VARP.2246
772         FY1=YFSP + (Y1-FYB)*YCONV                                  VARP.2247
776         FX2=XFSP + (X2-FXL)*XCONV                                  VARP.2248
1002        FY2=YFSP + (Y2-FYB)*YCONV                                  VARP.2249
1006        FX3=XFSP + (X3-FXL)*XCONV                                  VARP.2250
1012        FY3=YFSP + (Y3-FYB)*YCONV                                  VARP.2251
1016        FX4=XFSP + (X4-FXL)*XCONV                                  VARP.2252
1022        FY4=YFSP + (Y4-FYB)*YCONV                                  VARP.2253
1026        CALL FCTEST                                                VARP.2254
1027        IF( IX.NE.IXP1 ) GO TO 2190                                VARP.2255
1031        IFLIN=2                                                    VARP.2256
1032        CALL FLMLIN                                                VARP.2257
1033  2190  IF( IY.NE.IYP1 ) GO TO 2200                                VARP.2258
1035        IFLIN=3                                                    VARP.2259
1036        CALL FLMLIN                                                VARP.2260
1037  2200  IFLIN=4                                                    VARP.2261
1040        CALL FLMLIN                                                VARP.2262
1041        IFLIN=5                                                    VARP.2263
1042        CALL FLMLIN                                                VARP.2264
1043  2299  CONTINUE                                                   VARP.2265
1053        IF( M.LE.3 ) CALL FLMADV                                   VARP.2266
1056        GO TO KZONE,( 900,2030,2040 )                              VARP.2267
      C ***** THIS SECTION , 3000-3999 , PLOTS VELOCITY VECTORS ON FILM.  VARP.2268
      C ***** THIS SECTION , 3000-3999 , PLOTS VELOCITY VECTORS ON FILM.  VARP.2269
1061  3000  IFMTD=0                                                    VARP.2270
1062        IFPNT=1                                                    VARP.2271
1063        IFTEST=2                                                   VARP.2272
1064        M=0                                                        VARP.2273
1065        MMAX=IPMM + JPMM + KPMM                                    VARP.2274
1067        IF( IPM(1).LT.1 ) MMAX=MMAX - 1                            VARP.2275
1073        IF( JPM(1).LT.1 ) MMAX=MMAX - 1                            VARP.2276
1076        IF( KPM(1).LT.1 ) MMAX=MMAX - 1                            VARP.2277
1101        IF( IPROJ.EQ.0 ) GO TO 3050                                VARP.2278
      C ***** THIS SECTION , 3050 , SETS CONSTANTS FOR I X K PLANE.    VARP.2279
1102  3050  IPLOPT=1                                                   VARP.2280
1103        I1=2                                                       VARP.2281
1104        I2=IBP1                                                    VARP.2282
1105        K1=2                                                       VARP.2283
1106        K2=KBP1                                                    VARP.2284
1110        ASSIGN 3055 TO KVEC                                        VARP.2285
1111        DO 3059 J4=1,JPMM                                          VARP.2286
1113        IF( JPM(J4).LT.1 ) GO TO 3060                              VARP.2287
1116        J1=JPM(J4)                                                 VARP.2288
1117        J2=J1                                                      VARP.2289
```

```
1121          ENCODE(30,9140,IFLB) J1                                  VARP.2290
1130          GO TO 3100                                               VARP.2291
1131   3055   CONTINUE                                                 VARP.2292
1131   3059   CONTINUE                                                 VARP.2293
       C ***** THIS SECTION , 3060 , SETS CONSTANTS FOR I X J PLANE.   VARP.2294
1134   3060   IPLOPT=2                                                 VARP.2295
1135          I1=2                                                     VARP.2296
1136          I2=IBP1                                                  VARP.2297
1137          J1=2                                                     VARP.2298
1140          J2=JBP1                                                  VARP.2299
1142          ASSIGN 3065 TO KVEC                                      VARP.2300
1143          DO 3069 K4=1,KPMM                                        VARP.2301
1145          IF( KPM(K4).LT.1 ) GO TO 3070                            VARP.2302
1150          K1=KPM(K4)                                               VARP.2303
1151          K2=K1                                                    VARP.2304
1153          ENCODE(30,9150,IFLB) K1                                  VARP.2305
1162          GO TO 3100                                               VARP.2306
1163   3065   CONTINUE                                                 VARP.2307
1163   3069   CONTINUE                                                 VARP.2308
       C ***** THIS SECTION , 3070 , SETS CONTANTS FOR J X K PLANE.    VARP.2309
1166   3070   IPLOPT=3                                                 VARP.2310
1167          J1=2                                                     VARP.2311
1170          J2=JBP1                                                  VARP.2312
1171          K1=2                                                     VARP.2313
1172          K2=KBP1                                                  VARP.2314
1174          ASSIGN 3075 TO KVEC                                      VARP.2315
1175          DO 3079 I4=1,IPMM                                        VARP.2316
1177          IF( IPM(I4).LT.1 ) GO TO 900                             VARP.2317
1202          I1=IPM(I4)                                               VARP.2318
1203          I2=I1                                                    VARP.2319
1205          ENCODE(30,9160,IFLB) I1                                  VARP.2320
