309
5-7-75

Dwr. 1313

LA-5812

UC-32 and UC-34
Reporting Date: November 1974
Issued: March 1975

RICE:
A Computer Program for Multicomponent Chemically Reactive Flows at All Speeds

by

W. C. Rivard
O. A. Farmer
T. D. Butler

MASTER

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

UNITED STATES
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
CONTRACT W-7405-ENG. 36

Work partially supported by the Air Force Weapons Laboratory
under Project Orders AFWL 74-032 and 75-076.

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22151
Price: Printed Copy $5.45 Microfiche $2.25

CONTENTS

RICE: A COMPUTER PROGRAM FOR MULTICOMPONENT

CHEMICALLY REACTIVE FLOWS AT ALL SPEEDS

by

W. C. Rivard
O. A. Farmer
T. D. Butler

ABSTRACT

The fluid dynamics of chemically reactive mixtures are calculated at arbitrary flow speeds with the RICE program. The dynamics are governed by the two-dimensional, time-dependent Navier-Stokes equations together with the species transport equations and the mass-action rate equations for the chemical reactions. The mass and momentum equations for the mixture are solved implicitly by the ICE technique. The equations for total energy and species transport are solved explicitly while the chemical rate equations are solved implicitly with a time step that may be a submultiple of the hydrodynamic time step. Application is made to continuous wave HF chemical lasers to compute the supersonic mixing and chemical reactions that take place in the lasing cavity.

I. INTRODUCTION

The RICE computer program has been written to study the two-dimensional dynamics of multicomponent flows with chemical reactions. The program has been applied principally to investigate the species mixing and chemical reactions that take place in the cavities of continuous flow chemical lasers. In this application, supersonic streams of reactants, which may either be impinging or flowing parallel to one another, mix under the action of molecular and turbulent diffusion and react exothermally to produce vibrationally excited species that serve as the lasing medium. The RICE program has also been applied to incompressible flows, to deflagration and detonation waves, and to the turbulent mixing of supersonic streams.[1]

RICE solves the full, two-dimensional Navier-Stokes equations by the Implicit Continuous-fluid Eulerian (ICE) solution method.[2] The implicit solution procedure for the mass and momentum equations for the mixture uses a different technique for the pressure iteration than is described in Ref. 2. In place of solving a Poisson equation for the pressure field, a relaxation technique, devised by Chorin[3] and modified by Hirt and Cook,[4] is used. This procedure provides a more convenient treatment for the boundary conditions. An explicit solution of the total energy equation follows the solution of the mass and momentum equations. The species transport equations use Fick's law[5] to model the effects of molecular diffusion. Separate transport equations are solved explicitly for each species. The effects of nonequilibrium chemical reactions are described through an implicit solution of the mass-action rate equations with generalized Arrhenius kinetics. The total energy equation includes the effects of the energy flux due to species diffusion and of the heats of formation from the chemical reactions.

The differential equations that govern the dynamics are given in Sec. II. The finite difference equations and the solution method are described in Sec. III. The initial and boundary conditions and numerical stability are also discussed in this

BLANK PAGE

section. A method that provides a considerable increase in accuracy for the same computational mesh and significantly decreases the computational time is described in Ref. 6.

The computer program is described in Sec. IV. Card by card descriptions of the input data are given. The FORTRAN symbols, flow diagrams, and FORTRAN listing are given in Appendixes A-C.

A comprehensive example problem is solved in Sec. V. The fluid dynamics, species mixing, and chemical reactions that take place in the cavity of a continuous flow chemical laser are calculated to illustrate the features of the code. The input data is prepared in detail and samples of the various printed and plotted output are shown.

II. THE MATHEMATICAL MODEL

The differential equations that govern the fluid dynamics, species mixing, and chemical reactions are presented in this section. The equations are written in cylindrical coordinates with axial symmetry and zero azimuthal velocity. The equations for plane coordinates are obtained by setting the radial distance r equal to unity and the normal stress component $\sigma_{\phi\phi}$ equal to zero.

A. Mixture Equations

The motion of the fluid mixture is governed by the equations for mass, momentum, and energy conservation. These are the continuity equation,

$$\frac{\partial\rho}{\partial t} + \frac{1}{r}\frac{\partial\rho v r}{\partial r} + \frac{\partial\rho u}{\partial z} = 0, \tag{2.1}$$

the usual Navier-Stokes equations,

$$\frac{\partial\rho v}{\partial t} + \frac{1}{r}\frac{\partial\rho v^2 r}{\partial r} + \frac{\partial\rho u v}{\partial z} = -\frac{\partial p}{\partial r} + \frac{1}{r}\frac{\partial\sigma_{rr} r}{\partial r} + \frac{\partial\sigma_{rz}}{\partial z} - \frac{\sigma_{\phi\phi}}{r} \tag{2.2}$$

$$\frac{\partial\rho u}{\partial t} + \frac{1}{r}\frac{\partial\rho u v r}{\partial r} + \frac{\partial\rho u^2}{\partial z} = -\frac{\partial p}{\partial z} + \frac{1}{r}\frac{\partial\sigma_{rz} r}{\partial r} + \frac{\partial\sigma_{zz}}{\partial z}, \tag{2.3}$$

and the energy equation,

$$\frac{\partial\rho E}{\partial t} + \frac{1}{r}\frac{\partial\rho E v r}{\partial r} + \frac{\partial\rho E u}{\partial z} = -\frac{1}{r}\frac{\partial p v r}{\partial r} - \frac{\partial p u}{\partial z} + \frac{1}{r}\frac{\partial}{\partial r}\left(\sigma_{rr} v r + \sigma_{rz} u r\right) + \frac{\partial}{\partial z}\left(\sigma_{rz} v + \sigma_{zz} u\right) - \frac{1}{r}\frac{\partial r q_r}{\partial r} - \frac{\partial q_z}{\partial z} + \dot{q}_c . \tag{2.4}$$

In these equations, ρ is the density; v and u are the velocity components in the r and z directions respectively; p is the pressure; σ_{rr}, σ_{rz}, $\sigma_{\phi\phi}$, and σ_{zz} are the components of viscous stress; E is the specific total energy given by

$$E = 1/2\ (u^2 + v^2) + I \tag{2.5}$$

where I is the specific internal energy; q_r and q_z are the r and z components of energy flux due to heat conduction and species diffusion; and $\dot{q}_c$ is the rate of energy release from chemical reactions.

The constitutive equation for a Newtonian fluid relates the viscous stress tensor to the rate of strain tensor:

$$\begin{aligned}
\sigma_{rr} &= 2\mu\frac{\partial v}{\partial r} + \lambda\left(\frac{1}{r}\frac{\partial v r}{\partial r} + \frac{\partial u}{\partial z}\right) \\
\sigma_{rz} &= \mu\left(\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z}\right) \\
\sigma_{zz} &= 2\mu\frac{\partial u}{\partial z} + \lambda\left(\frac{1}{r}\frac{\partial v r}{\partial r} + \frac{\partial u}{\partial z}\right) \\
\sigma_{\phi\phi} &= 2\mu\frac{v}{r} + \lambda\left(\frac{1}{r}\frac{\partial v r}{\partial r} + \frac{\partial u}{\partial z}\right),
\end{aligned} \tag{2.6}$$

where μ and λ are the first and second coefficients of viscosity.

The energy flux components q_r and q_z are given by

$$q_r = -\kappa\frac{\partial T}{\partial r} - \sum_k \rho h_k D_k \frac{\partial(\rho_k/\rho)}{\partial r},$$

and (2.7)

$$q_z = -\kappa\frac{\partial T}{\partial z} - \sum_k \rho h_k D_k \frac{\partial(\rho_k/\rho)}{\partial z},$$

where κ is the thermal conductivity of the mixture, T is the temperature, h_k is the specific enthalpy of species k, D_k is the effective binary diffusion coefficient for species k into the mixture, and ρ_k is the density of species k. The first term in Eq. (2.7) is the energy flux due to thermal conductivity, while the second term is the flux due to the interdiffusion of the species. For this latter term, Fick's law is used to describe the effects of ordinary diffusion which is the only mixing mechanism considered. Other mixing mechanisms associated with pressure and temperature gradients are generally small compared to ordinary diffusion for continuous flow chemical laser systems.

The rate of energy release from chemical reactions $\dot{q}_c$ is defined in Sec. II.C.

The caloric equation of state of the mixture that expresses the pressure as a function of the density and specific internal energy is arbitrary. In this report the perfect gas relation,

$$p = (\gamma-1)\,\rho I, \tag{2.8}$$

is used, where γ is the specific heat ratio $\gamma = C_p/C_v$. The specific heats of the fluid mixture at constant pressure and constant volume are given by

$$C_p = \rho^{-1} \sum_k (C_p)_k\, \rho_k$$

and (2.9)

$$C_v = \rho^{-1} \sum_k (C_v)_k\, \rho_k$$

where $(C_p)_k$ and $(C_v)_k$ are the constant specific heats for species k at constant pressure and constant volume, respectively.

B. Species Transport Equations

The motions of the individual species are determined through the species transport equations:

$$\frac{\partial \rho_k}{\partial t} + \frac{1}{r}\frac{\partial \rho_k v r}{\partial r} + \frac{\partial \rho_k u}{\partial z} = \frac{1}{r}\frac{\partial}{\partial r}\left[\rho D_k r \frac{\partial}{\partial r}(\rho_k/\rho)\right]$$

$$+ \frac{\partial}{\partial z}\left[\rho D_k \frac{\partial}{\partial z}(\rho_k/\rho)\right] + (\dot{\rho}_k)_c\ , \tag{2.10}$$

where $(\dot{\rho}_k)_c$ is the rate of change of the density of species k from chemical reactions and is defined in Sec. II.C. As in Eq. (2.7), ordinary diffusion, described by Fick's law, is the only mixing mechanism considered.

Mass conservation requires that the sum of the species densities should equal the density of the mixture, i.e.,

$$\rho = \sum_k \rho_k . \tag{2.11}$$

If we sum Eq. (2.10) over all the species and use Eq. (2.11) with the mixture continuity equation, Eq. (2.1), we obtain the result that

$$\sum_k \left\{\frac{1}{r}\frac{\partial}{\partial r}\left[\rho D_k r \frac{\partial}{\partial r}(\rho_k/\rho)\right] + \frac{\partial}{\partial z}\left[\rho D_k \frac{\partial}{\partial z}(\rho_k/\rho)\right]\right\} = 0, \tag{2.12}$$

since mass is neither created nor destroyed by chemistry. In other words, Eq. (2.12) states that molecular diffusion alone produces no net mass transfer, hence no change in the mixture density. Equation (2.12) is identically satisfied when D_k is the same for all species. In general, however, this is not the case and Eq. (2.12) is viewed as a constraint on the diffusion coefficients.

C. Chemistry Equations

The equations that govern the finite rate chemical reactions are described in this section. A chemical reaction is written as

$$a_1X_1 + a_2X_2 + \cdots + a_kX_k \rightleftharpoons b_1X_1 + b_2X_2 + \cdots + b_kX_k ,$$

$$\text{e.g., } F + H_2 \rightarrow HF + H, \tag{2.13}$$

where X_k represents the chemical species and a_k and b_k are the dimensionless stoichiometric coefficients, some of which may be zero. In the example, X_1 through X_4 correspond to F, H_2, HF, and H respectively, while

$$a_k = \left[1,\ 1,\ 0,\ 0\right] \text{ and } b_k = \left[0,\ 0,\ 1,\ 1\right] .$$

The rates at which species concentrations are changing during the reaction are proportional to one another according to the law of definite proportions. For Eq. (2.13), the law of definite proportions is written as

$$\frac{\dot{\rho}_1}{M_1(b_1 - a_1)} = \frac{\dot{\rho}_2}{M_2(b_2 - a_2)} = \frac{\dot{\rho}_k}{M_k(b_k - a_k)}\ , \tag{2.14}$$

where M_k is the molecular weight of species k. Equation (2.14) provides a natural definition for the reaction progress variable ω,

$$\dot{\omega} \equiv \frac{\dot{\rho}_k}{M_k(b_k - a_k)} \quad . \tag{2.15}$$

The reaction rate is defined as the rate of change of ω. The dependence of the reaction rate upon the species concentrations and thermodynamic state is expressed by the dilute-gas or mass-action rate so that

$$\dot{\omega} = K_f \prod_k (\rho_k/M_k)^{a_k} - K_b \prod_k (\rho_k/M_k)^{b_k}, \tag{2.16}$$

where K_f and K_b are the rate multipliers for the forward and backward reactions. The rate multipliers have the form

$$K_f \text{ or } K_b = CT^{\zeta} e^{-E^*/T} \tag{2.17}$$

with constants C, ζ, and E^* which is the ratio of the activation energy to the universal gas constant R. The constants are generally different for the forward and backward reactions. With $\zeta = 0$, Eq. (2.17) reduces to the Arrenhius form.

The rate of energy release $\dot{q}_c$ from a chemical reaction is given by

$$\dot{q}_c = q\dot{\omega}, \tag{2.18}$$

where q is the heat of reaction which is positive for exothermic reactions and negative for endothermic reactions. The rate of change of the density of species k from a chemical reaction is obtained from Eqs. (2.15) and (2.16),

$$(\dot{\rho}_k)_c = M_k\,(b_k - a_k)\dot{\omega}. \tag{2.19}$$

In the presence of several chemical reactions, the net rate of energy release and the net rate of change of species densities are simply the sum of the rates for each reaction.

III. THE FINITE DIFFERENCE EQUATIONS AND NUMERICAL METHOD

The purpose of this section is to define the finite difference equations that are used to approximate the differential equations presented in Sec. II and to describe the numerical method for their solution. The solution method is based on the ICE technique. It is quite general and permits the solution of a variety of different flows ranging from very compressible to completely incompressible. For incompressible flows the method reduces to the Simplified Marker and Cell Method.[7]

Equations (2.1) to (2.4) are written in a more general form than those in Ref. 2 in order to incorporate variable coefficients of viscosity and thermal conductivity and to include the additional effects of species diffusion and chemical reactions in the energy equation. The method of solution is altered slightly to include these effects and to incorporate a different iteration procedure for the pressure and velocity field that simplifies the inclusion of various boundary conditions. This new iteration scheme is equivalent to that in Ref. 2.

The flow region of interest is divided into a large number of finite-size, space-fixed, rectangular zones called cells. Collectively, these cells form the computing mesh over which the finite difference equations are to be solved. Figure 1 shows a typical computational cell and the locations of

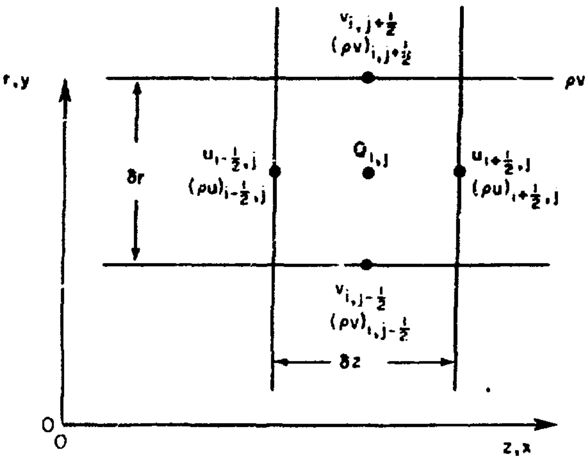


Fig. 1. A typical computational cell and the locations of the variables that appear in the finite difference equations.

the variables that appear in the finite difference equations.

The components of velocity and momentum density are centered on the boundaries between cells as indicated, and $Q_{i,j}$ represents cell-centered quantities such as ρ, ρE, c^2, $p(\bar{p})$, I, μ, λ, κ, D_k, and ρ_k where c is the adiabatic sound speed and $\bar{p}$ is a pressure that is defined later.

The solution is accomplished by algebraically solving the finite difference equations over a time step δt. The solution at time $t = \delta t$ is calculated using the specified initial conditions at time $t = 0$. In general, the solution at $t = (n + 1)\delta t$ is calculated using the solution at $t = n\delta t$.

A. Mass and Momentum

The finite difference approximation to the continuity equation for the mixture, Eq. (2.1), is written as

$$\rho^{n+1}_{i,j} = \tilde{\rho}_{i,j} + \theta\delta t\left\{\frac{1}{r_j\delta r}\left[(\rho v)^{n+1}_{i,j-\frac{1}{2}} r_{j-\frac{1}{2}} - (\rho v)^{n+1}_{i,j+\frac{1}{2}} r_{j+\frac{1}{2}}\right] + \frac{1}{\delta z}\left[(\rho u)^{n+1}_{i-\frac{1}{2},j} - (\rho u)^{n+1}_{i+\frac{1}{2},j}\right]\right\} \quad (3.1)$$

where the components of momentum density at time level (n+1) are defined later and $\tilde{\rho}_{i,j}$ is determined from quantities at time level n:

$$\begin{aligned}\tilde{\rho}_{i,j} = \rho^n_{i,j} + \delta t\Bigg[(1-\theta)\left\{\frac{1}{r_j\delta r}\left[(\rho v)^n_{i,j-\frac{1}{2}} r_{j-\frac{1}{2}} - (\rho v)^n_{i,j+\frac{1}{2}} r_{j+\frac{1}{2}}\right] + \frac{1}{\delta z}\left[(\rho u)^n_{i-\frac{1}{2},j} - (\rho u)^n_{i+\frac{1}{2},j}\right]\right\} \\ + \frac{1}{r_j(\delta r)^2}\left[r_{j+\frac{1}{2}}\tau_{i,j+\frac{1}{2}}\left(\rho^n_{i,j+1} - \rho^n_{i,j}\right) - r_{j-\frac{1}{2}}\tau_{i,j-\frac{1}{2}}\left(\rho^n_{i,j} - \rho^n_{i,j-1}\right)\right] \\ + \frac{1}{(\delta z)^2}\left[\tau_{i+\frac{1}{2},j}\left(\rho^n_{i+1,j} - \rho^n_{i,j}\right) - \tau_{i-\frac{1}{2},j}\left(\rho^n_{i,j} - \rho^n_{i-1,j}\right)\right]\Bigg] . \quad (3.2)\end{aligned}$$

In these equations θ is a parameter used to vary the relative time centering of the convection terms. It ranges in value from zero for a purely explicit calculation to 1.0 for a completely time advanced treatment of the convection terms. A value $\theta = 0.5$ is usually chosen for most compressible flow calculations because this value time centers the terms and eliminates first order time errors that arise in the difference procedure.

The diffusion terms in Eq. (3.2), i.e., the terms involving τ, are added to cancel low order diffusional truncation errors. These errors, when left alone, can lead to numerical instability. The τ terms are calculated from the values of τ' given by

$$(\tau')_{i,j+\frac{1}{2}} = -(2\theta - 1)\frac{\delta t}{2}\left\{\left(v^n_{i,j+\frac{1}{2}}\right)^2 + \frac{1}{2}\left[(c^2)^n_{i,j} + (c^2)^n_{i,j+1}\right]\right\} + \frac{\delta r}{8r_{j+\frac{1}{2}}}\left(r_{j+3/2}\, v^n_{i,j+3/2} - r_{j-\frac{1}{2}}\, v^n_{i,j-\frac{1}{2}}\right), \quad (3.3)$$

$$(\tau')_{i+\frac{1}{2},j} = -(2\theta - 1)\frac{\delta t}{2}\left\{\left(u^n_{i+\frac{1}{2},j}\right)^2 + \frac{1}{2}\left[(c^2)^n_{i,j} + (c^2)^n_{i+1,j}\right]\right\} + \frac{\delta z}{8}\left(u^n_{i+3/2,j} - u^n_{i-\frac{1}{2},j}\right) , \quad (3.4)$$

where the square of the adiabatic sound speed is $c^2 = \gamma(\gamma - 1)I$ for a perfect gas. The value of τ is determined from the algebraic sign of τ',

$$\tau_{i,j+\frac{1}{2}} = \begin{cases}(1 + \xi)\ (\tau')_{i,j+\frac{1}{2}} \text{ if } (\tau')_{i,j+\frac{1}{2}} \geq 0 \\ \\ (1 - \xi)\ (\tau')_{i,j+\frac{1}{2}} \text{ if } (\tau')_{i,j+\frac{1}{2}} < 0,\end{cases} \quad (3.5)$$

where ξ is a constant ranging between zero and unity. The other values of τ' and τ are found similarly.