1214          GO TO 3100                                               VARP.2321
1215   3075   CONTINUE                                                 VARP.2322
1215   3079   CONTINUE                                                 VARP.2323
1220          GO TO 900                                                VARP.2324
1220   3100   CALL FPLAN                                               VARP.2325
1221          CALL FLMID                                               VARP.2326
1222          M=M + 1                                                  VARP.2327
1224          XT=0.02                                                  VARP.2328
1225          YT=0.52                                                  VARP.2329
1227          CALL FLMLAB                                              VARP.2330
1230          XT=0.06                                                  VARP.2331
1231          YT=0.56                                                  VARP.2332
1233          ENCODE(18,9030,IFLB)                                     VARP.2333
1241          NCH=18                                                   VARP.2334
1242          CALL FLMLAB                                              VARP.2335
1243          IFOPT=3                                                  VARP.2336
1244          CALL FLMOPT                                              VARP.2337
1245          IFLIN=2                                                  VARP.2338
1246   3200   JJ=1 + (J1-2)*I2K2                                       VARP.2339
1253          DO 3299 J=J1,J2                                          VARP.2340
1255          JJ=JJ + I2K2                                             VARP.2341
1256          KK=(I1-2)*K2NC                                           VARP.2342
1261          YAV=.5*( Y(J-1)+Y(J) )                                   VARP.2343
1265          DO 3299 I=I1,I2                                          VARP.2344
1270          KK=KK + K2NC                                             VARP.2345
```

```
1271          LWPC=(K1-2)*NWPC                                          VARP.2346
1274          XAV=.5*( X(I-1)+X(I) )                                    VARP.2347
1300          DO 3299 K=K1,K2                                           VARP.2348
1303          LWPC=LWPC + NWPC                                          VARP.2349
1304          IJK=JJ+KK+LWPC                                            VARP.2350
1307          IMJK=IJK - K2NC                                           VARP.2351
1310          IJMK=IJK - I2K2                                           VARP.2352
1312          IJKM=IJK - NWPC                                           VARP.2353
1314          ZAV=.5*( Z(K-1)+Z(K) )                                    VARP.2354
1317          GO TO( 3220,3240,3260 ),IPLOPT                            VARP.2355
        C ***** THIS SECTION , 3220 , COMPUTES FILM COORD FOR I X K PLANE.  VARP.2356
1327  3220    UAV=.5*( U(IMJK)+U(IJK) )                                 VARP.2357
1332          X14A=XAV                                                  VARP.2358
1334          VAV=.5*( W(IJKM)+W(IJK) )                                 VARP.2359
1337          Y12A=ZAV                                                  VARP.2360
1341          GO TO 3280                                                VARP.2361
        C ***** THIS SECTION , 3240 , COMPUTES FILM COORD FOR I X J PLANE.  VARP.2362
1343  3240    UAV=.5*( U(IMJK)+U(IJK) )                                 VARP.2363
1346          X14A=XAV                                                  VARP.2364
1350          VAV=.5*( V(IJMK)+V(IJK) )                                 VARP.2365
1353          Y12A=YAV                                                  VARP.2366
1355          GO TO 3280                                                VARP.2367
        C ***** THIS SECTION , 3260 , COMPUTES FILM COORD FOR J X K PLANE.  VARP.2368
1357  3260    UAV=.5*( V(IJMK)+V(IJK) )                                 VARP.2369
1362          X14A=YAV                                                  VARP.2370
1364          VAV=.5*( W(IJKM)+W(IJK) )                                 VARP.2371
1367          Y12A=ZAV                                                  VARP.2372
1371  3280    FX1=XFSP + (X14A-FXL)*XCONV                               VARP.2373
1375          FY1=YFSP + (Y12A-FYB)*YCONV                               VARP.2374
1401          FX2=XFSP + (X14A+UAV*DROU-FXL)*XCONV                      VARP.2375
1407          FY2=YFSP + (Y12A+VAV*DROU-FYB)*YCONV                      VARP.2376
1415          CALL ECTEST                                               VARP.2377
1416          CALL FLMPNT                                               VARP.2378
1417          CALL FLMLIN                                               VARP.2379
1420  3299    CONTINUE                                                  VARP.2380
1430          IF( M.LE.MMAX ) CALL FLMADV                               VARP.2381
1433          GO TO KVEC,( 3055,3065,3075 )                             VARP.2382
        C ***** THIS SECTION , 4000-4999 , PLOTS CONTOURS ON FILM.      VARP.2383
        C ***** THIS SECTION , 4000-4999 , PLOTS CONTOURS ON FILM.      VARP.2384
1436  4000    IFMID=0                                                   VARP.2385
1437          IFPNT=2                                                   VARP.2386
1440          IFTEST=2                                                  VARP.2387
1441          M=0                                                       VARP.2388
1442          MMAX=IPMM + JPMM + KPMM                                   VARP.2389
1444          IF( IPM(1).LT.1 ) MMAX=MMAX - 1                           VARP.2390
1450          IF( JPM(1).LT.1 ) MMAX=MMAX - 1                           VARP.2391
1453          IF( KPM(1).LT.1 ) MMAX=MMAX - 1                           VARP.2392
1456          IF( IPROJ.EQ.0 ) GO TO 4030                               VARP.2393
        C ***** THIS SECTION , 4030 , SETS CONSTANTS FOR I X K PLANE.   VARP.2394
1457  4030    IPLOPT=1                                                  VARP.2395
1460          I1=2                                                      VARP.2396
1461          I2=IBP1                                                   VARP.2397
1462          I1P=I1+1                                                  VARP.2398
1464          I2M=I2-1                                                  VARP.2399
1465          K1=2                                                      VARP.2400
1466          K2=KBP1                                                   VARP.2401
```

```
1470          K1P=K1+1                                                VARP.2402
1471          K2M=K2-1                                                VARP.2403
1472          ASSIGN 4035 TO KCON                                     VARP.2404
1473          DO 4039 J4=1,JPMM                                       VARP.2405
1475          IF( JPM(J4).LT.1 ) GO TO 4040                           VARP.2406
1500          J1=JPM(J4)                                              VARP.2407
1501          J2=J1P=J2M=J1                                           VARP.2408
1504          ENCODE(30,9140,IFLB) J1                                 VARP.2409
1514          GO TO 4080                                              VARP.2410
1515    4035  CONTINUE                                                VARP.2411
1515    4039  CONTINUE                                                VARP.2412
        C ***** THIS SECTION , 4040 , SETS CONSTANTS FOR I X J PLANE. VARP.2413
1520    4040  IPLOPT=2                                                VARP.2414
1521          I1=2                                                    VARP.2415
1522          I2=IBP1                                                 VARP.2416
1523          I1P=I1+1                                                VARP.2417
1525          I2M=I2-1                                                VARP.2418
1526          J1=2                                                    VARP.2419
1527          J2=JBP1                                                 VARP.2420
1531          J1P=J1+1                                                VARP.2421
1532          J2M=J2-1                                                VARP.2422
1533          ASSIGN 4045 TO KCON                                     VARP.2423
1534          DO 4049 K4=1,KPMM                                       VARP.2424
1536          IF( KPM(K4).LT.1 ) GO TO 4050                           VARP.2425
1541          K1=KPM(K4)                                              VARP.2426
1542          K2=K1P=K2M=K1                                           VARP.2427
1545          ENCODE(30,9150,IFLB) K1                                 VARP.2428
1555          GO TO 4080                                              VARP.2429
1556    4045  CONTINUE                                                VARP.2430
1556    4049  CONTINUE                                                VARP.2431
        C ***** THIS SECTION , 4050 , SETS CONTANTS FOR J X K PLANE.  VARP.2432
1561    4050  IPLOPT=3                                                VARP.2433
1562          J1=2                                                    VARP.2434
1563          J2=JBP1                                                 VARP.2435
1564          J1P=J1+1                                                VARP.2436
1566          J2M=J2-1                                                VARP.2437
1567          K1=2                                                    VARP.2438
1570          K2=KBP1                                                 VARP.2439
1572          K1P=K1+1                                                VARP.2440
1573          K2M=K2-1                                                VARP.2441
1574          ASSIGN 4055 TO KCON                                     VARP.2442
1575          DO 4059 I4=1,IPMM                                       VARP.2443
1577          IF( IPM(I4).LT.1 ) GO TO 900                            VARP.2444
1602          I1=IPM(I4)                                              VARP.2445
1603          I2=I1P=I2M=I1                                           VARP.2446
1606          ENCODE(30,9160,IFLB) I1                                 VARP.2447
1616          GO TO 4080                                              VARP.2448
1617    4055  CONTINUE                                                VARP.2449
1617    4059  CONTINUE                                                VARP.2450
1622          GO TO 900                                               VARP.2451
1622    4080  DO 4799 MV=1,NOVP                                       VARP.2452
1624          CALL FPLAN                                              VARP.2453
1625          CALL FLMID                                              VARP.2454
1626          M=M+1                                                   VARP.2455
1630          XT=0.02                                                 VARP.2456
1631          YT=0.52                                                 VARP.2457
```

```
1633          CALL FLMLAB                                                VARP.2458
1635          VTP=VNTP(MV)                                               VARP.2459
1636          NCTP=NCVTP(MV)                                             VARP.2460
1640          GO TO ( 4082,4084,4086,4088 ),VTP                          VARP.2461
1650    4082  ENCODE(30,9040,IFLB)                                       VARP.2462
1656          GO TO 4090                                                 VARP.2463
1657    4084  ENCODE(30,9050,IFLB)                                       VARP.2464
1665          GO TO 4090                                                 VARP.2465
1666    4086  ENCODE(30,9060,IFLB)                                       VARP.2466
1674          GO TO 4090                                                 VARP.2467
1675    4088  ENCODE(30,9170,IFLB)                                       VARP.2468