This procedure of variable coefficients of diffusion has proven very successful over a variety of problems tested. It has obvious advantages over the scheme originally proposed for the ICE method because it automatically supplies just the amount of diffusion required to stabilize the computational scheme and applies it only in regions where it is necessary. The extension of this procedure to the full set of field equations is described in Ref. 6.

The finite difference approximations to the momentum equations, Eqs. (2.2) and (2.3), are written as

$$(\rho v)^{n+1}_{i,j+\frac{1}{2}} = (\tilde{\rho v})_{i,j+\frac{1}{2}} + \frac{\phi\delta t}{\delta r}\,\delta\bar{p}_{i,j} \tag{3.6}$$

and

$$(\rho u)^{n+1}_{i+\frac{1}{2},j} = (\tilde{\rho u})_{i+\frac{1}{2},j} + \frac{\phi\delta t}{\delta z}\,\delta\bar{p}_{i,j}\ , \tag{3.7}$$

where the tilde values of momentum density are computed from quantities at time level n, and where $\delta\bar{p}_{i,j}$ is obtained by iterating on the condition that the finite difference approximation to the continuity equation, Eq. (3.1), is satisfied to within a specified deviation from zero. The iteration procedure for determining $\delta\bar{p}_{i,j}$ is described later. The role of ϕ is similar to that of θ in Eq. (3.1). The tilde values of momentum density are given by

In Eqs. (3.8) and (3.9) straightforward centered differences are used for the various terms. Typical terms are given by

$$(\rho v^2)_{i,j} = \frac{1}{4}\,\rho^n_{i,j}\,(v^n_{i,j+\frac{1}{2}} + v^n_{i,j-\frac{1}{2}})^2,$$

$$(\sigma_{rr})_{i,j} = \frac{2}{\delta r}\,\mu^n_{i,j}\,(v^n_{i,j+\frac{1}{2}} - v^n_{i,j-\frac{1}{2}}) + \lambda^n_{i,j}\left[\frac{1}{r_j\delta r}\,(r_{j+\frac{1}{2}}\,v^n_{i,j+\frac{1}{2}} - r_{j-\frac{1}{2}}\,v^n_{i,j-\frac{1}{2}}) + \frac{1}{\delta z}\,(u^n_{i+\frac{1}{2},j} - u^n_{i-\frac{1}{2},j})\right] .$$

$$(\rho uv)_{i-\frac{1}{2},j+\frac{1}{2}} = \frac{1}{16}\,(\rho^n_{i,j} + \rho^n_{i,j+1} + \rho^n_{i-1,j+1} + \rho^n_{i-1,j})\,(u^n_{i-\frac{1}{2},j} + u^n_{i-\frac{1}{2},j+1})\,(v^n_{i,j+\frac{1}{2}} + v^n_{i-1,j+\frac{1}{2}}),$$

$$\begin{aligned}(\tilde{\rho v})_{i,j+\frac{1}{2}} = (\rho v)^n_{i,j+\frac{1}{2}} + \delta t\Bigg[&\frac{1}{r_{j+\frac{1}{2}}\delta r}\Big\{r_j\left[(\rho v)^2_{i,j} - (\sigma_{rr})_{i,j}\right] - r_{j+1}\left[(\rho v^2)_{i,j+1} - (\sigma_{rr})_{i,j+1}\right]\Big\}\\ &+ \frac{1}{\delta z}\left[(\rho uv)_{i-\frac{1}{2},j+\frac{1}{2}} - (\sigma_{rz})_{i-\frac{1}{2},j+\frac{1}{2}} - (\rho uv)_{i+\frac{1}{2},j+\frac{1}{2}} + (\sigma_{rz})_{i+\frac{1}{2},j+\frac{1}{2}}\right] + \frac{1}{\delta r}\,(p^n_{i,j} - p^n_{i,j+1})\\ &- \frac{1}{r_{j+\frac{1}{2}}}\,(\sigma_{\phi\phi})_{i,j+\frac{1}{2}}\Bigg],\end{aligned} \tag{3.8}$$

and

$$\begin{aligned}(\tilde{\rho u})_{i+\frac{1}{2},j} = (\rho u)^n_{i+\frac{1}{2},j} + \delta t\Bigg[&\frac{1}{r_j\delta r}\Big\{r_{j-\frac{1}{2}}\left[(\rho uv)_{i+\frac{1}{2},j-\frac{1}{2}} - (\sigma_{rz})_{i+\frac{1}{2},j-\frac{1}{2}}\right] - r_{j+\frac{1}{2}}\left[(\rho uv)_{i+\frac{1}{2},j+\frac{1}{2}} - (\sigma_{rz})_{i+\frac{1}{2},j+\frac{1}{2}}\right]\Big\}\\ &+ \frac{1}{\delta z}\left[(\rho u^2)_{i,j} - (\sigma_{zz})_{i,j} - (\rho u^2)_{i+1,j} + (\sigma_{zz})_{i+1,j} + p^n_{i,j} - p^n_{i+1,j}\right]\Bigg].\end{aligned} \tag{3.9}$$

$$(\sigma_{rz})_{i-\frac{1}{2},j+\frac{1}{2}} = \frac{1}{4}\,(\mu^n_{i,j} + \mu^n_{i,j+1} + \mu^n_{i-1,j+1} + \mu^n_{i-1,j})\left[\frac{1}{\delta r}\,(u^n_{i-\frac{1}{2},j+1} - u^n_{i-\frac{1}{2},j}) + \frac{1}{\delta z}\,(v^n_{i,j+\frac{1}{2}} - v^n_{i-1,j+\frac{1}{2}})\right],$$

$$(\sigma_{\phi\phi})_{i,j+\frac{1}{2}} = \frac{1}{r_{j+\frac{1}{2}}}\,(\mu^n_{i,j} + \mu^n_{i,j+1})\,v^n_{i,j+\frac{1}{2}} + \frac{1}{4}\,(\lambda^n_{i,j} + \lambda^n_{i,j+1})\left\{\frac{1}{r_{j+\frac{1}{2}}\,\delta r}\left[r_{j+1}\,(v^n_{i,j+3/2} + v^n_{i,j+\frac{1}{2}}) - r_j\,(v^n_{i,j+\frac{1}{2}} + v^n_{i,j-\frac{1}{2}})\right] + \frac{1}{\delta z}\left[(u^n_{i+\frac{1}{2},j} + u^n_{i+\frac{1}{2},j+1}) - (u^n_{i-\frac{1}{2},j} + u^n_{i-\frac{1}{2},j+1})\right]\right\}.$$

$$(\rho u^2)_{i,j} = \frac{1}{4}\,\rho^n_{i,j}\,(u^n_{i+\frac{1}{2},j} + u^n_{i-\frac{1}{2},j})^2.$$

The remaining terms in Eqs. (3.8) and (3.9) are defined similarly.

The viscosity coefficients are variables here rather than constants as they were in the original formulation of the ICE method. This proves very helpful in problems for which the variation of viscosity with temperature is important and, as shown later, for inserting the appropriate viscosity values into the equations to maintain numerical stability.

The solution of Eqs. (3.1), (3.6), and (3.7) for $\rho^{n+1}_{i,j}$, $(\rho v)^{n+1}_{i,j+\frac{1}{2}}$, and $(\rho u)^{n+1}_{i+\frac{1}{2},j}$ is accomplished by iterating on the pressure $\bar{p}_{i,j} = p^n_{i,j} + \delta\bar{p}_{i,j}$. The original ICE method obtained the $\bar{p}$ pressure field by solving a Poisson equation and from this the mass and momentum densities were determined. The iteration procedure in RICE that simultaneously solves for the $\bar{p}$ pressure and mass and momentum densities is desirable for the ease with which different boundary conditions can be incorporated into the computer program.

The criterion for convergence of the iteration procedure is that the continuity equation, Eq. (3.1), is satisfied to within a specified small amount ε, i.e., Eq. (3.1) is written as

$$D = \rho^{n+1}_{i,j} - \tilde{\rho}_{i,j} - \theta\delta t\left\{\frac{1}{r_j\,\delta r}\left[(\rho v)^{n+1}_{i,j-\frac{1}{2}}\,r_{j-\frac{1}{2}} - (\rho v)^{n+1}_{i,j+\frac{1}{2}}\,r_{j+\frac{1}{2}}\right] + \frac{1}{\delta z}\left[(\rho u)^{n+1}_{i-\frac{1}{2},j} - (\rho u)^{n+1}_{i+\frac{1}{2},j}\right]\right\}, \qquad (3.10)$$

and the convergence requirement is $|D| < \varepsilon$. The procedure is begun by estimating the mass and momentum densities at time level (n+1) as the mass density at time level n and the tilde values of momentum density, but the choice is arbitrary.

If the convergence criterion $|D| < \varepsilon$ is satisfied, the quantities in cell (i,j) are not changed. If however, the criterion is not satisfied, the pressure is changed by an amount

$$\delta\bar{p}_{i,j} = -\,\Omega D/\left[1/(c^2)^n_{i,j} + 2\theta\phi\delta t^2\,(\delta r^{-2} + \delta z^{-2})\right], \qquad (3.11)$$

$$\text{i.e., } \bar{p}_{i,j} \leftarrow \bar{p}_{i,j} + \delta\bar{p}_{i,j}, \qquad (3.12)$$

where Ω is a constant over-relaxation factor usually chosen between 1.0 and 1.8. At the start of the iteration $\bar{p}_{i,j} = p^n_{i,j}$. The choice for the value of Ω is problem-dependent to some extent. If the flow is rather compressible with Mach number greater than unity, good predictions of $\delta\bar{p}$ are obtained for $\Omega = 1$. For very small values of the Mach number, Ω greater than 1.5 produces the most rapid convergence.

When the $\bar{p}$ pressure is changed according to Eq. (3.11) the mass and momentum densities are also changed as follows

$$\rho^{n+1}_{i,j} \leftarrow \rho^{n+1}_{i,j} + \delta\bar{p}_{i,j}/(c^2)_{i,j},$$

$$(\rho u)^{n+1}_{i+\frac{1}{2},j} \leftarrow (\rho u)^{n+1}_{i+\frac{1}{2},j} + \frac{\phi\delta t}{\delta z}\,\delta\bar{p}_{i,j},$$

$$(\rho u)^{n+1}_{i-\frac{1}{2},j} \leftarrow (\rho u)^{n+1}_{i-\frac{1}{2},j} - \frac{\phi\delta t}{\delta z}\,\delta\bar{p}_{i,j},$$

$$(\rho v)^{n+1}_{i,j+\frac{1}{2}} \leftarrow (\rho v)^{n+1}_{i,j+\frac{1}{2}} + \frac{\phi\delta t}{\delta r}\,\delta\bar{p}_{i,j},$$

$$(\rho v)^{n+1}_{i,j-\frac{1}{2}} \leftarrow (\rho v)^{n+1}_{i,j-\frac{1}{2}} - \frac{\phi\delta t}{\delta r}\,\delta\bar{p}_{i,j}. \qquad (3.13)$$

This process is repeated in all computing cells until $|D| < \varepsilon$ in every cell of the computing mesh.

When the convergence criterion is satisfied for every cell, the values of the mass and momentum densities are the solution values at time level (n+1). In addition, the $\bar{p}$ pressure field is an intermediate pressure that reflects the density at time level (n+1) and the specific internal energy at time level (n). The velocities are obtained from the mass and momentum densities as

$$u^{n+1}_{i+\frac{1}{2},j} = 2(\rho u)^{n+1}_{i+\frac{1}{2},j} \Big/ (\rho^{n+1}_{i,j} + \rho^{n+1}_{i+1,j}),$$

$$v^{n+1}_{i,j+\frac{1}{2}} = 2(\rho v)^{n+1}_{i,j+\frac{1}{2}} \Big/ (\rho^{n+1}_{i,j} + \rho^{n+1}_{i,j+1}). \qquad (3.14)$$

B. Total Energy

The velocity components of the mixture at time level (n+1) obtained after the iteration procedure are used for the explicit calculation of the total energy density. An intermediate value $(\rho\tilde{E})$ of the total energy density is obtained that neglects the effects of species mixing and chemistry in Eq. (2.4). The finite difference equation for $(\rho\tilde{E})$ is

$$\begin{aligned}(\rho\tilde{E})_{i,j} = (\rho E)^n_{i,j} + \frac{\delta t}{r_j\delta r}\Bigg[r_{j-\frac{1}{2}}\Bigg\{v^{n+1}_{i,j-\frac{1}{2}}\Big[(\rho E)^n_{i,j-\frac{1}{2}} \\ + \bar{p}_{i,j-\frac{1}{2}} - (\sigma_{rr})_{i,j-\frac{1}{2}}\Big] - (\sigma_{rz})_{i,j-\frac{1}{2}}\, u^{n+1}_{i,j-\frac{1}{2}} \\ - (\hat{\kappa})^n_{i,j-\frac{1}{2}}\,(I^n_{i,j} - I^n_{i,j-1})/\delta r\Bigg\} \\ - r_{j+\frac{1}{2}}\Bigg\{v^{n+1}_{i,j+\frac{1}{2}}\Big[(\rho E)^n_{i,j+\frac{1}{2}} + \bar{p}_{i,j+\frac{1}{2}} - (\sigma_{rr})_{i,j+\frac{1}{2}}\Big] \\ - (\sigma_{rz})_{i,j+\frac{1}{2}}\, u^{n+1}_{i,j+\frac{1}{2}} - (\hat{\kappa})^n_{i,j+\frac{1}{2}}\,(I^n_{i,j+1} \\ - I^n_{i,j})/\delta r\Bigg\}\Bigg] + \frac{\delta t}{\delta z}\Bigg[u^{n+1}_{i-\frac{1}{2},j}\Big\{(\rho E)^n_{i-\frac{1}{2},j} + \bar{p}_{i-\frac{1}{2},j} \\ - (\sigma_{zz})_{i-\frac{1}{2},j}\Big] - (\sigma_{rz})_{i-\frac{1}{2},j}\, v^{n+1}_{i-\frac{1}{2},j} \\ - (\hat{\kappa})^n_{i-\frac{1}{2},j}\,(I^n_{i,j} - I^n_{i-1,j})/\delta z \\ - \Bigg\{u^{n+1}_{i+\frac{1}{2},j}\Big[(\rho E)^n_{i+\frac{1}{2},j} + \bar{p}_{i+\frac{1}{2},j} - (\sigma_{zz})_{i+\frac{1}{2},j}\Big] \\ - (\sigma_{rz})_{i+\frac{1}{2},j}\, v^{n+1}_{i+\frac{1}{2},j} - (\hat{\kappa})^n_{i+\frac{1}{2},j}\,(I^n_{i+1,j} \\ - I^n_{i,j})/\delta z\Bigg\}\Bigg]. \qquad (3.15)\end{aligned}$$

In this equation all quantities that are required at spatial locations in the mesh other than where they are defined (recall Fig. 1) are obtained by simple averages, e.g.,

$$(\rho E)^n_{i,j-\frac{1}{2}} = 1/2\Big[(\rho E)^n_{i,j} + (\rho E)^n_{i,j-1}\Big],$$

$$\begin{aligned}(\sigma_{rr})_{i,j-\frac{1}{2}} = \frac{1}{2\delta r}\,(\mu^n_{i,j} + \mu^n_{i,j-1})\,(v^{n+1}_{i,j+\frac{1}{2}} - v^{n+1}_{i,j-\frac{3}{2}}) \\ + 1/4(\lambda^n_{i,j} + \lambda^n_{i,j-1})\Bigg[\frac{1}{r_{j-\frac{1}{2}}\delta r}\,(r_{j+\frac{1}{2}}\, v^{n+1}_{i,j+\frac{1}{2}} \\ - r_{j-\frac{3}{2}}\, v^{n+1}_{i,j-\frac{3}{2}}) + \frac{1}{\delta z}\,(u^{n+1}_{i+\frac{1}{2},j} + u^{n+1}_{i+\frac{1}{2},j-1} \\ - u^{n+1}_{i-\frac{1}{2},j} - u^{n+1}_{i-\frac{1}{2},j-1})\Bigg],\end{aligned}$$

and similarly for the other quantities. The energy flux due to thermal conductivity in Eq. (2.7) has been written in terms of gradients of I rather than T, and κ has been replaced by $\hat{\kappa}$ in Eq. (3.15). This gradient transformation is correct only when C_v is a constant and $\hat{\kappa} = \kappa/C_v$. However, since the thermal conduction term is used for numerical stability in Eq. (3.15) the I-gradient form is retained even when C_v is variable. This form allows for a significantly larger time step than would be possible otherwise. Numerical stability is discussed in detail in Sec. III.F.

A tilde value of the specific internal energy is obtained from the tilde value of the total energy density as

$$\tilde{I}_{i,j} = (\tilde{\rho E})_{i,j}/\rho^{n+1}_{i,j} - 1/8\left[(u^{n+1}_{i+\frac{1}{2},j} + u^{n+1}_{i-\frac{1}{2},j})^2 + (v^{n+1}_{i,j+\frac{1}{2}} + v^{n+1}_{i,j-\frac{1}{2}})^2\right]. \quad (3.16)$$

For the flow of a single fluid, the tilde values of (ρE) and I are in fact the values at time level (n+1). For multicomponent flows the energy flux due to species diffusion is added to ($\tilde{\rho E}$) after solution of the species transport equations and the energy release (or absorption) from chemical reaction is included after solution of the chemistry equations.

C. Species Transport

The solution procedure for the species transport equations is similar to the procedure used for solution of the total energy equation. A tilde value of species density is obtained by solving the finite-difference approximation to Eq. (2.10) neglecting the species diffusion and chemistry terms. These terms are again treated separately and their respective contributions are added to the tilde value to yield the final species density at time level (n+1). This procedure, while not a necessity in the solution, enables each effect to be handled separately and greatly simplifies the programming of the computer code.