1703    4090  NCH=30                                                     VARP.2469
1704          YT=0.56                                                    VARP.2470
1706          CALL FLMLAB                                                VARP.2471
        C ***** THIS SECTION , 4100-4199 , TRANSFERS CONTOUR VARIABLE TO CQ  VARP.2472
        C ***** ARRAY AND COMPUTES QMX,QMN.                              VARP.2473
1707    4100  JJ=1 + (J1-2)*I2K2                                         VARP.2474
1713          ICQ=0                                                      VARP.2475
1714          DO 4189 J=J1,J2                                            VARP.2476
1716          JJ=JJ + I2K2                                               VARP.2477
1717          KK=(I1-2)*K2NC                                             VARP.2478
1723          DO 4189 I=I1,I2                                            VARP.2479
1724          KK=KK + K2NC                                               VARP.2480
1725          LWPC=(K1-2)*NWPC                                           VARP.2481
1731          DO 4189 K=K1,K2                                            VARP.2482
1732          LWPC=LWPC + NWPC                                           VARP.2483
1733          IJK=JJ+KK+LWPC                                             VARP.2484
1736          IMJK=IJK - K2NC                                            VARP.2485
1737          IJMK=IJK - I2K2                                            VARP.2486
1741          IJKM=IJK - NWPC                                            VARP.2487
1743          ICQ=ICQ+1                                                  VARP.2488
1745          GO TO ( 4120,4130,4140,4150 ),VTP                          VARP.2489
1755    4120  IF(NOCONC.GE.1) GO TO 4160                                 VARP.2490
1760          CQ(IJK)=P(IJK)                                             VARP.2491
1762          GO TO 4160                                                 VARP.2492
1762    4130  CQ(IJK)=.5*( U(IMJK)+U(IJK) )                              VARP.2493
1766          GO TO 4160                                                 VARP.2494
1767    4140  CQ(IJK)=.5*( W(IJKM)+W(IJK) )                              VARP.2495
1773          GO TO 4160                                                 VARP.2496
1774    4150  CQ(IJK)=.5*( V(IJMK)+V(IJK) )                              VARP.2497
2000    4160  IF( ICQ.EQ.1 ) QMX=QMN=CQ(IJK)                             VARP.2498
2006          QMX=AMAX1(QMX,CQ(IJK))                                     VARP.2499
2012          QMN=AMIN1(QMN,CQ(IJK))                                     VARP.2500
2015    4189  CONTINUE                                                   VARP.2501
2024          DELQ=(QMX-QMN)/FLOAT(NCTP+1)                               VARP.2502
2030          IF( ABS(DELQ) .LT. 1.E-20 ) GO TO 4798                     VARP.2503
2033          ENCODE(30,9070,IFLB) QMX,QMN                               VARP.2504
2045          YT=0.50                                                    VARP.2505
2047          CALL FLMLAB                                                VARP.2506
2050          YT=YT-.02                                                  VARP.2507
2052          ENCODE(30,9080,IFLB) DELQ                                  VARP.2508
2062          CALL FLMLAB                                                VARP.2509
        C ***** THIS SECTION , 4200-4399 , COMPUTES CONTOUR VALUES AND PLOTS  VARP.2510
        C ***** ON FILM.                                                 VARP.2511
2063    4200  DO 4499 L=1,NCTP                                           VARP.2512
2065          QO=QCON(L)=QMN+FLOAT(L)*DELQ                               VARP.2513
```

```
      C ***** THIS SECTION , 4210-4219 , FLAGS EACH CONTOUR CELL.          VARP.2514
      C ***** COF=0 IF QO LESS THAN CQ , COF=1 IF QO EQ OR GREATER THAN CQ. VARP.2515
2071  4210  JJ=1 + (J1-2)*I2K2                                             VARP.2516
2076        DO 4219 J=J1,J2                                                VARP.2517
2077        JJ=JJ+I2K2                                                     VARP.2518
2100        KK=(I1-2)*K2NC                                                 VARP.2519
2104        DO 4219 I=I1,I2                                                VARP.2520
2105        KK=KK + K2NC                                                   VARP.2521
2106        LWPC=(K2-2)*NWPC                                               VARP.2522
2112        DO 4219 K=K1,K2                                                VARP.2523
2114        LWPC=LWPC + NWPC                                               VARP.2524
2115        IJK=JJ + KK + LWPC                                             VARP.2525
2120        COF(IJK)=1                                                     VARP.2526
2122        IF( CQ(IJK).LT.QO ) COF(IJK)=0                                 VARP.2527
2125  4219  CONTINUE                                                       VARP.2528
      C ***** THIS SECTION , 4250-4399 , COMPUTES AND PLOTS CONTOUR LINES. VARP.2529
2135  4250  JJ=1 + (J1P-2)*I2K2                                            VARP.2530
2142        DO 4399 J=J1P,J2M                                              VARP.2531
2143        JJ=JJ + I2K2                                                   VARP.2532
2144        KK=(I1P-2)*K2NC                                                VARP.2533
2150        GO TO( 4260,4252,4254 ),IPLOPT                                 VARP.2534
      C ***** THIS SECTION , 4252 , SETS CONSTANTS FOR I X J PLANE.        VARP.2535
2161  4252  Y1=Y(J-1)                                                      VARP.2536
2162        Y2=Y(J)                                                        VARP.2537
2163        Y3=Y2                                                          VARP.2538
2164        Y4=Y1                                                          VARP.2539
2165        Y5=Y2                                                          VARP.2540
2166        Y6=Y1                                                          VARP.2541
2166        Y7=Y(J-2)                                                      VARP.2542
2170        Y8=Y7                                                          VARP.2543
2170        Y9=Y7                                                          VARP.2544
2171        IY=J                                                           VARP.2545
2173        GO TO 4260                                                     VARP.2546
      C ***** THIS SECTION , 4254 , SETS CONSTANTS FOR J X K PLANE.        VARP.2547
2176  4254  X1=Y(J)                                                        VARP.2548
2177        X2=X1                                                          VARP.2549
2177        X3=Y(J-1)                                                      VARP.2550
2201        X4=X3                                                          VARP.2551
2201        X5=Y(J-2)                                                      VARP.2552
2203        X6=X5                                                          VARP.2553
2203        X7=X5                                                          VARP.2554
2204        X8=X3                                                          VARP.2555
2206        X9=X1                                                          VARP.2556
2207        IX=J                                                           VARP.2557
2211  4260  DO 4399 I=I1P,I2M                                              VARP.2558
2213        KK=KK + K2NC                                                   VARP.2559
2214        LWPC=(K1P-2)*NWPC                                              VARP.2560
2220        GO TO ( 4262,4262,4270 ),IPLOPT                                VARP.2561
      C ***** THIS SECTION , 4262 , SETS CONSTANTS FOR I X K OR I X J PLANE. VARP.2562
2231  4262  X1=X(I)                                                        VARP.2563
2232        X2=X1                                                          VARP.2564
2232        X3=X(I-1)                                                      VARP.2565
2234        X4=X3                                                          VARP.2566
2234        X5=X(I-2)                                                      VARP.2567
2236        X6=X5                                                          VARP.2568
2236        X7=X5                                                          VARP.2569
```

```
2237          X8=X3                                                    VARP.2570
2241          X9=X1                                                    VARP.2571
2242          IX=I                                                     VARP.2572
2244   4270   DO 4399 K=K1P,K2M                                        VARP.2573
2246          LWPC=LWPC + NWPC                                         VARP.2574
2247          IJK=JJ + KK + LWPC                                       VARP.2575
2252          IMJK=IJK - K2NC                                          VARP.2576
2253          IJKM=IJK - LWPC                                          VARP.2577
2254          IMJKM=IJKM - K2NC                                        VARP.2578
2256          GO TO ( 4272,4280,4272 ),IPLOPT                          VARP.2579
        C ***** THIS SECTION , 4272 , SETS CONSTANTS FOR I X K OR J X K PLANE.  VARP.2580
2267   4272   Y1=Z(K-1)                                                VARP.2581
2270          Y2=Z(K)                                                  VARP.2582
2271          Y3=Y2                                                    VARP.2583
2272          Y4=Y1                                                    VARP.2584
2273          Y5=Y2                                                    VARP.2585
2274          Y6=Y1                                                    VARP.2586
2274          Y7=Z(K-2)                                                VARP.2587
2276          Y8=Y7                                                    VARP.2588
2276          Y9=Y7                                                    VARP.2589
2277          IY=K                                                     VARP.2590
2301   4280   IF1=CQF(IJKM)                                            VARP.2591
2303          IF2=CQF(IJK)                                             VARP.2592
2305          IF3=CQF(IMJK)                                            VARP.2593
2307          IF4=CQF(IMJKM)                                           VARP.2594
2311          IFT=IF1+IF2+IF3+IF4                                      VARP.2595
        C ***** THIS CHECKS TO SEE WHETHER A CONTOUR VARIABLE NEEDS A SHIFT.  VARP.2596
2314          IF( IFT.GT.3 .OR. IFT.LT.1 ) GO TO 4398                  VARP.2597
2326          Q1=CQ(IJKM)                                              VARP.2598
2327          Q2=CQ(IJK)                                               VARP.2599
2331          Q3=CQ(IMJK)                                              VARP.2600
2332          Q4=CQ(IMJKM)                                             VARP.2601
        C ***** THIS SECTION CALCULATES VOLUME CONSTANTS.               VARP.2602
2334          QBR=.25*(Q1+Q2+Q3+Q4)                                    VARP.2603
2337          XI1=.5*(X1+X9-X4-X8)                                     VARP.2604
2344          XI2=.5*(X2+X1-X3-X4)                                     VARP.2605
2350          XI3=.5*(X3+X4-X5-X6)                                     VARP.2606
2354          XI4=.5*(X4+X8-X6-X7)                                     VARP.2607
2360          XK1=.5*(X1+X4-X8-X9)                                     VARP.2608
2364          XK2=.5*(X2+X3-X4-X1)                                     VARP.2609
2370          XK3=.5*(X3+X5-X6-X4)                                     VARP.2610
2374          XK4=.5*(X4+X6-X7-X8)                                     VARP.2611
2400          YI1=.5*(Y1+Y9-Y4-Y8)                                     VARP.2612