The tilde value of the density of species k is

$$(\tilde{\rho}_k)_{i,j} = (\rho_k)^n_{i,j} + \delta t\left[\frac{1}{r_j\delta r}\left\{r_{j-\frac{1}{2}}(\rho_k v)_{i,j-\frac{1}{2}} - r_{j+\frac{1}{2}}(\rho_k v)_{i,j+\frac{1}{2}}\right\} + \frac{1}{\delta z}\left\{(\rho_k u)_{i-\frac{1}{2},j} - (\rho_k u)_{i+\frac{1}{2},j}\right\} + \frac{1}{r_j(\delta r)^2}\left\{r_{j+\frac{1}{2}}\beta_{i,j+\frac{1}{2}}\left[(\rho_k)^n_{i,j+1} - (\rho_k)^n_{i,j}\right] - r_{j-\frac{1}{2}}\beta_{i,j-\frac{1}{2}}\left[(\rho_k)^n_{i,j} - (\rho_k)^n_{i,j-1}\right]\right\} + \frac{1}{(\delta z)^2}\left\{\beta_{i+\frac{1}{2},j}\left[(\rho_k)^n_{i+1,j} - (\rho_k)^n_{i,j}\right] - \beta_{i-\frac{1}{2},j}\left[(\rho_k)^n_{i,j} - (\rho_k)^n_{i-1,j}\right]\right\}\right], \quad (3.17)$$

where interpolated donor cell differencing is used for the convection terms such that

$$(\rho_k v)_{i,j-\frac{1}{2}} = 1/2\left\{(\rho_k)^n_{i,j} + (\rho_k)^n_{i,j-1} - \frac{\delta t}{\delta r}\left[(\rho_k)^n_{i,j} - (\rho_k)^n_{i,j-1}\right] v^{n+1}_{i,j-\frac{1}{2}}\right\} v^{n+1}_{i,j-\frac{1}{2}}$$

and

$$(\rho_k u)_{i-\frac{1}{2},j} = 1/2\left\{(\rho_k)^n_{i,j} + (\rho_k)^n_{i-1,j} - \frac{\delta t}{\delta z}\left[(\rho_k)^n_{i,j} - (\rho_k)^n_{i-1,j}\right] u^{n+1}_{i-\frac{1}{2},j}\right\} u^{n+1}_{i-\frac{1}{2},j} \quad (3.18)$$

and similarly for $(\rho_k v)_{i,j+\frac{1}{2}}$ and $(\rho_k u)_{i+\frac{1}{2},j}$. The diffusion terms in Eq. (3.17), i.e., the terms involving β, are added to cancel low order diffusional truncation errors similar to the τ terms in Eq. (3.2). In like manner, the β terms are calculated from the values of the β' terms given by

$$(\beta')_{i,j+\frac{1}{2}} = \frac{\delta r}{8r_{j+\frac{1}{2}}}\left(r_{j+\frac{3}{2}} v^{n+1}_{i,j+\frac{3}{2}} - r_{j-\frac{1}{2}} v^{n+1}_{i,j-\frac{1}{2}}\right),$$

and

$$(\beta')_{i+\frac{1}{2},j} = \frac{\delta z}{8}\left(u^{n+1}_{i+\frac{3}{2},j} - u^{n+1}_{i-\frac{1}{2},j}\right), \quad (3.19)$$

with

$$\beta_{i,j+\frac{1}{2}} = \begin{cases} (1+\xi)\,(\beta')_{i,j+\frac{1}{2}} & \text{if } (\beta')_{i,j+\frac{1}{2}} \geq 0 \\ (1-\xi)\,(\beta')_{i,j+\frac{1}{2}} & \text{if } (\beta')_{i,j+\frac{1}{2}} < 0, \end{cases} \quad (3.20)$$

where ξ again ranges between zero and unity. The other values of β' and β are found similarly.

To achieve rigid conservation of mass, i.e., to insure that the mixture density equals the sum of the species densities in a computational cell, the tilde density of the last species is calculated as

$$(\tilde{\rho}_{NS})_{i,j} = \rho_{i,j}^{n+1} - \sum_{k=1}^{NS-1} (\tilde{\rho}_k)_{i,j}, \tag{3.21}$$

where NS is the total number of species. Mass conservation is insured when the sum of the species tilde densities equals the mixture density, since species diffusion and chemistry produce no net mass change. This approach is simple to include in the computer program and has produced excellent results.

The effects of species diffusion are calculated in accord with Eq. (2.10) subject to the constraint that there is no net mass transfer as expressed by Eq. (2.12). This constraint is enforced in the solution algorithm by requiring that there be no net mass transfer across any boundary of the computational cell. In finite difference form this yields the conditions that

$$\sum_k (\delta f_k)_{i,j+\frac{1}{2}} = \sum_k (\delta f_k)_{i-\frac{1}{2},j} = \sum_k (\delta f_k)_{i,j-\frac{1}{2}}$$

$$= \sum_k (\delta f_k)_{i+\frac{1}{2},j} = 0, \tag{3.22}$$

where

$$(\delta f_k)_{i,j+\frac{1}{2}} = 1/4 \left[(D_k)_{i,j} + (D_k)_{i,j+1}\right] (\rho_{i,j}^{n+1} + \rho_{i,j+1}^{n+1}) \left\{ \left[(\tilde{\rho}_k)/\rho^{n+1}\right]_{i,j+1} - \left[(\tilde{\rho}_k)/\rho^{n+1}\right]_{i,j} \right\} / \delta r,$$

$$(\delta f_k)_{i+\frac{1}{2},j} = 1/4 \left[(D_k)_{i,j} + (D_k)_{i+1,j}\right] (\rho_{i,j}^{n+1} + \rho_{i+1,j}^{n+1}) \left\{ \left[(\tilde{\rho}_k)/\rho^{n+1}\right]_{i+1,j} - \left[(\tilde{\rho}_k)/\rho^{n+1}\right]_{i,j} \right\} / \delta z, \tag{3.23}$$

for the top and right boundaries of the cell with similar expressions for the bottom and left boundaries. The diffusion coefficient $(D_k)_{i,j}$ is evaluated for low density gases as

$$(D_k)_{i,j} = \alpha_k \, \eta_{i,j},$$

where

$$\alpha_k = (M_H/M_k)^{\frac{1}{2}}, \tag{3.24}$$

and

$$\eta_{i,j} = \eta_o \, (T_{i,j})^{\frac{1}{2}} \Big/ \left\{ \rho_{i,j}^{n+1} \left[(C_p)_{i,j} - (C_v)_{i,j}\right] \right\}$$

α_k is scaled to the molecular weight of atomic hydrogen, $M_H = 1.0$, and η_o is a constant. The coefficient α_k describes the dependence of D_k on the molecular weight of species k. This inverse square root dependence is analogous to the dependence of the mean molecular velocity on the molecular weight at constant temperature in kinetic theory. The specific heats are given by Eq. (2.9) as

$$(C_p)_{i,j} = \sum_k (C_p)_k \, (\tilde{\rho}_k)_{i,j} \Big/ \rho_{i,j}^{n+1}$$

$$(C_v)_{i,j} = \sum_k (C_v)_k \, (\tilde{\rho}_k)_{i,j} \Big/ \rho_{i,j}^{n+1}$$

and the temperature as

$$T_{i,j} = \tilde{I}_{i,j}/(C_v)_{i,j}. \tag{3.25}$$

Equation (3.22) is identically satisfied, as is Eq. (2.12), when D_k has the same value for all species. This is the case, to good approximation, when the species have sufficiently similar molecular weights and the mixture density is sufficiently rarefied.

Generally, however, the diffusion coefficients are different for the different species so that Eq. (3.22) is not identically satisfied. If a summation yields a positive value instead of zero, those elements that are positive are reduced in magnitude. On the other hand, if a summation yields a negative value, those elements that are negative are reduced in magnitude. This is accomplished through a reduction factor so that on the top boundary of the cell

$$(\delta f_k^+)_{i,j+\frac{1}{2}} \leftarrow (\delta f_k^+)_{i,j+\frac{1}{2}} (1 - \Delta/\Delta^+) \text{ for } \Delta \geq 0,$$

or

$$(\delta f_k^-)_{i,j+\frac{1}{2}} \leftarrow (\delta f_k^-)_{i,j+\frac{1}{2}} (1 - \Delta/\Delta^-) \text{ for } \Delta < 0 \quad (3.26)$$

where $(\delta f_k^+)_{i,j+\frac{1}{2}}$ and $(\delta f_k^-)_{i,j+\frac{1}{2}}$ are the positive and negative elements respectively and Δ represents summations, i.e.,

$$\Delta = \sum_k (\delta f_k)_{i,j+\frac{1}{2}},$$

$$\Delta^+ = \sum_k (\delta f_k^+)_{i,j+\frac{1}{2}}, \quad \Delta^- = \sum_k (\delta f_k^-)_{i,j+\frac{1}{2}} .$$

A similar procedure is used for the right, bottom, and left boundaries.

The species tilde densities are updated to include the effects of molecular diffusion described in Eq. (2.10)

$$(\tilde{\rho}_k)_{i,j} \leftarrow (\tilde{\rho}_k)_{i,j} + \delta t \left\{ \frac{1}{r_j \delta r} \left[r_{j+\frac{1}{2}} (\delta f_k)_{i,j+\frac{1}{2}} - r_{j-\frac{1}{2}} (\delta f_k)_{i,j-\frac{1}{2}} \right] + \frac{1}{\delta z} \left[(\delta f_k)_{i+\frac{1}{2},j} - (\delta f_k)_{i-\frac{1}{2},j} \right] \right\} , \quad (3.27)$$

which assures that the mixture density $\rho_{i,j}^{n+1}$ does not change.

The energy flux due to the interdiffusion of the species is described by Eqs. (2.4) and (2.7) and is added to the tilde value of the total energy density as

$$(\rho \tilde{E})_{i,j} \leftarrow (\rho \tilde{E})_{i,j} + \delta t \left\{ \frac{1}{r_j \delta r} \left[r_{j+\frac{1}{2}} \sum_k (C_p)_k T_{i,j+\frac{1}{2}} (\delta f_k)_{i,j+\frac{1}{2}} - r_{j-\frac{1}{2}} \sum_k (C_p)_k T_{i,j-\frac{1}{2}} (\delta f_k)_{i,j-\frac{1}{2}} \right] + \frac{1}{\delta z} \left[\sum_k (C_p)_k T_{i+\frac{1}{2},j} (\delta f_k)_{i+\frac{1}{2},j} - \sum_k (C_p)_k T_{i-\frac{1}{2},j} (\delta f_k)_{i-\frac{1}{2},j} \right] \right\} . \quad (3.28)$$

D. Chemistry

The effects of a finite rate chemical reaction on the species densities are described by Eqs. (2.10), (2.16), and (2.19). To determine the progress of the reaction during the computational time step from $t = n\delta t$ to $t = (n+1)\delta t$, Eq. (2.16) is solved. The net effect of several reactions is simply the sum of the effects of each reaction and is discussed at the end of this section.

Equation (2.16) is solved implicitly for the change in concentration of a particular species, designated as the reference species for the reaction. The reference species is chosen automatically by the program as the species with the lowest concentration, i.e., with minimum value of $|(\tilde{\rho}_k)_{i,j} / M_k(b_k - a_k)|$, and a negative rate of change of concentration. The implicit solution of Eq. (2.16) insures that the concentration of the reference species, and hence the concentrations of the other species, will not be driven negative during the time step δt. In finite difference form Eq. (2.16) is written as

$$(\rho_r)_{i,j}^{n+1}/M_r = \frac{(\tilde{\rho}_r)_{i,j}/M_r + \delta t\, a_r K_b \prod_{k=1}^{NS} \left[(\tilde{\rho}_k)_{i,j}/M_k\right]^{b_k}}{1 + \frac{\delta t\, a_r M_r K_f}{(\tilde{\rho}_r)_{i,j}} \prod_{k=1}^{NS} \left[(\tilde{\rho}_k)_{i,j}/M_k\right]^{a_k}},$$

$$\text{for } \dot{\omega} \geq 0 \quad (3.29)$$

or

$$(\rho_r)_{i,j}^{n+1}/M_r = \frac{(\tilde{\rho}_r)_{i,j}/M_r + \delta t\, b_r K_f \prod_{k=1}^{NS} \left[(\tilde{\rho}_k)_{i,j}/M_k\right]^{a_k}}{1 + \frac{\delta t\, b_r M_r K_b}{(\tilde{\rho}_r)_{i,j}} \prod_{k=1}^{NS} \left[(\tilde{\rho}_k)_{i,j}/M_k\right]^{b_k}},$$

$$\text{for } \dot{\omega} < 0$$

where subscript r refers to the reference species and $\dot{\omega}$ is evaluated explicitly as

$$\dot{\omega} = K_f \prod_{k=1}^{NS} \left[(\tilde{\rho}_k)_{i,j}/M_k\right]^{a_k} - K_b \prod_{k=1}^{NS} \left[(\tilde{\rho}_k)_{i,j}/M_k\right]^{b_k}. \tag{3.30}$$

The selection of the appropriate reference species is accomplished automatically in the computer program for each computational cell at each time step. The rate multipliers are determined from Eq. (2.17) where the temperature is calculated from Eq. (3.25) using the most updated values of $(\tilde{\rho}_k)_{i,j}$ from Eq. (3.27) and $\tilde{I}_{i,j}$ from Eqs. (3.28) and (3.16).

The final values of the total energy density and the species densities are calculated from the increment in the progress variable ω defined through Eq. (2.15). The progress increment is determined from the solution for the reference species, i.e.,

$$\delta\omega = \left[(\rho_r)^{n+1}_{i,j} - (\tilde{\rho}_r)_{i,j}\right] / \left[M_r \; (b_r - a_r)\right]. \tag{3.31}$$

The rate of heat release due to chemistry, the last term in Eq. (2.4), is given by Eq. (2.18). Accordingly, the value of the total energy density at time level (n+1) is

$$(\rho E)^{n+1}_{i,j} = (\tilde{\rho E})_{i,j} + q\delta\omega. \tag{3.32}$$

The rate of change of the species densities due to chemistry, the last term in Eq. (2.10), is given by Eq. (2.19) so that the final values of the species densities are

$$(\rho_k)^{n+1}_{i,j} = (\tilde{\rho}_k)_{i,j} + \delta\omega \; M_k(b_k - a_k). \tag{3.33}$$

When several reactions are occurring simultaneously, a progress increment $\delta\omega_\ell$ is calculated for each reaction. In this case Eqs. (3.32) and (3.33) are replaced by

$$(\rho E)^{n+1}_{i,j} = (\tilde{\rho E})_{i,j} + \sum_{\ell=1}^{NR} q_\ell \; \delta\omega_\ell, \tag{3.34}$$

and

$$(\rho_k)^{n+1}_{i,j} = (\tilde{\rho}_k)_{i,j} + \sum_{\ell=1}^{NR} \delta\omega_\ell \; M_k \; (b_{k\ell} - a_{k\ell}), \tag{3.35}$$

where NR is the total number of reactions.

In certain situations it is necessary to utilize a time step δt_c which is a submultiple of the hydrodynamic time step δt for solution of Eq. (3.29). Such a situation can arise, for example, when a particular species is being consumed in different reactions at different rates or when the characteristic time for the chemistry is fast, but not infinite, compared to the characteristic time for the fluid dynamics. The final solution is obtained by subcycling through Eqs. (3.29) - (3.35) the integral number of times necessary with δt replaced by δt_c.

The computational cycle is completed with the calculation of the final value of the specific internal energy from Eq. (3.16) where $(\rho E)^{n+1}$ is used in place of $(\tilde{\rho E})$ and the final pressure from the equation of state, Eq. (2.8).

$$P^{n+1}_{i,j} = (\gamma^{n+1}_{i,j} - 1) \; \rho^{n+1}_{i,j} \; I^{n+1}_{i,j} , \tag{3.36}$$

where

$$\gamma^{n+1}_{i,j} = (C_p)^{n+1}_{i,j}/(C_v)^{n+1}_{i,j}$$

and the specific heats are given by Eq. (2.9) as

$$(C_p)^{n+1}_{i,j} = \sum_k (C_p)_k \; (\rho_k)^{n+1}_{i,j}/\rho^{n+1}_{i,j}$$

$$(C_v)^{n+1}_{i,j} = \sum_k (C_v)_k \; (\rho_k)^{n+1}_{i,j}/\rho^{n+1}_{i,j}.$$

E. Initial and Boundary Conditions

To begin the solution of the finite difference equations in time, it is necessary to specify a set of initial conditions for each computational cell. Values are required for ρ, u, v, I, and ρ_k, which are the dependent variables in the governing equations. Uniform initial conditions are specified conveniently in the computer program through the input data.

Several different boundary conditions around the perimeter of the computing mesh are programmed:

1) No-slip rigid walls,

2) Free-slip rigid walls,

3) Prescribed input,

4) Continuative output.

For the first two, the rigid walls are also adiabatic or nonconducting. Rigid walls can be specified along any boundary of the computing mesh. The free-slip wall is equivalent to a reflective plane of symmetry. Input can be prescribed anywhere along the left and top boundaries of the mesh with up to two distinct inflow openings on each boundary. Inflow is prescribed along the left boundary by specifying values for ρ, u, v = 0, I, and ρ_k and along the top boundary by specifying values for ρ, u = 0, v, I, and ρ_k. A continuative output boundary can be Prescribed anywhere along the right boundary with up to two distinct openings. The bottom boundary of the computing mesh is either a free-slip or no-slip, adiabatic, rigid wall.

The prescribed boundary conditions are enforced through the use of fictitious cells that surround the computing mesh as shown in Fig. 2. For rigid walls the tangential velocities are set in the fictitious cells according to Fig. 3, which illustrates the situation for the bottom boundary.

The velocities are set similarly for rigid walls along the other boundaries. The velocities are the only quantities that are set in the non-fluid cells. When the value of any quantity is needed on a wall boundary, e.g., the value of μ that is required to compute σ_{rz} on the bottom boundary for the z-momentum equation, it is obtained

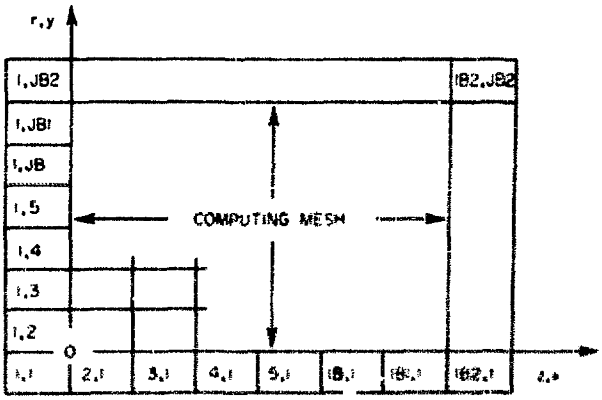


Fig. 2. Computing mesh and surrounding fictitious cells.

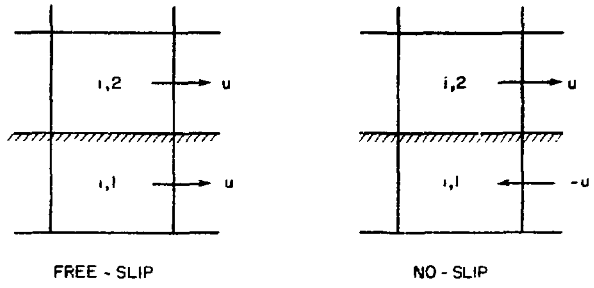


Fig. 3. Velocity boundary conditions for free-slip and no-slip rigid walls.

from averages of the nearby quantities in the fluid.

At an inflow boundary the specified quantities are located on the cell boundary with the u or v components of velocity. Figure 4 shows typical inflow cells along the left and top of the computing mesh. Values of quantities at the center of an inflow cell are obtained by linear extrapolation from their values on the boundary and nearby in the fluid.

The right boundary of the computing mesh can be a continuative outflow boundary, a rigid wall, or a combination with up to two outflow openings. The continuative outflow boundary is gradient-free in the axial direction across columns IB, IB1, and IB2 for all quantities except the velocities.

The boundary conditions that are available around the perimeter of the computing mesh are appropriate for the calculation of a variety of interior flows. The calculation of exterior flows, i.e., flows around obstacles, is also possible with the RICE program through the option of specifying obstacles in the interior of the computing mesh.

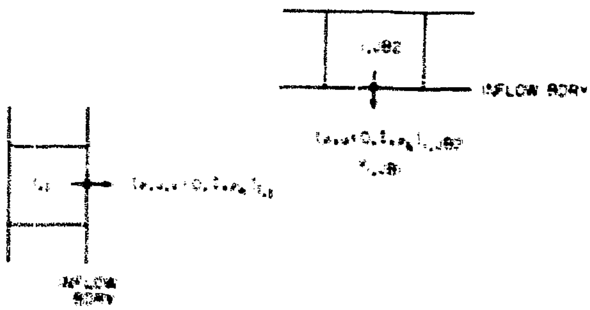


Fig. 4. Typical inflow cells along the left and top boundaries of the computing mesh.

Obstacles are built from combinations of solid and diagonally cut rectangular cells. The boundaries of an obstacle can be a combination of free-slip and no-slip, adiabatic, rigid walls and the right and bottom boundaries can have prescribed inflow to the computing mesh when they coincide with the cell boundaries. A continuative outflow boundary is not available for the interior obstacles. The boundary conditions are enforced on the interior obstacles in the same way as they are around the perimeter of the mesh for obstacle boundaries that coincide with cell boundaries. For obstacle boundaries that are formed by diagonally cut cells, the treatment of boundary conditions is described in Ref. 8.

F. Numerical Stability

The finite difference equations described earlier in this section for the mixture and species are generally unstable unless artificial diffusion of some sort is added. An unstable solution either oscillates or grows unbounded in time and in no way resembles the physical flow. Estimates of the amount of artificial diffusion needed for stability can easily be made. However, the computational time step is inversely proportional to the excess diffusion added so that overestimates can be costly in the total computer time required to obtain a solution. In addition, excessive diffusion reduces the spatial resolution of the calculation and hence the accuracy for a given cell size.

In the mass equation for the mixture, diffusion is added in the calculation of $\tilde{\rho}$ in Eq. (3.2) through the α-terms. The amount of diffusion needed is determined very accurately by calculating for each cell the value of the low-order diffusional truncation errors, which are the source of nonlinear instabilities. Diffusion is added in an amount to cancel these errors, which often have values that are orders of magnitude different in the r- and z-directions.

The momentum equations for the mixture, Eqs. (3.6) and (3.7), are stabilized by adding diffusion through the viscosity coefficients μ and λ. The amount of diffusion needed is estimated as

$$(2\mu+\lambda)/\rho > 3/2(u_{max})^2\delta t \text{ and } 3/2(v_{max})^2\delta t \qquad (3.37)$$

and

$$\mu/\rho > 1/2(u_{max})^2\delta t \text{ and } 1/2(v_{max})^2\delta t, \qquad (3.38)$$

which were derived from a truncation error analysis neglecting the second-order spatial errors. Inequalities (3.37) and (3.38) reduce to a single inequality when $\lambda = \mu$. The computational time step is restricted by the amount of excess diffusion furnished according to the relation

$$(2\mu + \lambda)\delta t/\rho < 1/2(\delta r)^2(\delta z)^2/[(\delta r)^2 + (\delta z)^2] \qquad (3.39)$$

derived from a Fourier analysis. The total diffusion added in accord with inequality (3.37) is considered excess to insure stability in a stagnation region. A similar inequality exists for the excess diffusion in the mass equation but need not be considered since inequality (3.39) is by far the more restrictive one. Simultaneous satisfaction of inequalities (3.37) - (3.39) generally provides a numerically stable solution to the momentum equations.

The total energy equation, Eq. (3.15), is stabilized by adding diffusion through the coefficient $\hat{k}$. The amount of diffusion needed is estimated as

$$\hat{k}/\rho > 1/2[(\gamma u)_{max}]^2\delta t \text{ and } 1/2\ [(\gamma v)_{max}]^2\delta t, \qquad (3.40)$$

which again is derived from a truncation error analysis neglecting the second-order spatial errors. The maximum excess diffusion in a stagnation region is $\hat{k}/\rho$ which results in an inequality similar to (3.39), i.e.,

$$\hat{k}\delta t/\rho < 1/2(\delta r)^2(\delta z)^2/[(\delta r)^2 + (\delta z)^2]. \qquad (3.41)$$

The stability of the species transport equations is provided by locally canceling the low-order diffusional truncation errors as is done in the mass equation for the mixture.

In summary, the finite difference equations in RICE are generally numerically stable when μ, λ, $\hat{k}$, and δt are chosen to satisfy simultaneously the inequalities (3.37) - (3.41). These inequalities reduce to inequalities involving only the ratio μ/ρ

and δt when λ and $\hat{\kappa}$ are written as

$$\lambda = A\mu, \ \hat{\kappa} = B\mu \tag{3.42}$$

where A and B are constants usually chosen as unity and $(\gamma_{max})^2$ respectively. *For this choice, inequalities* (3.37), (3.38), and (3.40) reduce to a single inequality. In addition, for the usual case where $(\gamma_{max})^2 \leq 3$, inequality (3.41) is automatically satisfied when (3.39) is satisfied.

The values of δt and μ/ρ determined for numerical stability are constants specified by the input data. To insure that the ratio $\mu_{i,j}/\rho_{i,j}$ is a constant in the presence of varying density, the shear viscosity is computed as

$$\mu_{i,j} = (\mu/\rho)\ \rho_{i,j}. \tag{3.43}$$

In certain situations the time step may have to be reduced because of strong coupling between the chemistry and the fluid dynamics. Strong coupling occurs when either the heat of reaction or the reaction rate is sufficiently large that a drastic change is produced in the internal energy density and hence the pressure in a single hydrodynamic time step.

IV. THE COMPUTER PROGRAM

The RICE computer program is described in this section. A global description of the subroutines and their functions and a card-by-card description of the input data are given. The program details are presented in three appendixes. Appendix A defines the FORTRAN IV symbols that are used in the program. The flow diagrams for each subroutine are presented in Appendix B and the FORTRAN IV listing of the program is given in Appendix C.

A. Global Description

The structure of the program and calling sequence for the various subroutines is shown in Fig. 5. The calculation begins with the RICE main program. The input data is read and in the case where the calculation is being continued from the results of a previous calculation, those results are read from TAPE5 by subroutine RTAPE5. In this case the data read from TAPE5 serves as the initial conditions for the new calculation; otherwise, uniform initial conditions are set as directed by the input

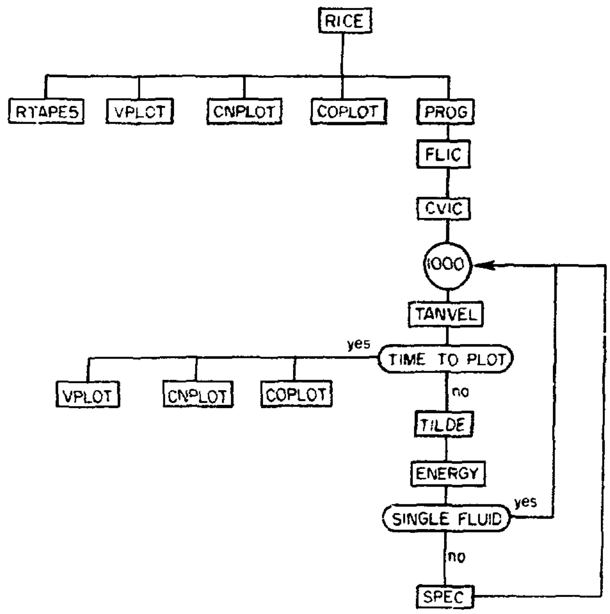


Fig. 5. Structure of the RICE program and the calling sequence for the various subroutines.

data. The plot routines VPLØT, CNPLØT, and CØPLØT are called next to convert the r,z coordinates of the computing mesh boundaries to the appropriate coordinates for later plots on microfilm. The final call is to subroutine PRØG, which assumes control of the calculation from this point to the end. The first call by PRØG is to subroutine FLIC. FLIC assigns values to the matrix of flags $FL_{i,j}$ that is used to distinguish the various types of cells in the computing mesh and around its perimeter. Thirteen different types of cells exist. Subroutine CVIC is called next to set initial conditions and inflow boundary data. The setting of initial conditions is bypassed in CVIC when the initial data is obtained from tape. The inflow boundary data must be specified in all cases, however, which offers the option of changing the inflow boundary data from what was on tape. Statement 1000 begins the repetitive portion of the computational cycle. Calculation of c^2, μ, T, and p are made prior to calling subroutine TANVEL. TANVEL sets the tangential velocities in the fictitious cells that form the rigid wall boundaries around the computing mesh and in the solid cells that form the boundaries of interior obstacles. Plots on microfilm are made at designated time increments. Subroutine VPLØT plots the velocity vectors for each cell in the computing mesh, which display the direction and magnitudes of the fluid motion. CNPLØT plots contours of designated quantities. Contours of ρ, p, I, T, Mach number, and species densities can be selected. The single plot produced by subroutine CØPLØT shows the distributions and relative concentrations in the computing mesh of the species selected for contour plots. This routine is convenient for producing color plots of the species distributions. In addition to plotted data, printed data is also available on microfilm and paper at designated time intervals, which do not have to be the same as the intervals for plotted data. Subroutine TILDE is called next to calculate values for $\tilde{\rho}$, $\widetilde{\rho v}$, and $\widetilde{\rho u}$. TILDE also performs the pressure iteration, which yields the final values for ρ^{n+1}, ρv^{n+1}, and ρu^{n+1} and the value for $\bar{p}$. Subroutine ENERGY follows TILDE for the calculation of the total energy density neglecting the effects of species diffusion and chemistry. For a single fluid the computational cycle is completed and return is to statement 1000. Subroutine SPEC is called for multicomponent flows to solve the species transport and chemistry equations and to update the total energy density to its final value. This completes the cycle for multicomponent flows and return is to statement 1000.

B. The Preprocessor (PREP) Program

The preprocessor is a feature of the RICE program that provides a capability for true variable dimensioning. From the size of the computing mesh (IB2,JB2) and the number of species NSPEC, the blocks of CØMMØN data are automatically constructed and written on file FSET8. The blocks are then read from this file and inserted in the RICE programs using the UPDATE features on the CDC-7600. To change the dimensions of CØMMØN variables, the UPDATE feature requires the following three cards.

```
*IDENT, name,
*DELETE, RCØM2. 24, RCØM2. 31
*READ, FSET8.
```

In this way only the precise amount of data storage needed for each problem is allocated in the machine. The large core memory of the CDC-7600 can be used by changing the word CØMMØN to LCM in format number 44 in the PREP program, which is listed in Appendix C.

C. Input Data

The input data is described card by card in this section. The total number of cards generally varies from one problem to the next. The first data card is read by PREP and is followed by an end-of-file card while the remaining cards are read by RICE.

Card No. 1: IB2, JB2, NSPEC (Format 3I6) where

IB2 = the number of cells in the z-direction including those in the two fictitious columns at the right and left boundaries.

JB2 = the number of cells in the r-direction including those in the two fictitious rows at the top and bottom boundaries.

NSPEC = the total number of species in the problem, $1 \leq \text{NSPEC} \leq 7$. NSPEC cannot be zero here since it is used to establish array dimensions.

Card No. 2: End-of-file card.

Card No. 3: NAME (Format 10A8), where cols. 2-80 are used for problem identification on plots and data prints.

Card No. 4: SLG, VEL, DEN, SMØ, TEM (Format 12F6.2) where these are the physical scales that have been used to establish the dimensionless input data for the problem.

SLG = L_s, the length scale.

VEL = U_s, the velocity scale.

DEN = ρ_s, the density scale.

SMØ = M_s, the molecular weight scale.

TEM = T_s, the temperature scale.

These quantities are used for reference only and are not used in the program.

Card No. 5: ITD, NTD, NSDMP, NFILE, NWDMP (Format 24I3) where

ITD = 0. TAPE5 is neither read nor written (NTD through NWDMP are not used).

= 1. TAPE5 is written but not read (NTD and NFILE are not used).

= 2. TAPE5 is read but not written (NSDMP and NWDMP are not used).

= 3. TAPE5 is read and written.

NTD = the number of the data set (dump) on TAPE5 that contains the data to be used for initial conditions.

NSDMP = the number of cycles between writes (dumps) on TAPE5.

NFILE = the number of the last dump that was written on TAPE5.

NWDMP = the number of the first dump to be written on TAPE5. Generally NWDMP = NFILE + 1.

Card No. 6: IB2, JB2, NØ, (NSL(K), K=1,4), LPR, CYCLE, ITC, (JPLØT(K), K=1,5), NSPEC, (ISPEC(K), K=1,7), NR (Format 8I3, I6, 15I3) where

IB2 = the same as on card no. 1.

JB2 = the same as on card no. 1.

NØ = the number of obstacles.

NSL(K) = the flags for rigid boundaries around the perimeter of the computing mesh. Values of either 0 for free slip or 1 for no slip are assigned for the bottom, right, top, and left boundaries in that order. The assigned values are ignored for inflow and outflow cells.

LPR = -2. omits plot identification, velocity vector plot (VPLØT), contour plots (CNPLØT) and all printed data on film. Used primarily to produce color plot by subroutine CØPLØT.

= -1. same as LPR = -2 except *does not* omit contour plots (CNPLØT).

= 0. Same as LPR = -2 except omits color plot (CØPLØT) instead of velocity vector plot (VPLØT).

= 1. allows all plotting and printing on film and printing on paper at the designated time intervals.

= 2. same as LPR = 1 except omit CØPLØT.

= 4. allows printing on paper only.

= 5. allows plotting and printing on film only.

CYCLE = the number of the computational cycle at which the calculation is to begin. CYCLE is also a counter which is advanced by one at each time step.

ITC = 1 for cylindrical coordinates or 0 for Cartesian coordinates.

JPLØT(K) = the flags for contour plots of ρ, p, I, T, and Mach number, in that order. When contour plots are desired JPLØT= 1, otherwise JPLØT=0.

NSPEC = the number of species in the problem, $0 \leq$ NSPEC ≤ 7. NSPEC may be 0 or 1 for a single fluid where the only difference is in the temperature calculation, i.e., $C_v = 1$ for NSPEC = 0 and C_v is specified later for NSPEC = 1.

ISPEC(K) = the flags for contour plots of species densities for up to seven species. When density contours are desired for species k, ISPEC(k) = 1, otherwise ISPEC(k) = 0.

NR = the number of chemical reactions, NR $\leq$ 3.

Card No. 7: DX, DY, TIME, DT, TSTØP, TPR, TPL, TPLD, DTC (Format 12F6.2) where

DX = $\delta z(\delta x)$, the cell dimension in the z-direction.

DY = $\delta r(\delta y)$, the cell dimension in the r-direction.

TIME = the initial time for the problem. TIME is also a counter and is advanced by δt each cycle.

DT = δt, the computational time step for the fluid dynamics.

TSTØP = the time at which the calculation is to stop. The total number of computational cycles to be made is (TSTØP - TIME)/DT.

TPR = the time interval for printing data on paper. Successive prints on paper are made until CYCLE is advanced from its initial value to values ≥ 0. From then on, printing on paper occurs at the specified time interval TPR.

TPL = the time interval for plotting data on film. Successive plots on film are made until CYCLE is advanced from its initial value to values > 1. From then on, plotting occurs at the specified time interval TPL.

TPLD = the time interval for printing data on film. Successive prints on film are made until CYCLE is advanced from its initial value to values > 1. From then on, printing on film occurs at the specified time interval TPLD.

DTC = δt_c, the computational time step for solution of the chemistry equations. The ratio $\delta t/\delta t_c$ must be an integer.

Card No. 8: ALA, BIE, THETA, PHI, ØMEGA, GAMMA, DXØU, ASQ (Format 12F6.2) where

ALA = A, the ratio of the viscosity coefficients λ/μ (usually 1.0 since the viscosities are artificial), see Eq. (3.42).

BIE = B, the ratio of the coefficient $\hat{\kappa}$ to the shear viscosity (usually $B = (\gamma_{max})^2$ from truncation error considerations), see Eq. (3.42).

THETA = θ, the time level parameter for the convection terms in the mixture mass equation, Eq. (3.1). A value of θ = 1/2 is usually used for most compressible flows, while $\theta = 1$ is used for incompressible flows.

PHI = ϕ, the time level parameter for the pressure gradient in the momentum equations, Eqs. (3.6) and (3.7). The values of ϕ are usually chosen similar to the values of θ.

ØMEGA = Ω, the constant over-relaxation factor for the pressure iteration, Eq. (3.11). The value of Ω is usually chosen as unity for compressible flows and $\geq$ 1.5 for incompressible flows.

GAMMA = γ, the specific heat ratio for a single fluid problem. When NSPEC > 0, γ is a cell variable determined from the specific heats of the mixture given by Eq. (2.9).

DXØU = the scale for the velocity vectors in the velocity vector plots. It is usually chosen as the smaller of $\delta z/u$ and $\delta r/v$, where u and v are estimates of the largest velocity components expected in the computing mesh.

ASQ = the sound speed used for incompressible flow calculations. ASQ = 0 for compressible flows and ASQ is set so that the maximum Mach number in the computing region is less than 10^{-5} for incompressible flows.

Card No. 9: FLØ(M), M = 1,12 (Format 12F6.2), where

FLØ(M) = the coordinates of the flow openings along the right, top, and left boundaries of the computing mesh. The coordinates along the right and left boundaries must be integral multiples of δr, while those along the top boundary must be integral multiples of δz. The openings along the right boundary are by definition outflow openings, while those along the top and left boundaries are inflow openings. FLØ(1)-FLØ(4) define the r-coordinates of the openings along the right boundary in the direction of increasing r. Figure 6, illustrates the cases for two, one, and zero openings on the right boundary. The origin of the coordinate system is the upper corner of the cell (1,1). FLØ(5)-FLØ(8) define the z-coordinates of the openings along the top boundary in the direction of increasing z. The assignment of z-coordinates is similar to the cases illustrated in Fig. 6. FLØ(9)-FLØ(12) define the r-coordinates of the openings along the left boundary in the direction of increasing r similar to what is done along the right boundary.

Card No. 10: UØ, VØ, VINT1, VINT2, UINL1, UINL2 (Format 12F6.2) where

UØ = the initial value of u for the fluid cells.

VØ = the initial value of v for the fluid cells.

VINT1 = the radial velocity of the fluid flowing into the computing mesh through the first opening along the top boundary.

VINT2 = the radial velocity of the fluid flowing into the computing mesh through the second opening along the top boundary.

UINL1 = the axial velocity of the fluid flowing into the computing mesh through the first opening along the left boundary.

UINL2 = the axial velocity of the fluid flow-

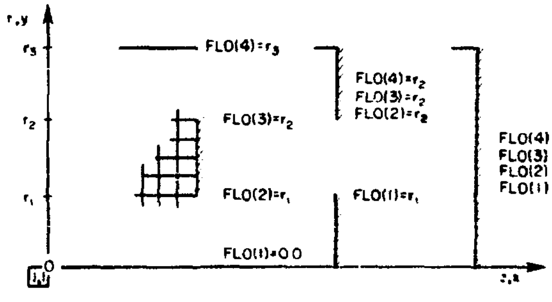


Fig. 6. Description of flow openings along the right boundary. The figure illustrates two, one, and zero openings.

ing into the computing mesh through the second opening along the left boundary.

Card No. 11: RHØØ, RINT1, RINT2, RINL1, RINL2 (Format 12F6.2) where

RHØØ = ρ_o, the initial value of ρ for the fluid cells. This quantity must always be non-zero even when the initial data is obtained from TAPE5 since it is used to define ε which establishes the convergence requirement for the pressure iteration, see Eq. (3.10). The value of ε is set in subroutine CVIC as $\varepsilon = 10^{-4}$ RHØØ. RHØØ is also used in specifying the constant value for the ratio $\mu/\rho = \mu_o/\rho_o$ determined for numerical stability.

RINT1 = the density of the fluid flowing into the computing mesh through the first opening along the top boundary.

RINT2 = the density of the fluid flowing into the computing mesh through the second opening along the top boundary.

RINL1 = the density of the fluid flowing into the computing mesh through the first opening along the left boundary.

RINL2 = the density of the fluid flowing into the computing mesh through the second opening along the left boundary.

Card No. 12: MUØ, MINT1, MINT2, MINL1, MINL2 (Format 12F6.2) where

MUØ = μ_o, the initial value of the shear viscosity μ for the fluid cells. MUØ and RHØØ are used to specify the constant value for the ratio $\mu/\rho = \mu_o/\rho_o$ determined for numerical stability.

MINT1 = the shear viscosity of the fluid flowing into the computing mesh through the first opening along the top boundary.

MINT2 = the shear viscosity of the fluid flowing into the computing mesh through the second opening along the top boundary.

MINL1 = the shear viscosity of the fluid flowing into the computing mesh through the first opening along the left boundary.

MINL2 = the shear viscosity of the fluid flowing into the computing mesh through the second opening along the left boundary.

The values of the shear viscosity of the fluid flowing into the computing mesh must be set in accord with the densities assigned on card no. 11 and the required ratio of μ/ρ determined for numerical stability.