2405          YI2=.5*(Y2+Y1-Y3-Y4)                                     VARP.2613
2411          YI3=.5*(Y3+Y4-Y5-Y6)                                     VARP.2614
2415          YI4=.5*(Y4+Y8-Y6-Y7)                                     VARP.2615
2421          YK1=.5*(Y1+Y4-Y8-Y9)                                     VARP.2616
2425          YK2=.5*(Y2+Y3-Y4-Y1)                                     VARP.2617
2431          YK3=.5*(Y3+Y5-Y6-Y4)                                     VARP.2618
2435          YK4=.5*(Y4+Y6-Y7-Y8)                                     VARP.2619
        C ***** THIS SECTION CALCULATES CONTOUR CELL VOLUME - QX,QZ.    VARP.2620
2441          VOL=.25*( XI1*YK1-XK1*YI1 + XI2*YK2-XK2*YI2 + XI3*YK3-XK3*YI3  VARP.2621
             1          + XI4*YK4-XK4*YI4 )                             VARP.2622
2460          QX1=Q1*(YK1+YI1)                                         VARP.2623
2462          QX2=Q2*(YK2-YI2)                                         VARP.2624
2465          QX3=Q3*(YI3+YK3)                                         VARP.2625
```

```
2467          QX4=Q4*(YK4-YI4)                                          VARP.2626
2472          QK1=Q1*(XI1+XK1)                                          VARP.2627
2474          QK2=Q2*(XK2-XI2)                                          VARP.2628
2477          QK3=Q3*(XI3+XK3)                                          VARP.2629
2501          QK4=Q4*(XK4-XI4)                                          VARP.2630
2504          QX=.5*(QX1+QX2-QX3-QX4)                                   VARP.2631
2511          QZ=.5*(QK3+QK4-QK1-QK2)                                   VARP.2632
2515          IF(QX.EQ.0.0 .AND. QZ .EQ.0.0) GO TO 4285                 VARP.2633
2524          QRS=1./(QX*QX+QZ*QZ)                                      VARP.2634
2526          GO TO 4290                                                VARP.2635
2527     4285 QRS=0.0                                                   VARP.2636
        C ***** THIS SECTION CALCULATES SHIFTED VERTEX POSITION.        VARP.2637
2532     4290 DELX= (QO-QBR) *VOL*QX*QRS                                VARP.2638
2535          DELZ=(QO-QBR)*VOL*QZ*QRS                                  VARP.2639
2537          X4= X4+DELX                                               VARP.2640
2541          Y4= Y4+DELY                                               VARP.2641
        C ***** THIS SECTION PLOTS THE CONTOUR SHIFT ON FILM.           VARP.2642
2543          FX2=XFSP + (X4-FXL)*XCONV                                 VARP.2643
2547          FY2=YFSP + (Y4-FYB)*YCONV                                 VARP.2644
2553          CALL FCTEST                                               VARP.2645
2554          IF( L.GT.1 .AND. L.LT.NCTP ) GO TO 4340                   VARP.2646
2564          IFOPT=6                                                   VARP.2647
2565          CALL FLMOPT                                               VARP.2648
2566          IFOPT=7                                                   VARP.2649
2567          IF( L.EQ.NCTP ) IFOPT=8                                   VARP.2650
2572          CALL FLMOPT                                               VARP.2651
2573          CALL FLMPNT                                               VARP.2652
2574          IFOPT=3                                                   VARP.2653
2575          CALL FLMOPT                                               VARP.2654
2576     4340 IF( IX.EQ.3 ) GO TO 4360                                  VARP.2655
        C ***** THIS SECTION PLOTS LEFT CONTOUR SEGMENT.                VARP.2656
2600          IF( IF3.EQ.IF4 ) GO TO 4350                               VARP.2657
2603          FX3=FFX3(K)                                               VARP.2658
2604          FY3=FFY3(K)                                               VARP.2659
2606          IFLIN=3                                                   VARP.2660
2607          CALL FLMLIN                                               VARP.2661
        C ***** THIS SECTION PLOTS BOTTOM CONTOUR SEGMENT.              VARP.2662
2610     4350 IF( IF1.EQ.IF4 ) GO TO 4380                               VARP.2663
2612          IF( IY.LT.4 ) GO TO 4380                                  VARP.2664
2615          IFLIN=6                                                   VARP.2665
2616          CALL FLMLIN                                               VARP.2666
2617          GO TO 4380                                                VARP.2667
2620     4360 IF( IY.GT.3 ) GO TO 4350                                  VARP.2668
2625     4380 FFX3(K)=FX2                                               VARP.2669
2626          FFY3(K)=FY2                                               VARP.2670
2630          FX1=FX2                                                   VARP.2671
2630          FY1=FY2                                                   VARP.2672
2632     4398 CONTINUE                                                  VARP.2673
2632     4399 CONTINUE                                                  VARP.2674
2642     4499 CONTINUE                                                  VARP.2675
2644          XT=.02                                                    VARP.2676
2646          DO 4519 LF=1,NCTP,2                                       VARP.2677
2647          YT=YT-.02                                                 VARP.2678
2651          ENCODE(30,9090,IFLB) QCON(LF),QCON(LF+1)                  VARP.2679
2664          CALL FLMLAB                                               VARP.2680
2665     4519 CONTINUE                                                  VARP.2681
```

```
VRIO
2670  4798 IF(M.LE.MMAX) CALL FLMADV                                    VARP.2682
2674  4799 CONTINUE                                                     VARP.2683
2677       GO TO KCON,( 4035,4045,4055 )                                VARP.2684
       C ***** FORMATS ***** FORMATS ***** FORMATS.                     VARP.2685
       9000 FORMAT(7H TIME =,1PE11.4,7H NCYC =,I5)                      VARP.2686