Card No. 13: SIEØ, SIET1, SIET2, SIEL1, SIEL2 (Format 12F6.2) where

SIEØ = the initial value of the specific internal energy I for the fluid cells.

SIET1 = the specific internal energy of the fluid flowing into the computing mesh through the first opening along the top boundary.

SIET2 = the specific internal energy of the fluid flowing into the computing mesh through the second opening along the top boundary.

SIEL1 = the specific internal energy of the fluid flowing into the computing mesh through the first opening along the left boundary.

SIEL2 = the specific internal energy of the fluid flowing into the computing mesh through the second opening along the left boundary.

This completes the input data for problems involving a single fluid with NSPEC = 0 and no interior obstacles. The following additional data are required when NSPEC > 0.

Card No. 14: ETAØ (Format 12F6.2) where

ETAØ = η_o, the constant used for calculating the molecular diffusion coefficients for the species, see Eq. (3.24).

Card No. 15: FAØ(1), FAT1(1), FAT2(1), FAL1(1), FAL2(1) (Format 12F6.2) where

FAØ(1) = the initial value of the density of species (1) for the fluid cells.

FAT1(1) = the density of species (1) flowing into the computing mesh through the first opening along the top boundary.

FAT2(1) = the density of species (1) flowing into the computing mesh through the second opening along the top boundary.

FAL1(1) = the density of species (1) flowing into the computing mesh through the first opening along the left boundary.

FAL2(1) = the density of species (1) flowing into the computing mesh through the second opening along the left boundary.

Card Nos. 16-21: Similar to card no. 15 but for the remaining species (2) through (7). The species densities prescribed for the initial and boundary conditions should sum to the values prescribed for the mixture densities.

Card No. 22: SM(K), K=1,7 (Format 12F6.2) where

SM(K) = M_k, the molecular weight of species k.

Card No. 23: EP(K), K=1,7 (Format 12F6.2) where

EP(K) = the minimum value for the concentration of species k that can enter into a chemical reaction. EP(k) usually equals zero.

Card No. 24: GAMA(K), K=1,7 (Format 12F6.2) where

GAMA(K) = γ_k, the specific heat ratio for species k.

Card No. 25: CV(K), K=1,7 (Format 12F6.2) where

CV(K) = $(C_v)_k$, the specific heat at constant volume for species k.

Card No. 26: ALPHA(K), K=1,7 (Format 12F6.2) where

ALPHA(K) = α_k, a function of M_k for the calculation of the molecular diffusion coefficient, see Eq. (3.24).

The following data cards define the chemical reactions according to Eq. (2.13) and their respective rate multipliers given by Eq. (2.17). One card is used for each reaction and its rate multipliers. The total number of reactions, and hence the total number of data cards, is NR defined on card no. 6. If either NSPEC or NR is zero on card no. 6, the chemical reaction data should be omitted.

Card No. 27: (AM(K,L), K=1,7), (BM(K,L), K=1,7), Q(L), CF(L), ZETAF(L), EF(L), CB(L), ZETAB(L), EB(L) (Format 14I2, 2x, 7F6.2) where this card illustrates the data for chemical reaction number L.

AM(K,L) = a_k, the stoichiometric coefficients of the reactants, see Eq. (2.13).

BM(K,L) = b_k, the stoichiometric coefficients of the products, see Eq. (2.13).

Q(L) = q, the heat of reaction, see Eq. (2.18).

CF(L) = C, a constant for the calculation of the forward rate multiplier K_f, see Eq. (2.17).

ZETAF(L) = ζ, the temperature exponent for the calculation of the forward rate multiplier, see Eq. (2.17).

EF(L) = $E^{\dagger}$, the ratio of the activation energy to the universal gas constant for the forward rate multiplier, see Eq. (2.17).

CB(L) = C, a constant for the calculation of the backward rate multiplier, see Eq. (2.17).

ZETAB(L) = ζ, the temperature exponent for the calculation of the backward rate multiplier, see Eq. (2.17).

EB(L) = $E^{\dagger}$, the ratio of the activation energy to the universal gas constant for the backward rate multiplier, see Eq. (2.17).

The following data cards define the interior obstacles. The number of obstacles is NØ specified on card no. 6. Five data cards are required to define each obstacle. If NØ is zero these data cards should be omitted.

Card No. 28: ITØ(N), NSØ(N), (ØB(M,N), M = 1,6), (ØIB(M,N), M = 1,2) (Format 2I3, 11F6.2) where this card illustrates the data for obstacle number N.

ITØ(N) = the flags that define the type of obstacle. An obstacle must be rectangular or triangular in shape and can be made up of one or more cells. By combining obstacles, shapes other than simple rectangles and triangles are produced. Rectangular obstacles can have a single flow opening along either the right or bottom boundary to provide inflow to the computing mesh. The four obstacle types are shown in Fig. 7.

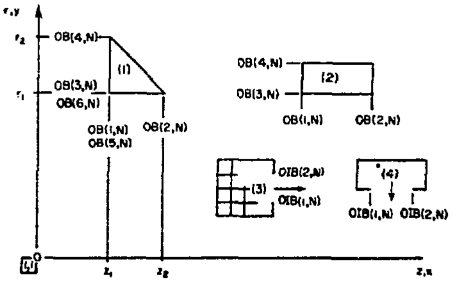


Fig. 7. Description of obstacles that can be placed in the interior of the computing mesh.

NSØ(N) = the flags that define the boundaries of obstacle number N as free slip, NSØ(N) = 0, or no slip, NSØ(N) = 1.

ØB(M,N) = the coordinates that define the location of the obstacle, see Fig. 7. For rectangular obstacles the z-coordinates of the left and right boundaries and the r-coordinates of the bottom and top boundaries define the obstacle's size and location. These coordinates are specified by ØB(1,N) - ØB(4,N). Triangular obstacles require additional data to specify the (z,r)-coordinates of the 90° angle. These coordinates are specified in ØB(5,N) and ØB(6,N), respectively, and are not used for rectangular obstacles.

ØIB(M,N) = the coordinates that define the flow opening along either the right or bottom boundary of the obstacle when these boundaries coincide with the cell boundaries, see Fig. 7. An opening along the right boundary must be an integral multiple of δr in height, while an opening along the bottom boundary must be an integral multiple of δz in width. The height or width of the opening is ØIB(2,N) - (ØIB(1,N).

Card No. 29: UINRØ(N), RINRØ(N), MINRØ(N), SIERØ(N) (Format 12F6.2) where for obstacle number N

UINRØ(N) = axial velocity of the fluid flowing into the computing mesh from a type (3) obstacle.

RINRØ(N) = density of the fluid flowing into the computing mesh from a type (3) obstacle.

MINRØ(N) = shear viscosity of the fluid flowing into the computing mesh from a type (3) obst .le. The value of MINRØ(N) should be determined as MINRØ(N) = (μ_o/ρ_o) RINRØ(N).

SIERØ(N) = specific internal energy of the fluid flowing into the computing mesh from a type (3) obstacle.

Card No. 30: VINBØ(N), RINBØ(N), MINBØ(N), SIEBØ(N) (Format 12F6.2) where for obstacle number N

VINBØ(N) = the radial velocity of the fluid flowing into the computing mesh from a type (4) obstacle.

RINBØ(N) = the density of the fluid flowing into the computing mesh from a type (4) obstacle.

MINBØ(N) = the shear viscosity of the fluid flowing into the computing mesh from a type (4) obstacle. The value of MINBØ(N) should be determined as MINBØ(N) = (μ_o/ρ_o) RINBØ(N).

SIEBØ(N) = the specific internal energy of the fluid flowing into the computing mesh from a type (4) obstacle.

Card No. 31: FARØ(K,N), K = 1,7 (Format 12F6.2) where for obstacle number N

FARØ(K,N) = the density of species k flowing into the computing mesh from a type (3) obstacle.

Card No. 32: FABØ(K,N), K = 1,7 (Format 12F6.2) where for obstacle number N

FABØ(K,N) = the density of species k flowing into the computing mesh from a type (4) obstacle.

Immediately after all the input data is read, it is written on paper and microfilm by the RICE main program. Following this, control is transferred to subroutine PRØG which then calls subroutine FLIC. The cell flags that differentiate the various types of cells are set in FLIC and printed on paper and film at the beginning of subroutine CVIC. The printed input data and cell flags provide a convenient check on the setup of the problem. The flagging scheme for the various computational cells is described in Table I.

TABLE I

CELL FLAGS

FL(I,J)	Cell Types
	Solid and fluid cells:
1	(a) Fluid cell
2	(b) Solid cell with free-slip boundary
3	(c) Solid cell with no-slip boundary
4	(d) Outflow cell
5	(e) Inflow cell
	Diagonally cut cells (solid portion is shaded):
6	(a)  With free-slip boundary
7	(b)  With free-slip boundary
8	(c)  With free-slip boundary
9	(d)  With free-slip boundary
10	(e)  With no-slip boundary
11	(f)  With no-slip boundary
12	(g) With no-slip boundary
13	(h)  With no-slip boundary

V. EXAMPLE PROBLEM

The capabilities of the RICE program are demonstrated through the calculation of an example problem. The complex fluid dynamics, species mixing, and nonequilibrium chemical reactions that take place in the lasing cavity of a continuous flow chemical laser provide a realistic and comprehensive problem for calculation. The particular laser system to be considered is an HF laser designed by the Aerospace Corporation.[9] F-atoms enter the lasing cavity through a bank of two-dimensional supersonic nozzles while H_2 is injected under high pressure through orifices located at the nozzle exit plane between adjacent nozzles. A portion of the nozzle bank is shown schematically in Fig. 8. The full nozzle bank contains 37 nozzles and is 1.25 cm deep. The F-atoms are obtained by the dissociation of SF_6. A nitrogen diluent is added for control of the exit conditions. Under conditions of complete dissociation of the SF_6, the fluorine nozzles flow N_2, S, and F. Hydrogen enters the lasing cavity under high pressure through a perforated tube. For the calculation, the series of perforations is approximated as a two-dimensional slit. The F-atoms and H_2 mix and react in the lasing cavity to produce vibrationally excited HF which serves as the lasing medium.

A. Input

Neglecting boundary layers, the uniform flow conditions at the exit plane of the fluorine nozzle are given in Table II. These conditions imply an exit Mach number of 4.3, a temperature of 656°K, and an exit pressure of 6.77 x 10^3 dynes/cm^2 (5.0 torr) for the mixture.

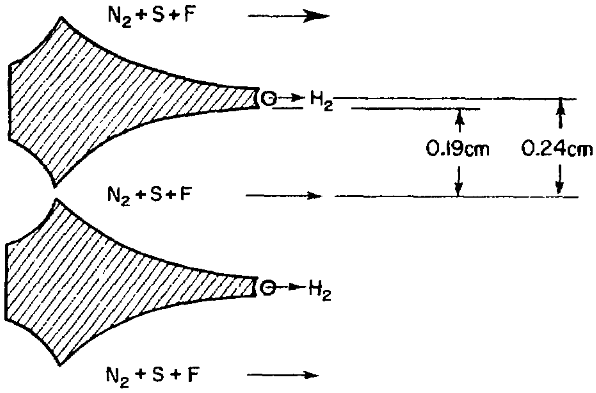


Fig. 8. Cross-sectional view of a portion of the laser nozzle bank.

Hydrogen enters the lasing cavity through a narrow slit, 0.016 cm, under the sonic conditions given in Table III. The injection pressure for these conditions is 2.96 x 10^5 dynes/cm^2 (222 torr).

The chemical reactions of H_2 with F populate several vibrational levels of HF that provide the lasing medium. For the purpose of this example problem, a single reaction equation is solved,

$$H_2 + F \rightarrow HF^* + H \qquad (5.1)$$

where the vibrational states of HF are represented collectively by HF*. The rate multiplier for the forward reaction K_f is given by Eq. (2.17), with $C = 1.6 \times 10^{14}$ cm^3/mole/s, $\zeta = 0$, and $E^{\ddagger} = 800°K$ from Ref. 10, while for the backward reaction, $K_b = 0$. The heat of reaction is $q = 5 \times 10^{11}$ erg/mole. Table IV summarizes the data describing the chemical reaction.

TABLE II

FLOW CONDITIONS AT THE EXIT PLANE OF THE FLUORINE NOZZLES

Mixture:	u(cm/s)	v(cm/s)	ρ(g/cm^3)	I(erg/g)
H_2 + S + F	2.30 x 10^5	0.	3.35 x 10^{-6}	4.82 x 10^9
Species (k):	ρ_k(g/cm^3)	M_k(g/mole)	γ_k	$(C_v)_k$(erg/g/°K)
1 = N_2	3.06 x 10^{-6}	28.0	1.40	7.47 x 10^6
2 = S	6.30 x 10^{-8}	32.0	1.67	3.92 x 10^6
3 = F	2.25 x 10^{-7}	19.0	1.67	6.61 x 10^6

TABLE III

HYDROGEN INJECTION CONDITIONS

Mixture:	u(cm/s)	v(cm/s)	$\rho(g/cm^3)$	I(erg/g)
H_2	1.33×10^5	0.	2.35×10^{-5}	3.15×10^{10}
Species (k):	$\rho_k(g/cm^3)$	M_k(g/mole)	γ_k	$(C_v)_k$(erg/g/°K)
$4 = H_2$	2.35×10^{-5}	2.0	1.4	1.05×10^8

TABLE IV

CHEMICAL REACTION DATA

$$H_2 + F \rightarrow HF^* + H$$

Species (k):	$\rho_k(g/cm^3)$	M_k(g/mole)	γ_k	$(C_v)_k$(erg/g/°K)
$5 = HF^*$	-	20.0	1.40	1.05×10^7
$6 = H$	-	1.0	1.67	1.26×10^8

a_k: 0, 0, 1, 1, 0, 0

b_k: 0, 0, 0, 0, 1, 1

$k_f = C\,T^{\zeta} e^{-E^{+}/T}$ where $C = 1.6 \times 10^{14}$ cm^3/mole/s, $\zeta = 0$, $E^{+} = 800°K$

$k_b = C\,T^{\zeta} e^{-E^{+}/T}$ where $C = \zeta = E^{+} = 0$

$q = 5 \times 10^{11}$ erg/mole

Mixing of the species is produced through the action of molecular diffusion. Coefficients that describe the diffusion of the various species into the mixture are obtained by approximating the mixture as pure nitrogen. The diffusion coefficients are defined by the data in Table V.

TABLE V

DIFFUSION COEFFICIENTS

$$D_k = \alpha_k \eta \ (cm^2/s), \quad \eta = \eta_o T^{3/2}/p \ (cm^2/s),$$

$$\alpha_k = (M_N/M_k)^{1/2}$$

$$\eta_o = 224 \text{ dynes } (°K)^{-3/2} s^{-1}$$

$$\alpha_k = 0.189,\ 0.177,\ 0.229,\ 0.707,\ 0.224,\ 1.0$$

The computational region begins slightly upstream of the nozzle exit plane and its upper and lower boundaries coincide with the center line of the hydrogen slit and the axis of the lower, adjacent fluorine nozzle, respectively. The computing region is shown in Fig. 9.

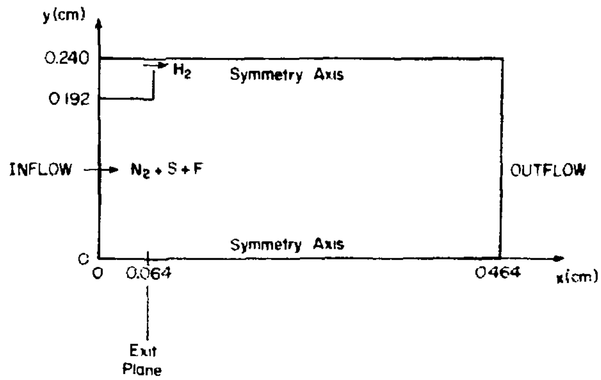


Fig. 9. Geometry of the computing region for the sample problem.

The dimensions of the computational cells are δx = 0.008 cm and δy = 0.008 cm. The number of cells in the x- and y-directions are IB2 = 62 and JB2 = 32, respectively.

The flow openings along the right and left boundaries are specified in Table VI.

TABLE VI

FLOW OPENING DATA

$FL\emptyset_m$ (cm):	0.,	0.240,	0.240,	0.240,
	0.,	0.,	0.,	0.,
	0.,	0.192,	0.192,	0.192

The shape of the interior obstacle is rectangular so that it can be described by a single RICE obstacle. The half-width of the H_2 injection slit is 0.008 cm. The data describing the obstacle geometry is given in Table VII.

TABLE VII

OBSTACLE DATA

OB_m (cm):	0,	0.064,	0.192,	0.240
OIB_m (cm):	0.232,	0.240		

The viscosity coefficients, thermal conductivity, and time step are determined for numerical stability by simultaneously satisfying inequalities (3.37)-(3.41). These inequalities reduce to

$\mu/\rho > 2.65 \times 10^{10}\ \delta t$, ... (3.37), (3.38), (3.40)

$\mu/\rho < 5.33 \times 10^{-6}/\delta t$, ... (3.39)

$\mu/\rho < 5.73 \times 10^{-6}/\delta t$, ... (3.41)

when A = 1., and $B = (\gamma_{max})^2 = 2.79$ in Eq. (3.42). Inequality (3.41) is satisfied automatically when (3.39) is satisfied. A solution of (3.37) and (3.39) yields

$$\delta t = 5\text{x}10^{-9}\text{s},\ \mu/\rho = 221\ \text{cm}^2/\text{s}. \qquad (5.2)$$

The above sets of data, given in Tables II-VII, with Eq. (5.2) are dimensionally consistent and can be used as input data for the calculation. However, dimensionless quantities that are more nearly of unit order are obtained by appropriately scaling the data. A convenient choice of scales is given in Table VIII.

The number of computational cycles required to obtain a steady state solution can be estimated as

$$\#\ \text{cycles} \approx 2.5\ (\text{IB2-2})\ \delta x/(u_{input}\ \delta t) \qquad (5.3)$$

where u_{input} refers to a typical axial velocity at the inflow boundary. With u_{input} as 2.3×10^5 cm/s, Eq. (5.3) indicates that about 1040 cycles are needed to obtain a steady state.

TABLE VIII

PHYSICAL SCALES

L_s = 1 cm, $u_s = 10^5$ cm/s, $\rho_s = 10^{-5}$ g/cm^3,

M_s = 1 g/mole, T_s = 100°K

The dimensionless quantities denoted by $\bar{\ }$ are obtained as

$$\bar{u} = u/u_s,\ \bar{v} = v/u_s$$
$$\bar{\rho} = \rho/\rho_s$$
$$\bar{I} = I/u_s^2$$
$$\bar{M}_k = M_k/M_s$$
$$\bar{C}_v = C_v\ T_s/u_s^2$$
$$\bar{C} = C\ \rho_s L_s/(M_s u_s)$$
$$\bar{E}^+ = E^+/T_s$$
$$\bar{q} = q/(u_s^2 M_s)$$
$$\bar{\eta}_o = \eta_o T_s^{3/2}/(\rho_s L_s u_s^3)$$
$$\overline{\delta t} = \delta t\ u_s/L_s$$
$$\bar{\mu} = \mu/(\rho_s u_s L_s)$$
$$\overline{\delta x} = \delta x/L_s,\ \overline{\delta y} = \delta y/L_s$$

Dimensionless quantities are used in the example problem.

B. Output

The dimensionless input data printed by RICE on paper and film is shown in Table IX. Although the problem contains six species, NSPEC is set to 7 to introduce a fictitious species whose density should be zero everywhere in the computing mesh for all time. The density of this last species is calculated by Eq. (3.21) and provides an indication of how well the sum of the real species densities compares to the mixture density. To minimize the influence of the fictitious species, its properties, M_7, γ_7, $(C_v)_7$, and α_7, are set to zero.

The time step for solution of the chemistry equations δt_c is set equal to the hydrodynamic time step δt because the reactant concentrations in the mix region are expected to be low and a low reaction rate will be produced. This is in fact confirmed by computing the characteristic time for the reaction from the steady solution data.

Table X shows the cell flags produced by subroutine FLIC and printed on paper by subroutine CVIC.

The initial values of the flow variables specified by the input data are printed on paper and film and reproduced for rows j = 30-32 in Table XI. The data, printed across the page, corresponds to i,j, $(FL_{i,j})$, $(u_{i,j})$, $(v_{i,j})$, $(\rho_{i,j})$, $(p_{i,j})$, $(I_{i,j})$, $(T_{i,j})$, $(\mu_{i,j})$ for the mixture, followed by i,j, $(FL_{i,j})$ $(\rho_k)_{i,j}$ for the species where k = 1-7. The seven species are designated in Table XI by the letters A-G, respectively.

Similarly, the data at the end of one computational cycle is printed in Table XII for the same rows. In accord with the input data, i.e., CYCLE = 0, this print occurs only on film unless paper printouts are specified by setting CYCLE equal to a negative number.

A steady state solution is obtained after 1000 cycles. Plots of the velocity vector field, contours of ρ, p, I, T, Mach number, and species densities, and the distribution of the species in the computing mesh are shown in Fig. 10. These plots are produced in accord with the specification of TPL.

The printed data is given in Table XIII for rows 20-22. This data is printed on paper and film as specified by TPR and TPLD.

TABLE IX

INPUT DATA READ AND PRINTED BY NICE

[illegible]

--SCALING QUANTITIES--

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

INFLOW DATA [illegible]

[illegible]

-- SPECIES DATA --

[illegible]

--SPECIES REACTION DATA--

[illegible]

-- OBSTACLE DATA --

ITO	NBO	OB(1)	OB(2)	OB(3)	OB(4)	OB(5)	OB(6)	OIB(1)	OIB(2)
3	0	0.	6.4000E-02	1.9200E-01	2.4000E-01	0.	0.	2.3200E-01	2.4000E-01

UIBO	RIBO	MIBO	SIBO
1.350E+00	2.350E+00	3.190E-03	3.190E+00

VIBO	RIBO	MIBO	SIBO
0.	0.	0.	0.

FAC(1)A	FAC(2)A	FAC(3)A	FAC(4)A	FAC(5)A	FAC(6)A	FAC(7)A
0.	0.	0.	2.350E+00	0.	0.	0.

FAC(1)B	FAC(2)B	FAC(3)B	FAC(4)B	FAC(5)B	FAC(6)B	FAC(7)B
0.	0.	0.	0.	0.	0.	0.

TABLE 3

CELL PLANS PRODUCED BY [illegible] AND PLOTTED BY [illegible]

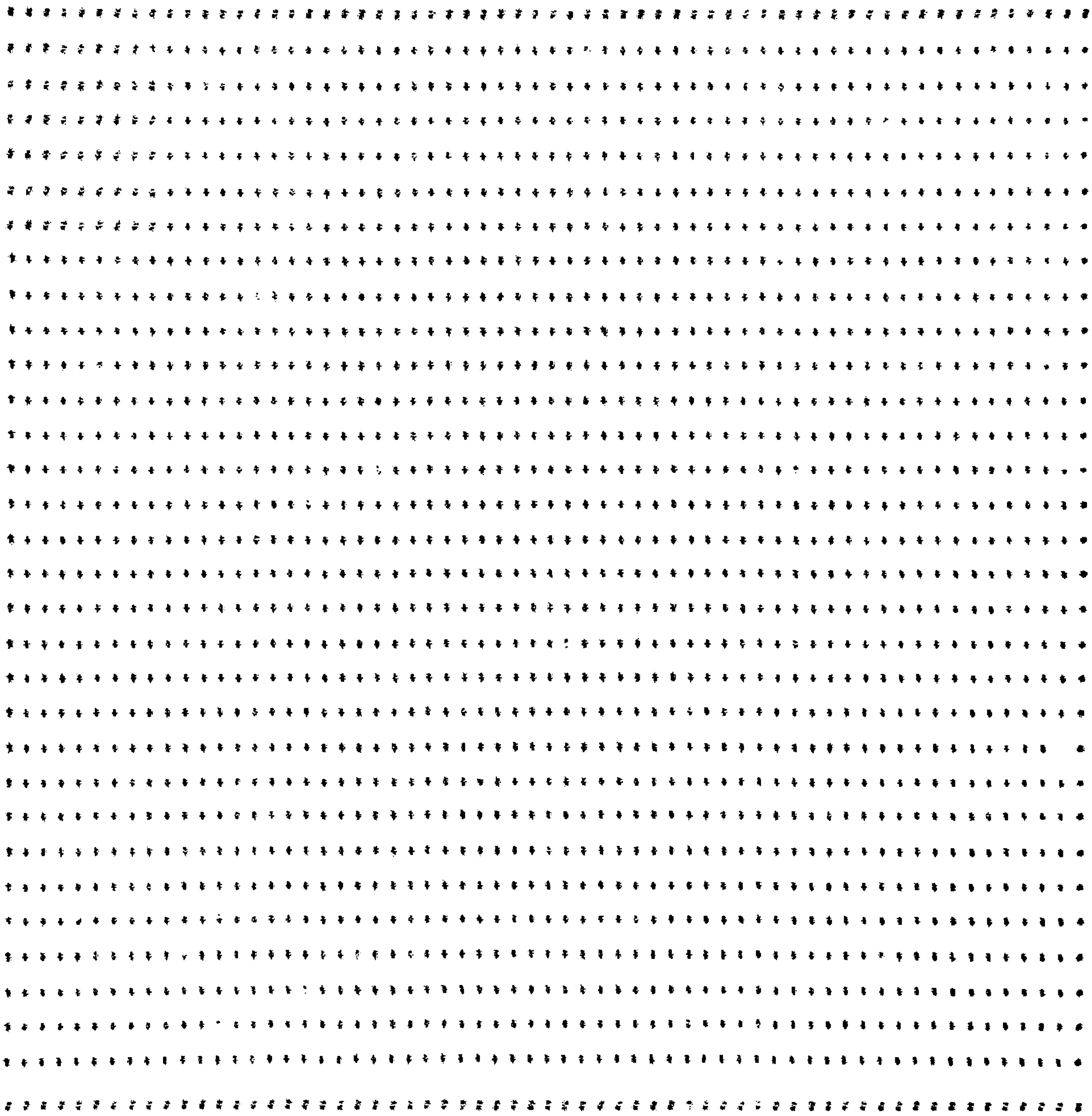


TABLE XI

INITIAL DATA AT CYCLE = 0, TIME = 0

CYCLE= 0 TIME= 0.000 RICE L.A. REPORT PROBLEM---8/23/74

I	J	FL	U	V	RHO	PB	SIE	TEMP.	MU
1	30	2	0.	0.	0.	0.	0.	0.	0.
2	30	2	0.	0.	0.	0.	0.	0.	0.
3	30	2	0.	0.	0.	0.	0.	0.	0.
4	30	2	0.	0.	0.	0.	0.	0.	0.
5	30	2	0.	0.	0.	0.	0.	0.	0.
6	30	2	0.	0.	0.	0.	0.	0.	0.
7	30	2	0.	0.	0.	0.	0.	0.	0.
8	30	2	0.	0.	0.	0.	0.	0.	0.
9	30	2	0.	0.	0.	0.	0.	0.	0.
10	30	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
11	30	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
12	30	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
13	30	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
14	30	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
15	30	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
1	31	2	0.	0.	0.	0.	0.	0.	0.
2	31	2	0.	0.	0.	0.	0.	0.	0.
3	31	2	0.	0.	0.	0.	0.	0.	0.
4	31	2	0.	0.	0.	0.	0.	0.	0.
5	31	2	0.	0.	0.	0.	0.	0.	0.
6	31	2	0.	0.	0.	0.	0.	0.	0.
7	31	2	0.	0.	0.	0.	0.	0.	0.
8	31	2	0.	0.	0.	0.	0.	0.	0.
9	31	5	1.3300E+00	0.	2.3500E+00	2.9610E+00	3.1500E+00	3.0000E+00	5.1900E-03
10	31	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
11	31	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
12	31	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
13	31	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
14	31	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
15	31	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
1	32	2	0.	0.	0.	0.	0.	0.	0.
2	32	2	0.	0.	0.	0.	0.	0.	0.
3	32	2	0.	0.	0.	0.	0.	0.	0.
4	32	2	0.	0.	0.	0.	0.	0.	0.
5	32	2	0.	0.	0.	0.	0.	0.	0.
6	32	2	0.	0.	0.	0.	0.	0.	0.
7	32	2	0.	0.	0.	0.	0.	0.	0.
8	32	2	0.	0.	0.	0.	0.	0.	0.
9	32	2	1.3300E+00	0.	0.	0.	0.	0.	0.
10	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.
11	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.
12	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.
13	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.
14	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.
15	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.

CYCLE= 0 TIME= 0.000 RICE L.A. REPORT PROBLEM---8/23/74
SPECIES DENSITIES

I	J	FL	A	B	C	D	E	F	G
1	30	2	0.	0.	0.	0.	0.	0.	0.
2	30	2	0.	0.	0.	0.	0.	0.	0.
3	30	2	0.	0.	0.	0.	0.	0.	0.
4	30	2	0.	0.	0.	0.	0.	0.	0.
5	30	2	0.	0.	0.	0.	0.	0.	0.
6	30	2	0.	0.	0.	0.	0.	0.	0.
7	30	2	0.	0.	0.	0.	0.	0.	0.
8	30	2	0.	0.	0.	0.	0.	0.	0.
9	30	2	0.	0.	0.	0.	0.	0.	0.
10	30	1	3.3500E-01	0.	0.	0.	0.	0.	0.
11	30	1	3.3500E-01	0.	0.	0.	0.	0.	0.
12	30	1	3.3500E-01	0.	0.	0.	0.	0.	0.
13	30	1	3.3500E-01	0.	0.	0.	0.	0.	0.
14	30	1	3.3500E-01	0.	0.	0.	0.	0.	0.
15	30	1	3.3500E-01	0.	0.	0.	0.	0.	0.

I	J	FL	A	B	C	D	E	F	G
1	31	2	0.	0.	0.	0.	0.	0.	0.
2	31	2	0.	0.	0.	0.	0.	0.	0.
3	31	2	0.	0.	0.	0.	0.	0.	0.
4	31	2	0.	0.	0.	0.	0.	0.	0.
5	31	2	0.	0.	0.	0.	0.	0.	0.
6	31	2	0.	0.	0.	0.	0.	0.	0.
7	31	2	0.	0.	0.	0.	0.	0.	0.
8	31	2	0.	0.	0.	0.	0.	0.	0.
9	31	5	0.	0.	0.	2.3500E+00	0.	0.	0.
10	31	1	3.3500E-01	0.	0.	0.	0.	0.	0.
11	31	1	3.3500E-01	0.	0.	0.	0.	0.	0.
12	31	1	3.3500E-01	0.	0.	0.	0.	0.	0.
13	31	1	3.3500E-01	0.	0.	0.	0.	0.	0.
14	31	1	3.3500E-01	0.	0.	0.	0.	0.	0.
15	31	1	3.3500E-01	0.	0.	0.	0.	0.	0.

I	J	FL	A	B	C	D	E	F	G
1	32	2	0.	0.	0.	0.	0.	0.	0.
2	32	2	0.	0.	0.	0.	0.	0.	0.
3	32	2	0.	0.	0.	0.	0.	0.	0.
4	32	2	0.	0.	0.	0.	0.	0.	0.
5	32	2	0.	0.	0.	0.	0.	0.	0.
6	32	2	0.	0.	0.	0.	0.	0.	0.
7	32	2	0.	0.	0.	0.	0.	0.	0.
8	32	2	0.	0.	0.	0.	0.	0.	0.
9	32	2	0.	0.	0.	0.	0.	0.	0.
10	32	2	0.	0.	0.	0.	0.	0.	0.
11	32	2	0.	0.	0.	0.	0.	0.	0.
12	32	2	0.	0.	0.	0.	0.	0.	0.
13	32	2	0.	0.	0.	0.	0.	0.	0.
14	32	2	0.	0.	0.	0.	0.	0.	0.
15	32	2	0.	0.	0.	0.	0.	0.	0.

TABLE XII

SOLUTION DATA AT CYCLE = 1, TIME = 0.0005

CYCLE: 1 TIME: .001 RICE L.S. REPORT PROBLEM---8/23/74

I	J	FL	U	V	RHO	PD	SIE	TEMP.	MU
1	30	2	0.	0.	0.	0.	0.	0.	0.
2	30	2	0.	0.	0.	0.	0.	0.	0.
3	30	2	0.	0.	0.	0.	0.	0.	0.
4	30	2	0.	0.	0.	0.	0.	0.	0.
5	30	2	0.	0.	0.	0.	0.	0.	0.
6	30	2	0.	0.	0.	0.	0.	0.	0.
7	30	2	0.	0.	0.	0.	0.	0.	0.
8	30	2	0.	0.	0.	0.	0.	0.	0.
9	30	2	0.	0.	0.	0.	0.	0.	0.
10	30	1	2.0773E+00	-9.7210E-02	2.9197E 01	7.3261E-02	6.2731E-01	7.9617E+00	6.4494E-04
11	30	1	2.3132E+00	-5.2692E-05	3.3115E-01	6.0219E-02	4.5402E-01	6.1000E+00	7.3149E-04
12	30	1	2.3000E+00	0.	3.3500E-01	6.0977E-02	4.5506E-01	6.0868E+00	7.4000E-04
13	30	1	2.3000E+00	0.	3.3500E-01	6.4693E-02	4.8279E-01	6.4630E+00	7.4000E-04
14	30	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
15	30	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
1	31	2	0.	0.	0.	0.	0.	0.	0.
2	31	2	0.	0.	0.	0.	0.	0.	0.
3	31	2	0.	0.	0.	0.	0.	0.	0.
4	31	2	0.	0.	0.	0.	0.	0.	0.
5	31	2	0.	0.	0.	0.	0.	0.	0.
6	31	2	0.	0.	0.	0.	0.	0.	0.
7	31	2	0.	0.	0.	0.	0.	0.	0.
8	31	2	0.	0.	0.	0.	0.	0.	0.
9	31	5	1.3300E+00	0.	2.3500E+00	2.9610E+[illegible]	3.1900E+00	3.0000E+00	5.1900E-03
10	31	1	1.6275E+00	0.	5.4377E-01	6.4398E-01	2.9607E+00	6.0822E+00	1.2012E-03
11	31	1	2.3061E+00	0.	3.3322E-01	1.4440E-01	1.0834E+00	1.4318E+01	7.3608E-04
12	31	1	2.3000E+00	0.	3.3500E-01	5.8390E-02	4.3581E-01	5.8319E+00	7.4000E-04
13	31	1	2.3000E+00	0.	3.3500E-01	6.4636E-02	4.8236E-01	6.4573E+00	7.4000E-04
14	31	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
15	31	1	2.3000E+00	0.	3.3500E-01	6.4588E-02	4.8200E-01	6.4525E+00	7.4000E-04
1	32	2	0.	0.	0.	0.	0.	0.	0.
2	32	2	0.	0.	0.	0.	0.	0.	0.
3	32	2	0.	0.	0.	0.	0.	0.	0.
4	32	2	0.	0.	0.	0.	0.	0.	0.
5	32	2	0.	0.	0.	0.	0.	0.	0.
6	32	2	0.	0.	0.	0.	0.	0.	0.
7	32	2	0.	0.	0.	0.	0.	0.	0.
8	32	2	0.	0.	0.	0.	0.	0.	0.
9	32	2	1.3300E+00	0.	0.	0.	0.	0.	0.
10	32	2	1.6275E+00	0.	0.	0.	0.	0.	0.
11	32	2	2.3061E+00	0.	0.	0.	0.	0.	0.
12	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.
13	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.
14	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.
15	32	2	2.3000E+00	0.	0.	0.	0.	0.	0.

TABLE XII (cont)

CYCLE: 1 TIME: .001 RICE L.A. REPORT PROBLEM---8/23/74

SPECIES DENSITIES

I	J	FL	A	B	C	D	E	F	G
1	30	2	0.	0.	0.	0.	0.	0.	0.
2	30	2	0.	0.	0.	0.	0.	0.	0.
3	30	2	0.	0.	0.	0.	0.	0.	0.
4	30	2	0.	0.	0.	0.	0.	0.	0.
5	30	2	0.	0.	0.	0.	0.	0.	0.
6	30	2	0.	0.	0.	0.	0.	0.	0.
7	30	2	0.	0.	0.	0.	0.	0.	0.
8	30	2	0.	0.	0.	0.	0.	0.	0.
9	30	2	0.	0.	0.	0.	0.	0.	0.
10	30	1	2.9244E-01	0.	0.	1.1041E-03	0.	0.	-1.5747E-03
11	30	1	3.3006E-01	0.	0.	0.	0.	0.	1.0865E-03
12	30	1	3.3528E-01	0.	0.	0.	0.	0.	-2.7634E-04
13	30	1	3.3500E-01	0.	0.	0.	0.	0.	0.
14	30	1	3.3500E-01	0.	0.	0.	0.	0.	0.
15	30	1	3.3500E-01	0.	0.	0.	0.	0.	0.

I	J	FL	A	B	C	D	E	F	G
1	31	2	0.	0.	0.	0.	0.	0.	0.
2	31	2	0.	0.	0.	0.	0.	0.	0.
3	31	2	0.	0.	0.	0.	0.	0.	0.
4	31	2	0.	0.	0.	0.	0.	0.	0.
5	31	2	0.	0.	0.	0.	0.	0.	0.
6	31	2	0.	0.	0.	0.	0.	0.	0.
7	31	2	0.	0.	0.	0.	0.	0.	0.
8	31	2	0.	0.	0.	0.	0.	0.	0.
9	31	5	0.	0.	0.	2.3500E+00	0.	0.	0.
10	31	1	2.9585E-01	0.	0.	2.3104E-01	0.	0.	1.6884E-02
11	31	1	3.1951E-01	0.	0.	1.2821E-03	0.	0.	1.2432E-02
12	31	1	3.3513E-01	0.	0.	0.	0.	0.	-1.2703E-04
13	31	1	3.3500E-01	0.	0.	0.	0.	0.	0.
14	31	1	3.3500E-01	0.	0.	0.	0.	0.	0.
15	31	1	3.3500E-01	0.	0.	0.	0.	0.	0.

I	J	FL	A	B	C	D	E	F	G
1	32	2	0.	0.	0.	0.	0.	0.	0.
2	32	2	0.	0.	0.	0.	0.	0.	0.
3	32	2	0.	0.	0.	0.	0.	0.	0.
4	32	2	0.	0.	0.	0.	0.	0.	0.
5	32	2	0.	0.	0.	0.	0.	0.	0.
6	32	2	0.	0.	0.	0.	0.	0.	0.
7	32	2	0.	0.	0.	0.	0.	0.	0.
8	32	2	0.	0.	0.	0.	0.	0.	0.
9	32	2	0.	0.	0.	0.	0.	0.	0.
10	32	2	0.	0.	0.	0.	0.	0.	0.
11	32	2	0.	0.	0.	0.	0.	0.	0.
12	32	2	0.	0.	0.	0.	0.	0.	0.
13	32	2	0.	0.	0.	0.	0.	0.	0.
14	32	2	0.	0.	0.	0.	0.	0.	0.
15	32	2	0.	0.	0.	0.	0.	0.	0.