       9010 FORMAT(15H PARTICLE PLOT )                                  VARP.2687
       9020 FORMAT(6H ZONES)                                            VARP.2688
       9030 FORMAT(18H VELOCITY VECTORS )                               VARP.2689
       9040 FORMAT(30H       PRESSURE CONTOURS        )                 VARP.2690
       9050 FORMAT(30H       U VELOCITY CONTOURS      )                 VARP.2691
       9060 FORMAT(30H       W VELOCITY CONTOURS      )                 VARP.2692
       9070 FORMAT(4HQMAX,1PE11.4,4HQMIN,E11.4)                         VARP.2693
       9080 FORMAT(15HQ(K)=QMN+K*DQ  ,4H DQ=,1PE11.4)                   VARP.2694
       9090 FORMAT(3H Q=,1PE12.5,3H W=,E12.5)                           VARP.2695
       9100 FORMAT(1H ,21H TIME IN FILM PLOT = ,1PE12.5)                VARP.2696
       9110 FORMAT(30HJ PLANE CONST.,I-HORZ.,K-VERT.)                   VARP.2697
       9120 FORMAT(30HK PLANE CONST.,I-HORZ.,J-VERT.)                   VARP.2698
       9130 FORMAT(30HI PLANE CONST.,J-HORZ.,K-VERT.)                   VARP.2699
       9140 FORMAT(9HJ PLANE =,I3,3X,15HI-HORZ.,K-VERT.)                VARP.2700
       9150 FORMAT(9HK PLANE =,I3,3X,15HI-HORZ.,J-VERT.)                VARP.2701
       9160 FORMAT(9HI PLANE =,I3,3X,15HJ-HORZ.,K-VERT.)                VARP.2702
       9170 FORMAT(30H       V VELOCITY CONTOURS      )                 VARP.2703
2702  99999 CALL SECOND(T2)                                             VARP.2704
2704       T3=T2-T1                                                     VARP.2705
2706       WRITE(6,9100) T3                                             VARP.2706
2714       IGOP(1)=IG1                                                  VARP.2707
2715       IGOP(2)=IG2                                                  VARP.2708
2717       IGOP(3)=IG3                                                  VARP.2709
2720       IGOP(4)=IG4                                                  VARP.2710
2722       IGOP(5)=IG5                                                  VARP.2711
2724       RETURN                                                       VARP.2712
2724       END                                                          VARP.2713
```

SUBPROGRAM LENGTH

03561

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

1	= 002730	2	= 002732	3	= 002752	4	= 002761	5	= 002766	6	= 003007
7	= 003020	100	= 000025	103	= 000033	106	= 000045	107	= 000064	110	= 000075
120	= 000116	190	= 000154	199	= 000154	200	= 000164	203	= 000201	206	= 000213
207	= 000232	210	= 000243	290	= 000276	299	= 000276	300	= 000301	800	= 000307
900	= 000320	1000	= 000335	1020	= 000343	1030	= 000351	1040	= 000362	1042	= 000370
1050	= 000401	1052	= 000403	1060	= 000413	1100	= 000442	1120	= 000504	1140	= 000510
1160	= 000514	1180	= 000517	2000	= 000542	2020	= 000554	2022	= 000562	2030	= 000577
2032	= 000605	2040	= 000622	2100	= 000640	2120	= 000704	2140	= 000725	2160	= 000746
2180	= 000767	2190	= 001034	2200	= 001040	3000	= 001062	3050	= 001103	3055	= 001132
3059	= 001132	3060	= 001135	3065	= 001164	3069	= 001164	3070	= 001167	3075	= 001216
3079	= 001216	3100	= 001221	3200	= 001247	3220	= 001327	3240	= 001343	3260	= 001357
3280	= 001372	4000	= 001437	4030	= 001460	4035	= 001516	4039	= 001516	4040	= 001521
4045	= 001557	4049	= 001557	4050	= 001562	4055	= 001620	4059	= 001620	4080	= 001623
4082	= 001651	4084	= 001660	4086	= 001667	4088	= 001676	4090	= 001704	4100	= 001710

```
      SUBROUTINE VRTAP                                                  VARP.2714
      LARGE PTM(30000)                                                  VE3.220
      INTEGER F,ERF,VDI,VDO,BUFL,PYML,PXML,CPRT,VNTP,VTP,PF,TD,TD1,TD2, INREAL.2
     1 PRFLM,CQF,SHIFT,OBLOC,OBTYP,FI,FO,FB,EM,OBTYPL                   INREAL.3
      REAL NU                                                           INREAL.4
C                                                                       INREAL.5
      DIMENSION U(1),V(1),W(1),P(1),CQ(1),CQF(1)                        DIMEN.2
     1 ,VOL(1)                                                          DIMEN.3
      DIMENSION SPEC(10),TSP(10)                                        VE3.1
      DIMENSION IA(1),QCON(1)                                           DIMEN.6
C                                                                       DIMEN.7
      COMMON/VBCOM/A(36000)                                             VE3.2
      COMMON/CONCEN/ NSPS(10),PMAXWS(10),PMINWS(10),BKGS(10),FACS(10)   VE3.3
     1 ,IPEDII(10),IPEL(10),IPES(10),PMIN(10)                           VE3.4
      COMMON /AMIX/ C(600)                                              COMM.5
      COMMON/EL/FO(20),FB(8),EM                                         COMM.6
      COMMON/FLS/FI(9)                                                  COMM.7
      COMMON/VRCON/ALX,ALY,ALZ,BO,BUFL,CEP,DT,DX(80),DXC,DY(80),DYC,    VE3.5
     1 DZ(60),DZC,EM0,EM6,ERF,GPX,GPY,GPZ,GX,GY,GZ,IBP1,IBP2,IBR,IBRED, VE3.6
     2 IBRIN,IBRST,IGOP(6),IJKMX,IJKNC,IJK2,IJP1K,IPL,IPR,ITER,I2JK2,   COMM.10
     * IKP2,IPARTF,IPLTF,ISFLG,RIBJBKB,LPR,DROU,                         COMM.11
     *IJKCON,ICON,JCON,                                                 COMM.12
     3 I2K2,JBP1,JBP2,JBR,JBRED,JBRIN,JBRST,JNC,JPA,JPF,J2NC,KBP1,KBP2, COMM.13
     4 KBR,KBRED,KBRIN,KBRST,KEP,KNC,KPB,KPT,KWA,KWB,KWF,KWL,KWR,KWT,   COMM.14
     5 K2NC,LABEL(8),LNPART,LNVRCM,LPARTC,NCTP,NCVTP(6),NCYC,NGOP,NOPI, COMM.15
     *NOCONC,                                                           COMM.16
     * OBLOC(1200),OBTYP(1200),NOB,NU,                                  VE3.7
     6 NOVP,NOVS,NPMX,QPRT,NTPE,NWPC,NWPP,PDX,PDY,PDZ,PI,PRFLM,RDT,     COMM.18
     7 RDX(80),RDY(80),RDZ(80),RRAN(20),SIGMA,TD,TD1,TD2,TFIN,TIMET,TTP,VE3.8
     8 TPL,TPLT,TPR,TPRT,TTD,TWTP,TWTD,UI,UPI,VDI,VDO,VI,VNTP(6),VPI,   COMM.20
     9 VTP,WI,WPI,WRAN(20),X(80),XPCEN,Y(80),YPCEN,Z(80),ZPCEN          VE3.9
      COMMON /MODSET/ SCFLM(200)                                        COMM.22