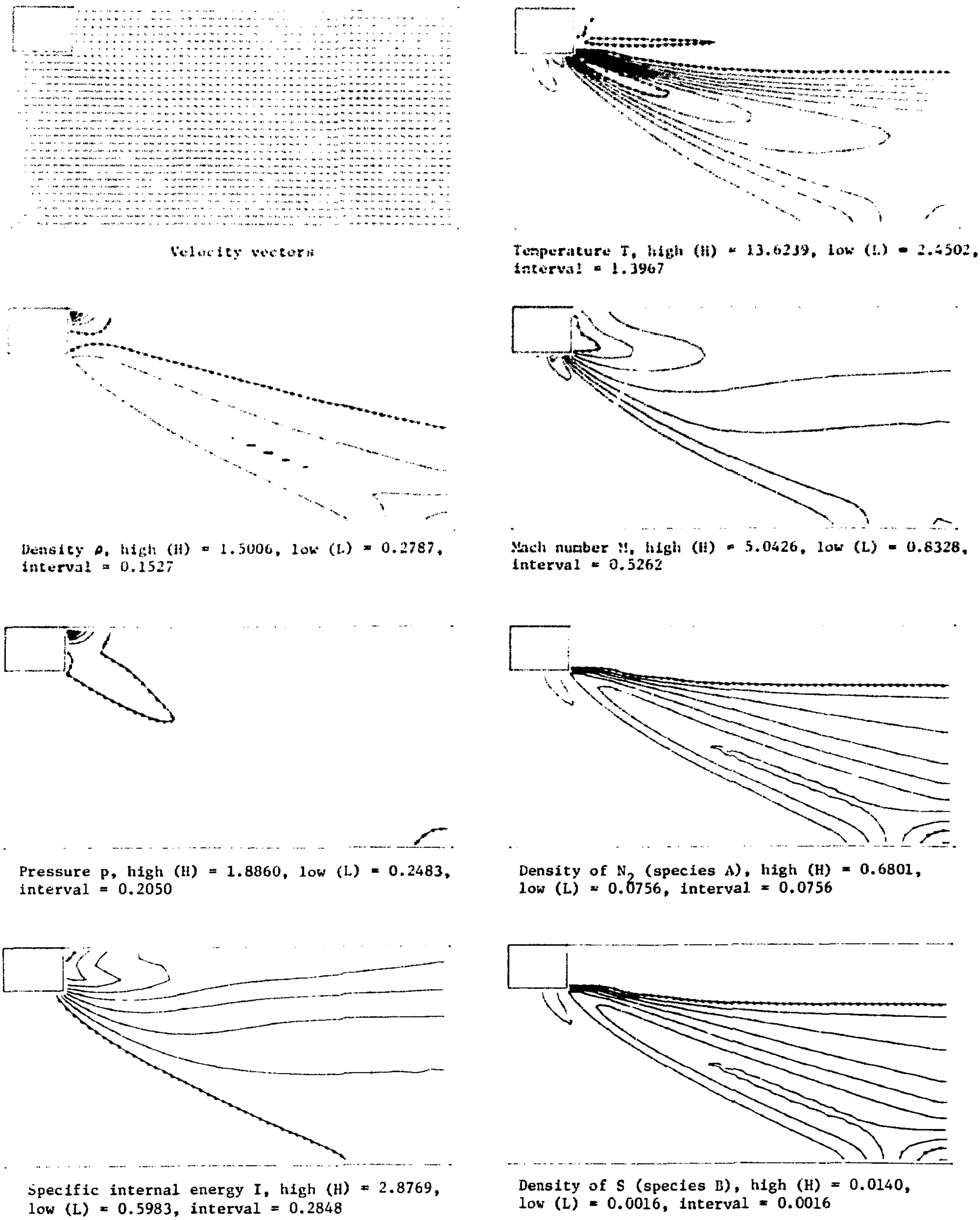
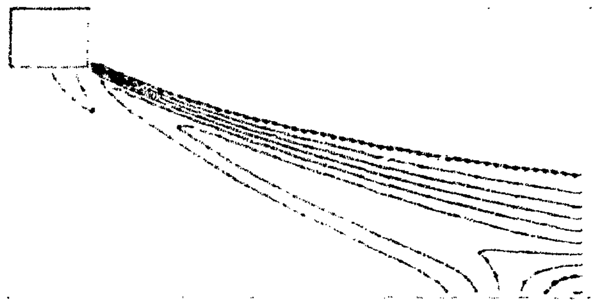
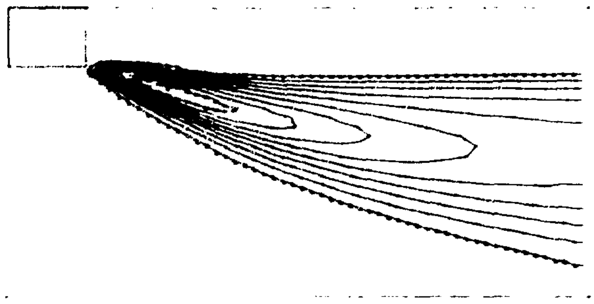


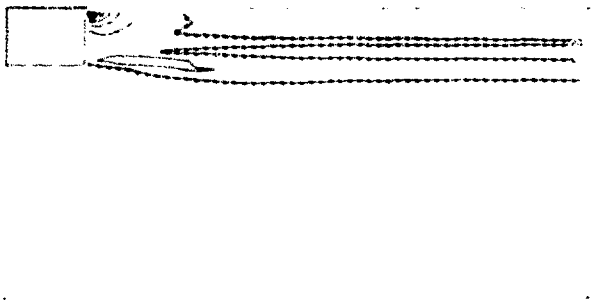
Fig. 10. Plots of the velocity vector field, contours of ρ, p, I, T, Mach number, and species densities, and the distribution of the species in the computing mesh.



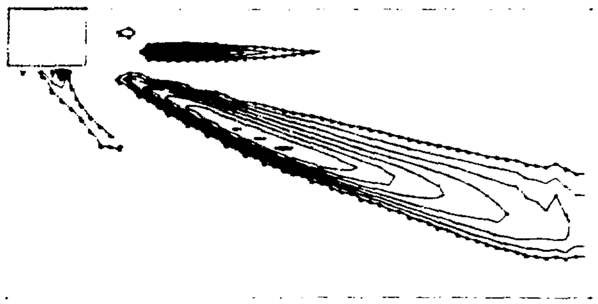
Density of F (species C), high (H) = 0.0500, low (L) = 0.0056, interval = 0.0056



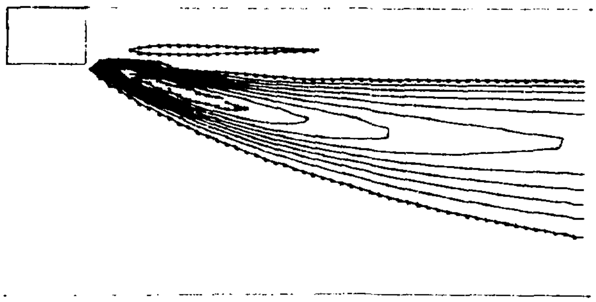
Density of H (species F), high (H) = 0.0011, low (L) = 0.0001, interval = 0.0001



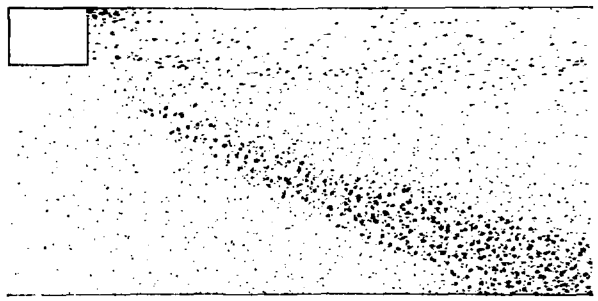
Density of H_2 (species D), high (H) = 1.4913, low (L) = 0.1657, interval = 0.1657



Density of fictitious species (species G), high (H) = 0.0034, low (L) = 0.0004, interval = 0.0004



Density of HF (species E), high (H) = 0.0258, low (L) = 0.0029, interval = 0.0029



Distribution of species densities

Fig. 10 (cont)

TABLE XIII

SOLUTION DATA AT CYCLE = 1000, TIME = 0.5

CYCLE= 1000 TIME= 0.000 RICE L.A. REPORT PROBLEM---8/23/74

I	J	FL	U	V	RHO	FB	SIE	TEMP.	MU
1	20	5	2.3000E+00	0.	3.3500E-01	6.7662E-02	4.8200E-01	6.5658E+00	7.4000E-04
2	20	1	2.3008E+00	1.1909E-04	3.3496E-01	6.7595E-02	4.8158E-01	6.5601E+00	7.3992E-04
3	20	1	2.3012E+00	6.6789E-04	3.3480E-01	6.7394E-02	4.8038E-01	6.5440E+00	7.3955E-04
4	20	1	2.3005E+00	5.8260E-04	3.3475E-01	6.7338E-02	4.8005E-01	6.5391E+00	7.3945E-04
5	20	1	2.2985E+00	-7.0293E-04	3.3504E-01	6.7643E-02	4.8181E-01	6.5630E+00	7.4008E-04
6	20	1	2.2966E+00	-2.4609E-03	3.3559E-01	6.8224E-02	4.8515E-01	6.6084E+00	7.4129E-04
7	20	1	2.2973E+00	-2.5001E-03	3.3600E-01	6.8626E-02	4.8742E-01	6.6402E+00	7.4220E-04
8	20	1	2.3000E+00	8.5346E-04	3.3569E-01	6.8306E-02	4.8558E-01	6.6164E+00	7.4153E-04
9	20	1	2.3069E+00	6.9191E-03	3.3437E-01	6.6911E-02	4.7755E-01	6.5086E+00	7.3860E-04
10	20	1	2.3123E+00	1.0266E-02	3.3245E-01	6.4705E-02	4.6447E-01	6.3307E+00	7.3436E-04
11	20	1	2.3099E+00	7.5784E-04	3.3163E-01	6.3121E-02	4.5422E-01	6.1881E+00	7.3255E-04
12	20	1	2.2897E+00	-3.3002E-02	3.3516E-01	6.4967E-02	4.6258E-01	6.2928E+00	7.4036E-04
13	20	1	2.2440E+00	-9.5775E-02	3.4731E-01	7.4054E-02	5.0883E-01	6.9062E+00	7.6720E-04
14	20	1	2.1760E+00	-1.7928E-01	3.7192E-01	9.3615E-02	6.0069E-01	8.1308E+00	8.2155E-04
15	20	1	2.1038E+00	-2.6662E-01	4.0967E-01	1.2398E-01	7.2219E-01	9.7564E+00	9.0495E-04
1	21	5	2.3000E+00	0.	3.3500E-01	6.7662E-02	4.8200E-01	6.5658E+00	7.4000E-04
2	21	1	2.3017E+00	2.3551E-04	3.3489E-01	6.7503E-02	4.8102E-01	6.5526E+00	7.3976E-04
3	21	1	2.3020E+00	9.9607E-04	3.3458E-01	6.7139E-02	4.7886E-01	6.5231E+00	7.3908E-04
4	21	1	2.2993E+00	1.9459E-04	3.3471E-01	6.7294E-02	4.7979E-01	6.5347E+00	7.3936E-04
5	21	1	2.2956E+00	-2.4822E-03	3.3548E-01	6.8166E-02	4.8489E-01	6.6037E+00	7.4107E-04
6	21	1	2.2948E+00	-4.4699E-03	3.3635E-01	6.9109E-02	4.9033E-01	6.6783E+00	7.4299E-04
7	21	1	2.2995E+00	-1.3272E-03	3.3630E-01	6.9001E-02	4.8963E-01	6.6711E+00	7.4288E-04
8	21	1	2.3088E+00	7.0977E-03	3.3461E-01	6.7154E-02	4.7893E-01	6.5273E+00	7.3915E-04
9	21	1	2.3175E+00	1.3783E-02	3.3180E-01	6.3932E-02	4.5981E-01	6.2677E+00	7.3294E-04
10	21	1	2.3147E+00	4.2426E-03	3.3024E-01	6.1563E-02	4.4488E-01	6.0600E+00	7.2948E-04
11	21	1	2.2865E+00	-3.6619E-02	3.3435E-01	6.4243E-02	4.5853E-01	6.2350E+00	7.3857E-04
12	21	1	2.2230E+00	-1.1088E-01	3.4968E-01	7.7114E-02	5.2628E-01	7.1346E+00	7.7242E-04
13	21	1	2.1395E+00	-2.0503E-01	3.8027E-01	1.0379E-01	6.5134E-01	8.8033E+00	8.4000E-04
14	21	1	2.0576E+00	-2.9543E-01	4.2435E-01	1.4305E-01	8.0425E-01	1.0800E+01	9.3738E-04
15	21	1	1.9972E+00	-3.6404E-01	4.7301E-01	1.8736E-01	9.4472E-01	1.2483E+01	1.0448E-03
1	22	5	2.3000E+00	0.	3.3500E-01	6.7662E-02	4.8200E-01	6.5658E+00	7.4000E-04
2	22	1	2.3032E+00	3.6838E-04	3.3475E-01	6.7336E-02	4.8004E-01	6.5393E+00	7.3945E-04
3	22	1	2.3022E+00	1.4077E-03	3.3429E-01	6.6804E-02	4.7689E-01	6.4954E+00	7.3843E-04
4	22	1	2.2957E+00	-7.5382E-04	3.3492E-01	6.7573E-02	4.8148E-01	6.5565E+00	7.3982E-04
5	22	1	2.2900E+00	-5.0152E-03	3.3653E-01	6.9460E-02	4.9256E-01	6.7069E+00	7.4338E-04
6	22	1	2.2948E+00	-5.9100E-03	3.3733E-01	7.0344E-02	4.9764E-01	6.7784E+00	7.4514E-04
7	22	1	2.3093E+00	4.7205E-03	3.3549E-01	6.8177E-02	4.8496E-01	6.6097E+00	7.4107E-04
8	22	1	2.3248E+00	1.7365E-02	3.3132E-01	6.3361E-02	4.5636E-01	6.2218E+00	7.3188E-04
9	22	1	2.3235E+00	1.1771E-02	3.2821E-01	5.9278E-02	4.3101E-01	5.8718E+00	7.2500E-04
10	22	1	2.2847E+00	-3.3734E-02	3.3233E-01	6.2571E-02	4.4931E-01	6.1065E+00	7.3410E-04
11	22	1	2.2001E+00	-1.1874E-01	3.5077E-01	8.0224E-02	5.4582E-01	7.3909E+00	7.7483E-04
12	22	1	2.0959E+00	-2.2177E-01	3.8723E-01	1.1579E-01	7.1358E-01	9.6332E+00	8.5538E-04
13	22	1	2.0054E+00	-3.1213E-01	4.3623E-01	1.6530E-01	9.0402E-01	1.1995E+01	9.6361E-04
14	22	1	1.9414E+00	-3.7206E-01	4.8464E-01	2.1610E-01	1.0645E+00	1.3730E+01	1.0706E-03
15	22	1	1.9003E+00	-4.0047E-01	5.1950E-01	2.5677E-01	1.1824E+00	1.4664E+01	1.1476E-03

TABLE XIII (cont)

CYCLE= 1000 TIME= 0.000 RICE L.A. REPORT PROBLEM---8/23/74

SPECIES DENSITIES

I	J	FL	A	B	C	D	E	F	G
1	20	5	3.0600E-01	6.3000E-03	2.2500E-02	0.	0.	0.	0.
2	20	1	3.0596E-01	6.2993E-03	2.2497E-02	5.4578E-08	3.8801E-08	1.9873E-09	2.0292E-04
3	20	1	3.0580E-01	6.2960E-03	2.2486E-02	-4.6779E-08	-4.0536E-08	3.4059E-10	2.1094E-04
4	20	1	3.0578E-01	6.2955E-03	2.2484E-02	1.4885E-08	6.6898E-08	1.4890E-09	1.9299E-04
5	20	1	3.0604E-01	6.3009E-03	2.2503E-02	-1.4365E-07	-1.0777E-07	-5.8626E-09	1.9059E-04
6	20	1	3.0655E-01	6.3114E-03	2.2540E-02	-1.2616E-07	1.5271E-07	-7.0183E-09	1.8066E-04
7	20	1	3.0668E-01	6.3162E-03	2.2565E-02	-8.3502E-08	-2.3883E-07	1.9164E-08	2.3236E-04
8	20	1	3.0655E-01	6.3113E-03	2.2540E-02	1.8115E-08	5.3956E-07	3.5999E-08	2.9206E-04
9	20	1	3.0525E-01	6.2847E-03	2.2446E-02	4.5479E-07	-6.4176E-07	6.9003E-08	3.7976E-04
10	20	1	3.0349E-01	6.2484E-03	2.2314E-02	-1.1610E-07	2.0001E-06	-1.0295E-08	3.8932E-04
11	20	1	3.0280E-01	6.2359E-03	2.2273E-02	-1.6426E-06	-3.6463E-06	1.2698E-07	2.3854E-04
12	20	1	3.0655E-01	6.3114E-03	2.2538E-02	6.6178E-07	5.0662E-06	-8.8003E-08	-2.4354E-04
13	20	1	3.1840E-01	6.5552E-03	2.3415E-02	-1.4757E-05	-1.0532E-05	-2.9155E-06	-1.0234E-03
14	20	1	3.4187E-01	7.0386E-03	2.5134E-02	-6.8179E-06	1.3764E-05	-1.7289E-06	-2.1354E-03
15	20	1	3.7733E-01	7.7606E-03	2.7742E-02	-5.5459E-06	-3.0017E-05	-2.2608E-06	-3.1296E-03
1	21	5	3.0600E-01	6.3000E-03	2.2500E-02	0.	0.	0.	0.
2	21	1	3.0590E-01	6.2979E-03	2.2492E-02	1.4638E-07	4.2304E-08	5.6744E-09	2.0367E-04
3	21	1	3.0562E-01	6.2922E-03	2.2473E-02	-2.3749E-07	4.9721E-08	-1.6699E-09	1.9853E-04
4	21	1	3.0579E-01	6.2956E-03	2.2484E-02	5.1168E-08	-1.7754E-08	-1.4140E-09	1.4216E-04
5	21	1	3.0651E-01	6.3104E-03	2.2538E-02	-8.0188E-07	1.2823E-07	-2.7429E-08	1.2951E-04
6	21	1	3.0728E-01	6.3263E-03	2.2593E-02	-1.8719E-08	-1.9978E-07	2.9814E-08	1.5470E-04
7	21	1	3.0713E-01	6.3232E-03	2.2584E-02	1.9185E-07	5.2214E-07	1.0454E-07	2.6800E-04
8	21	1	3.0548E-01	6.2894E-03	2.2460E-02	8.4140E-07	-4.3929E-07	1.1847E-07	3.6123E-04
9	21	1	3.0286E-01	6.2353E-03	2.2271E-02	1.3265E-06	1.9189E-06	1.1961E-07	4.3729E-04
10	21	1	3.0166E-01	6.2106E-03	2.2177E-02	-3.5792E-06	-2.1926E-06	1.0179E-07	1.9579E-04
11	21	1	3.0594E-01	6.2988E-03	2.2500E-02	-4.2569E-06	4.7192E-06	1.2076E-07	-3.6974E-04
12	21	1	3.2095E-01	6.6078E-03	2.3593E-02	-1.0955E-05	-5.9236E-06	-2.1776E-06	-1.4544E-03
13	21	1	3.5007E-01	7.2073E-03	2.5753E-02	-5.6918E-05	2.3905E-06	-9.9674E-06	-2.6992E-03
14	21	1	3.9105E-01	8.0513E-03	2.8656E-02	1.3989E-04	3.6616E-06	2.5478E-05	-3.5758E-03
15	21	1	4.3508E-01	8.9584E-03	3.1195E-02	6.4787E-04	5.7739E-04	1.1430E-04	-3.5654E-03
1	22	5	3.0600E-01	6.3000E-03	2.2500E-02	0.	0.	0.	0.
2	22	1	3.0575E-01	6.2949E-03	2.2481E-02	6.2118E-07	3.9903E-07	2.1960E-08	2.1959E-04
3	22	1	3.0539E-01	6.2875E-03	2.2455E-02	-1.3117E-06	-2.4807E-07	-2.5622E-08	1.5830E-04
4	22	1	3.0603E-01	6.3007E-03	2.2502E-02	-1.0119E-07	3.5852E-07	-1.3895E-08	8.3829E-05
5	22	1	3.0752E-01	6.3314E-03	2.2612E-02	-3.5302E-06	-6.0311E-07	-8.4672E-08	6.6052E-05
6	22	1	3.0810E-01	6.3433E-03	2.2656E-02	3.3048E-06	5.5372E-07	2.4046E-07	2.2199E-04
7	22	1	3.0625E-01	6.3052E-03	2.2514E-02	1.8407E-06	1.5779E-06	3.9179E-07	4.1152E-04
8	22	1	3.0233E-01	6.2244E-03	2.2232E-02	3.9390E-06	1.3928E-06	2.2659E-07	5.3149E-04
9	22	1	2.9977E-01	6.1717E-03	2.2031E-02	-2.3675E-06	3.1833E-06	5.2720E-07	2.3830E-04
10	22	1	3.0425E-01	6.2640E-03	2.2380E-02	-2.6937E-05	2.9107E-07	-5.6434E-08	-5.4240E-04
11	22	1	3.2233E-01	6.6362E-03	2.3678E-02	-1.8699E-05	7.6912E-06	-1.6372E-06	-1.8660E-03
12	22	1	3.5689E-01	7.3476E-03	2.6272E-02	-5.1153E-05	-4.5554E-05	-7.6440E-06	-3.1700E-03
13	22	1	4.0178E-01	8.2727E-03	2.9231E-02	4.9362E-04	1.2341E-04	6.0161E-05	-3.7334E-03
14	22	1	4.4460E-01	9.1555E-03	3.0763E-02	1.5051E-03	1.4974E-03	1.8503E-04	-3.0623E-03
15	22	1	4.7361E-01	9.7542E-03	2.9459E-02	3.0673E-03	4.6584E-03	3.7182E-04	-1.4200E-03

REFERENCES

1. W. C. Rivard, T. D. Butler, and O. A. Farmer, "Numerical Calculations of Viscous Supersonic Jet Mixing," Los Alamos Scientific Laboratory report (to be published).