      COMMON /SAFLM/ XCONV,ZCONV,FX1,FX2,FX3,FX4,FY1,FY2,FY3,FY4,       COMM.23
     1 FXL,FYB,PXL,PYB,TFLD(3),FFX3(75),FFY3(75)                        COMM.24
     2 ,ITCYC(3),XFC1(4),XFC2(4),YFC1(4),YFC2(4),PBTIM(2),XML,YML,ZML   COMM.25
     3 ,IFMID,IETEST,IPLOPT,IPM(20),IPMM,IPPP,IPROJ,JPM(20),JPMM,JPPP,  COMM.26
     4 KPM(20),KPMM,KPPP,NEC,NMC,NPS,NPV,PEC(10),PMC(10),YCONV          COMM.27
     5 ,IFLAB,IFLIN,IFPNT,IFOPT,NCH,XFSP,XT,YFSP,YT                     COMM.28
      COMMON /ZLAST/ ZLAS                                               COMM.29
C                                                                       COMM.30
      EQUIVALENCE (A(1),U),(A(2),V),(A(3),W),(A(4),P),(A(5),CQ),        EQUIV.2
     1 (A(6),CQF), (A(7),VOL)                                           EQUIV.3
      EQUIVALENCE (C(1),IA,QCON)                                        EQUIV.7
C                                                                       EQUIV.8
C ***** THIS SUBROUTINE VRTAP HANDLES BOTH MAG TAPE READ AND WRITE     VARP.2720
     1 FORMAT(1H ,19H TAPE FILE NUMBER =,I4,9H  TIMET =,1PE12.5,16H  CYCLVARP.2721
     1E NUMBER =,I6)                                                    VARP.2722
     2 FORMAT(7HTIMET =,1PE11.4,7H NCYC =,I5)                           VARP.2723
   50 IF( IBR.EQ.0 ) GO TO 500                                          VARP.2724
C ***** THIS SECTION , 100-500 , WRITES MAG TAPE FILES                 VARP.2725
  100 ITW=7                                                             VARP.2726
      WRITE(ITW) BUFL                                                   VARP.2727
      WRITE(ITW) (A(I),I=1,BUFL)                                        VARP.2728
      WRITE(ITW) (PTM(I),I=1,LPARTC)                                    VE3.221
      WRITE(VDO,1) TD,TIMET,NCYC                                        VARP.2729
      ENCODE(30,2,XIDFM) TIMET,NCYC                                     VARP.2730
```

```
 51          GO TO 200                                                    VARP.2731
 52      150 TD=0                                                         VARP.2732
 53          END FILE ITW                                                 VARP.2733
 55          REWIND 7                                                     VARP.2734
 57      200 GO TO 999                                                    VARP.2735
       C ***** THIS SECTION , 500-600 , READS MAG TAPE FILES              VARP.2736
 60      500 REWIND 8                                                     VARP.2737
 62          TD1=IBR                                                      VARP.2738
 63          ITPFL=KBR                                                    VARP.2739
 65      550 PRINT 603, ITPFL ,TD                                         VARP.2740
         603 FORMAT( 1H ,6HITPFL=,I2, 5X,3HTD=,I2)                        VARP.2741
 75          READ(8) BUFL                                                 VARP.2742
102          READ(8) (A(I),I=1,BUFL)                                      VARP.2743
110          READ(8) (PIM(I),I=1,LPARIC)                                  VE3.222
117          IT=ITPFL-TD                                                  VARP.2744
121          IF( IT ) 600,575,550                                         VARP.2745
123      575 WRITE(VDD,1) TD,TIMET,NCYC                                   VARP.2746
135          ENCODE(30,2,XIDFM) TIMET,NCYC                                VARP.2747
147          CALL FILMCO                                                  VARP.2748
150          IPLTF=0                                                      VARP.2749
151          DROU= 90.                                                    VE3.223
153          CALL VRID                                                    VARP.2750
154          TFIN= TFIN+DT                                                VE3.224
156          TPLT= TPLT+DT                                                VE3.225
157          TWID= 3.0E+06                                                VE3.226
160          ITER=0                                                       VARP.2751
161          GO TO 999                                                    VARP.2752
162      600 WRITE(VDD,601)                                               VARP.2753
         601 FORMAT(1H ,58H *** ERROR 004 - TAPE RESTART FILE NUMBER IS INCORRE VARP.2754
            1CT ***)                                                      VARP.2755
166          ERF=1                                                        VARP.2756
167      999 RETURN                                                       VARP.2757
170          END                                                          VARP.2758
```

SUBPROGRAM LENGTH

00301

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

```
1      - 000174   2      - 000205   50     - 000002   100    - 000003   150    - 000053   200    - 000060
500    - 000061   550    - 000066   575    - 000124   600    - 000163   601    - 000224   603    - 000215
999    - 000170
```

BLOCK NAMES AND LENGTHS

```
  *LCMR- 0072460
VRCOM  - 106240/01  CONCEN - 000132/02  AMIX   - 001130/03  FL     - 000035/04  FLS    - 000011/05  VRCON  - 006352/06
MODSET - 000310/07  SAFLM  - 000447/08  ZLAST  - 000001/11
```

VARIABLE ASSIGNMENTS

```
A      - 000000/01  BKGS   - 000036/02  BUFL   - 000004/06  C      - 000000/03  CPRT   - 000244   CQ     - 000004/01
CQF    - 000005/01  DROU   - 000435/06  DT     - 000006/06  DX     - 000007/06  DY     - 000130/06  DZ     - 000251/06
ERF    - 000374/06  F      - 000241     FACS   - 000050/02  FB     - 000024/04  FFX3   - 000021/08  FFY3   - 000134/08
```

ENVIRONMENTAL PROTECTION AGENCY
Technical Publications Branch
Office of Administration
Research Triangle Park, N.C. 27711

OFFICIAL BUSINESS

AN EQUAL OPPORTUNITY EMPLOYER

POSTAGE AND FEES PAID
ENVIRONMENTAL PROTECTION AGENCY
EPA - 335

U.S. MAIL

Return this sheet if you do NOT wish to receive this material ☐, or if change of address is needed ☐. (Indicate change, including ZIP code.)