2. F. H. Harlow and A. A. Amsden, "A Numerical Fluid Dynamics Calculation Method for All Flow Speeds," J. Comp. Phys. 8, 197-213 (1971).

3. A. J. Chorin, "Numerical Study of Thermal Convection in a Fluid Layer Heated from Below," AEC report NYO-1480-61 (August 1966).

4. C. W. Hirt and J. L. Cook, "Calculating Three-Dimensional Flows around Structures and over Rough Terrain," J. Comp. Phys. 10, 324-340 (1972).

5. R. B. Bird, W. E. Stewart, and E. N. Lightfoot, Transport Phenomena (John Wiley and Sons, Inc., New York, 1960), Chap. 16, p. 502.

6. W. C. Rivard, O. A. Farmer, T. D. Butler, and P. J. O'Rourke, "A Method for Increased Accuracy in Eulerian Fluid Dynamics Calculations," Los Alamos Scientific Laboratory report LA-5426-MS (October 1973).

7. A. A. Amsden and F. H. Harlow, "The SMAC Method: A Numerical Technique for Calculating Incompressible Fluid Flows," Los Alamos Scientific Laboratory report LA-4370 (May,1970).

8. T. D. Butler, "Recent Advances in Computational Fluid Dynamics," in Computing Methods in Applied Sciences and Engineering, Part 2, Lecture Notes in Computer Science no. 11 (1974).

9. H. Mirels and D. J. Spencer, "Power and Efficiency of a Continuous HF Chemical Laser," IEEE J. Quant. Elect. 7, 501-507 (1971).

10. N. Cohen, "A Review of Rate Coefficients for Reactions in the H_2-F_2 Laser System," Aerospace Corp. report TR-0172(2779)-2 (1971).

APPENDIX A

DEFINITION OF FØRTRAN VARIABLES

The FØRTRAN variables that appear in CØMMØN storage or that are used for input data are defined here in alphabetical order. The input data is defined in detail, card by card, in Sec. IV so that input variables are defined here by referencing the particular data card on which they appear.

FØRTRAN Symbol	Algebraic Symbol	Definition
ALA	A	Input data card no. 8.
ALPHA(K)	α_k	Input data card no. 26.
AM(K,L)	$a_{k,\ell}$	Input data card no. 27.
ASQ	-	Input data card no. 8.
BIE	B	Input data card no. 8.
BM(K,L)	$b_{k,\ell}$	Input data card no. 27.
CB(L)	C_ℓ	Input data card no. 27.
CF(L)	C_ℓ	Input data card no. 27.
CQ(I,J)	-	Dummy array used to store variables for plotting.
CSQ(I,J)	$(c^2)i,j$	Square of the adiabatic sound speed. Computed in subroutine PRØG as $c^2 = \gamma(\gamma - 1)I$.
CV(K)	$(C_v)_k$	Input data card no. 25.
CYCLE	-	Input data card no. 6.
DDX	-	Reciprocal of δz (δx). Computed in subroutine CVIC.
DDY	-	Reciprocal of δr (δy). Computed in subroutine CVIC.
DEN	ρ_s	Input data card no. 4.
DRHØB(I,K)	-	Temporary storage for the mass flux of species k across the top boundary of cell (i,j) produced by molecular diffusion to be used later as the mass flux across the bottom boundary of cell (i,j+1). Computed in subroutine SPEC as $(\delta f_k)_{i,j+\frac{1}{2}}$ according to Eq. (3.23).
DT	δt	Input data card no. 7.
DTC	δt_c	Input data card no. 7.
DX	$\delta z(dx)$	Input data card no. 7.
DXØU	-	Input data card no. 8.
DY	$\delta r(\delta y)$	Input data card no. 7.
EB(L)	E^+_ℓ	Input data card no. 27.
EF(L)	E_ℓ	Input data card no. 27.
EP(K)	-	Input data card no. 23.
ERRD	ϵ	Deviation from zero for convergence of the pressure iteration. Computed in subroutine CVIC as $\epsilon = 10^{-5}\ \rho_o$.
ETA(I,J)	$\eta_{i,j}$	Factor for the calculation of the molecular diffusion coefficients, $D_k = \lambda_k\eta$. Computed in subroutine SPEC as $\eta = \eta_o\ T^{\frac{3}{2}}/p$.
ETAØ	η_o	Input data card no. 14.
FA(I,J,K)	$(\rho_k)_{i,j}$	Density of species k in cell (i,j).
FABØ(K,N)	-	Input data card no. 32.
FAL1(K)	-	Input data card no. 15.
FAL2(K)	-	Input data card no. 15.
FAØ(K)	-	Input data card no. 15.
FARØ(K,N)	-	Input data card no. 31.

FØRTRAN Symbol	Algebraic Symbol	Definition
FAT1(K)	-	Input data card no. 15.
FAT2(K)	-	Input data card no. 15.
FL(I,J)	-	Cell flags (see Table I). Computed in subroutine FLIC.
FLØ(M)	-	Input data card no. 9.
GAMA(K)	γ_k	Input data card no. 24.
GAMMA	γ	Input data card no. 8.
IB	-	Number of cells in the z(x)-direction excluding the two fictitious columns along the right and left boundaries of computing mesh. Computed in program RICE as IB = IB2-2.
IB1	-	Computed in program RICE as IB1 = IB2-1.
IB2	-	Input data cards no. 1 and no. 6.
ISPEC(K)	-	Input data card no. 6.
IT	-	The number of iterations performed to satisfy Eq. (3.10).
ITC	-	Input data card no. 6.
ITD	-	Input data card no. 5.
ITMAX	-	The maximum number of iterations allowed for any cell when attempting to satisfy Eq. (3.10). Set in subroutine CVIC as ITMAX = 1000.
ITØ(N)	-	Input data card no. 28.
JB	-	Number of cells in the r(y)-direction excluding the two fictitious rows along the bottom and top boundaries of computing mesh. Computed in program RICE as JB = JB2-2.
JB1	-	Computed in program RICE as JB1 = JB2-1.
JB2	-	Input data cards no. 1 and no. 6.
JNM	-	Job identification obtained from the JØB card and printed on film.
JPLOT(M)	-	Input data card no. 6.
LPR	-	Input data card no. 6.
MINBØ(N)	-	Input data card no. 30.
MINT1	-	Input data card no. 12.
MINT2	-	Input data card no. 12.
MU(I,J)	$\mu_{i,j}$	Shear viscosity. Computed in subroutine PRØG according to Eq. (3.43).
MUØ	μ_o	Input data card no. 12.
NAME(M)	-	Input data card no. 3.
NCYDMP	-	The number of cycles between writes (dumps) on TAPE5 as specified by the input variable NSDMP. Set in program RICE as NCYDMP = NSDMP.
NFILE	-	Input data card no. 5.
NØ	-	Input data card no. 6.
NR	-	Input data card no. 6.
NSDMP	-	Input data card no. 5.
NSL(M)	-	Input data card no. 6.
NSØ(N)	-	Input data card no. 28.
NSPEC	-	Input data cards no. 1 and no. 6.
NTD	-	Input data card no. 5.
NWDMP	-	Input data card no. 5.
ØB(M,N)	-	Input data card no. 28.
ØIB(M,N)	-	Input data card no. 28.
ØMEGA	-	Input data card no. 8.

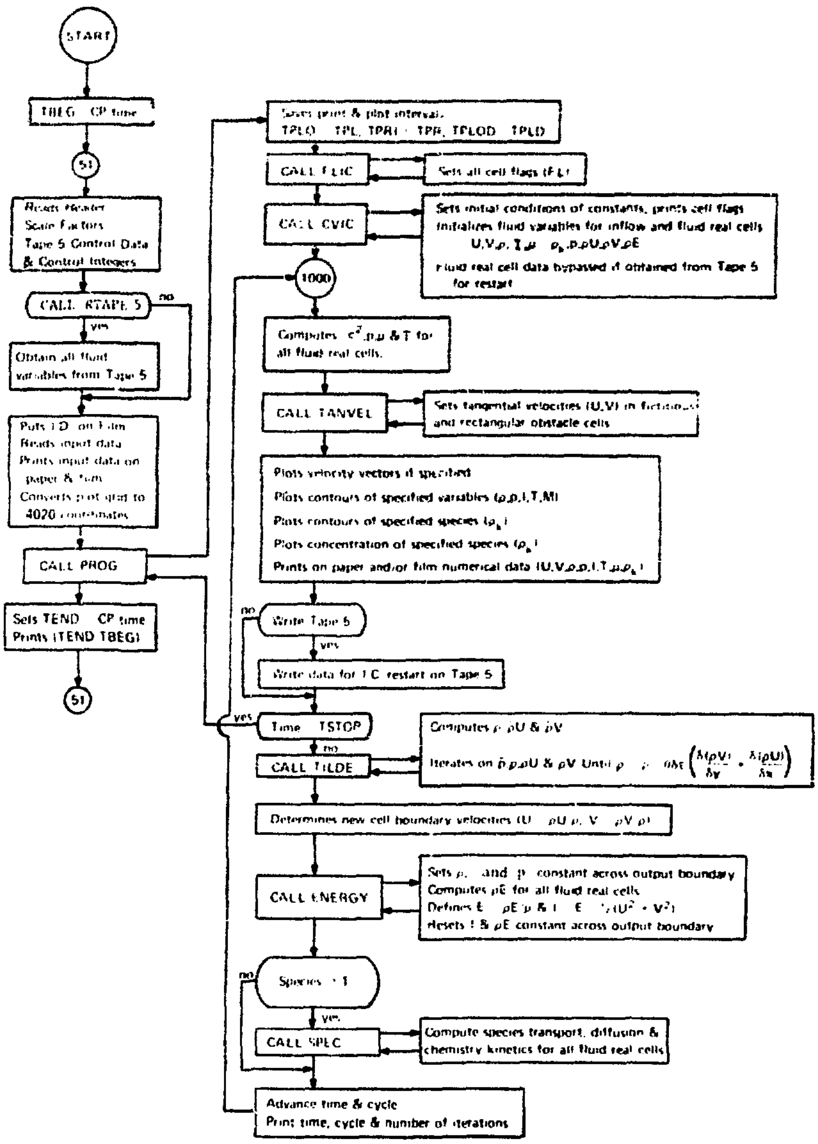
FORTRAN Symbol	Algebraic Symbol	Definition
P(I,J)	$p_{i,j}$	Final static pressure of the mixture. Computed in subroutine PRØG according to Eq. (3.36).
PB(I,J)	$\bar{p}_{i,j}$	Temporary static pressure of the mixture obtained by the pressure iteration. Computed in subroutine TILDE according to Eqs. (3.12).
PFAC	-	Term used in the pressure iteration for the calculation of the increment $\delta\bar{p}$ given by Eq. (3.11). Computed in subroutine CVIC as PFAC = $2\theta\phi\delta t^2$ $(\delta r^{-2} + \delta z^{-2})$.
PHI	ϕ	Input data card no. 8.
Q(L)	q_{ℓ}	Input data card no. 27.
RE(I,J)	$(\rho E)_{i,j}$	Total energy density. Computed in subroutine ENERGY according to Eq. (3.15) and updated in SPEC according to Eqs. (3.28) and (3.34).
RFLB(I)	-	Temporary storage for the mixture mass flux across the top boundary of cell (i,j) to be used later as the flux across the bottom boundary of cell (i,j+1).
RHØ(I,J)	$\rho_{i,j}$	Mass density of the mixture. Computed in subroutine TILDE according to Eq. (3.1).
RHØØ	ρ_o	Input data card no. 11.
RHØT(I,J)	$\tilde{\rho}_{i,j}$	Tilde value of the mixture density. Computed in subroutine TILDE according to Eq. (3.2).
RHØU(I,J)	$(\rho u)_{i+\frac{1}{2},j}$	Axial component of momentum density. Computed in subroutine TILDE according to Eq. (3.9).
RHØUT(I,J)	$(\tilde{\rho u})_{i+\frac{1}{2},j}$	Tilde value of the axial component of momentum density. Computed in subroutine TILDE according to Eq. (3.9).
RHØV(I,J)	$(\rho v)_{i,j+\frac{1}{2}}$	Radial component of momentum density. Computed in subroutine TILDE according to Eq. (3.6).
RHØVT(I,J)	$(\tilde{\rho v})_{i,j+\frac{1}{2}}$	Tilde value of the radial component of momentum density. Computed in subroutine TILDE according to Eq. (3.8).
RINBØ(N)	-	Input data card no. 30.
RINL1	-	Input data card no. 11.
RINL2	-	Input data card no. 11.
RINRØ(N)	-	Input data card no. 29.
RINT1	-	Input data card no. 11.
RINT2	-	Input data card no. 11.
RJ(J)	r_j	Radial coordinate of the center of the (J)th row of computational cells. Computed in subroutine CVIC as RJ(J) = (J-1)δr - δr/2.
RJ2(J)	$r_{j+\frac{1}{2}}$	Radial coordinate of the top boundary of (J)th row of computational cells. Computed in subroutine CVIC as RJ2(J) = (J-1)δr.
RRHØ	ρ_o^{-1}	Reciprocal of the initial density ρ. Computed in subroutine CVIC.
SECREQ	-	Computer time requested for the problem in seconds. Computed in program RICE from JØB card data.
SIE(I,J)	$I_{i,j}$	Specific internal energy. Computed in subroutine ENERGY according to Eq. (3.16) and updated in subroutine SPEC following the updates of the total energy density according to Eqs. (3.28) and (3.34).
SIEBØ(N)	-	Input data card no. 30.
SIEL1	-	Input data card no. 13.
SIEL2	-	Input data card no. 13.
SIEØ	-	Input data card no. 13.
SIERØ(N)	-	Input data card no. 29.
SIET1	-	Input data card no. 13.

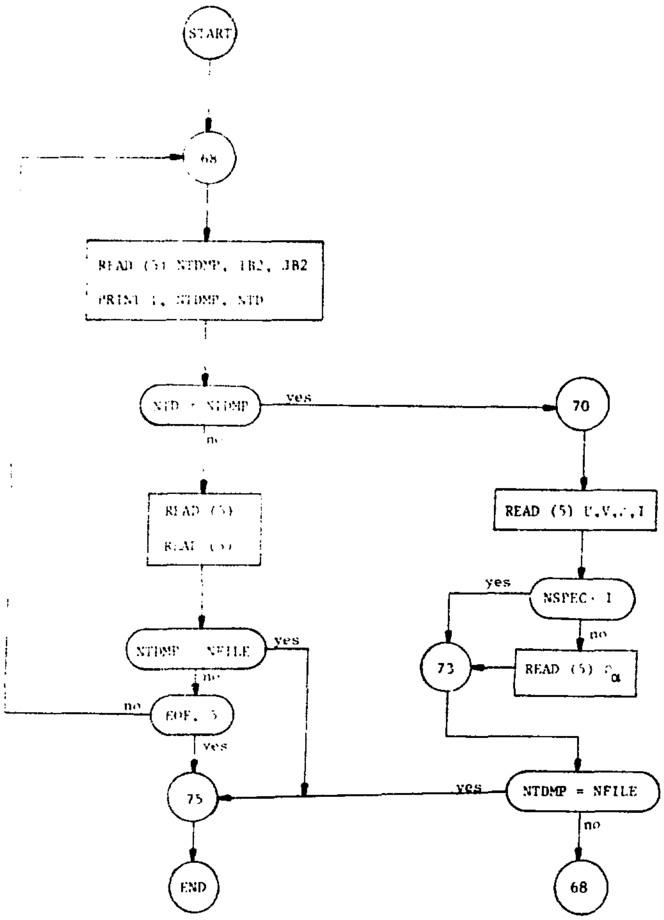
FORTRAN Symbol	Algebraic Symbol	Definition
SIET2	-	Input data card no. 13.
SLG	L_s	Input data card no. 4.
SM(K)	M_k	Input data card no. 22.
SMØ	M_s	Input data card no. 4.
STREB(I)	-	Temporary storage for the energy flux across the top boundary of cell (i,j) to be used later as the flux across the bottom boundary of cell (i,j+1).
TBEG	-	Time at the beginning of the program execution.
TE(I,J)	$T_{i,j}$	Computed in program RICE. Temperature computed according to Eq. (3.25) following the solutions of Eq. (3.21), (3.28), and (3.34).
TEM	T_s	Input data card no. 4.
TFAB(I,K)	-	Temporary storage for the mass flux of species k across the top boundary of cell (i,j) to be used later as the flux across the bottom boundary of cell (i,j+1).
THETA	-	Input data card no. 8.
TIME	-	Input data card no. 7.
TPL	-	Input data card no. 7.
TPLD	-	Input data card no. 7.
TPR	-	Input data card no. 7.
TSTØP	-	Input data card no. 7.
U(I,J)	$u_{i+\frac{1}{2},j}$	Axial component of velocity. Computed in subroutine PRØG according to Eq. (3.14).
UCØN	-	Factor used in the pressure iteration, Eq. (3.13).
UINL1	-	Input data card no. 10.
UINL2	-	Input data card no. 10.
UINRØ(N)	-	Input data card no. 29.
UØ	-	Input data card no. 10.
V(I,J)	$v_{i,j+\frac{1}{2}}$	Radial component of velocity. Computed in subroutine PRØG according to Eq. (3.14).
VCØN	-	Factor used in the pressure iteration, Eq. (3.13). Computed in CVIC as VCØN = $\rho\delta t/\delta r$.
VEL	u_s	Input data card no. 4.
VINBØ(N)	-	Input data card no. 30.
VINT1	-	Input data card no. 10.
VINT2	-	Input data card no. 10.
VØ	-	Input data card no. 10.
XI	ξ	Parameter used in the calculation of τ and β in Eqs. (3.5) and (3.20). Set in subroutine CVIC as XI = 1.
ZETAB(L)	ζ_ℓ	Input data card no. 27.
ZETAF(L)	ζ_ℓ	Input data card no. 27.

APPENDIX B

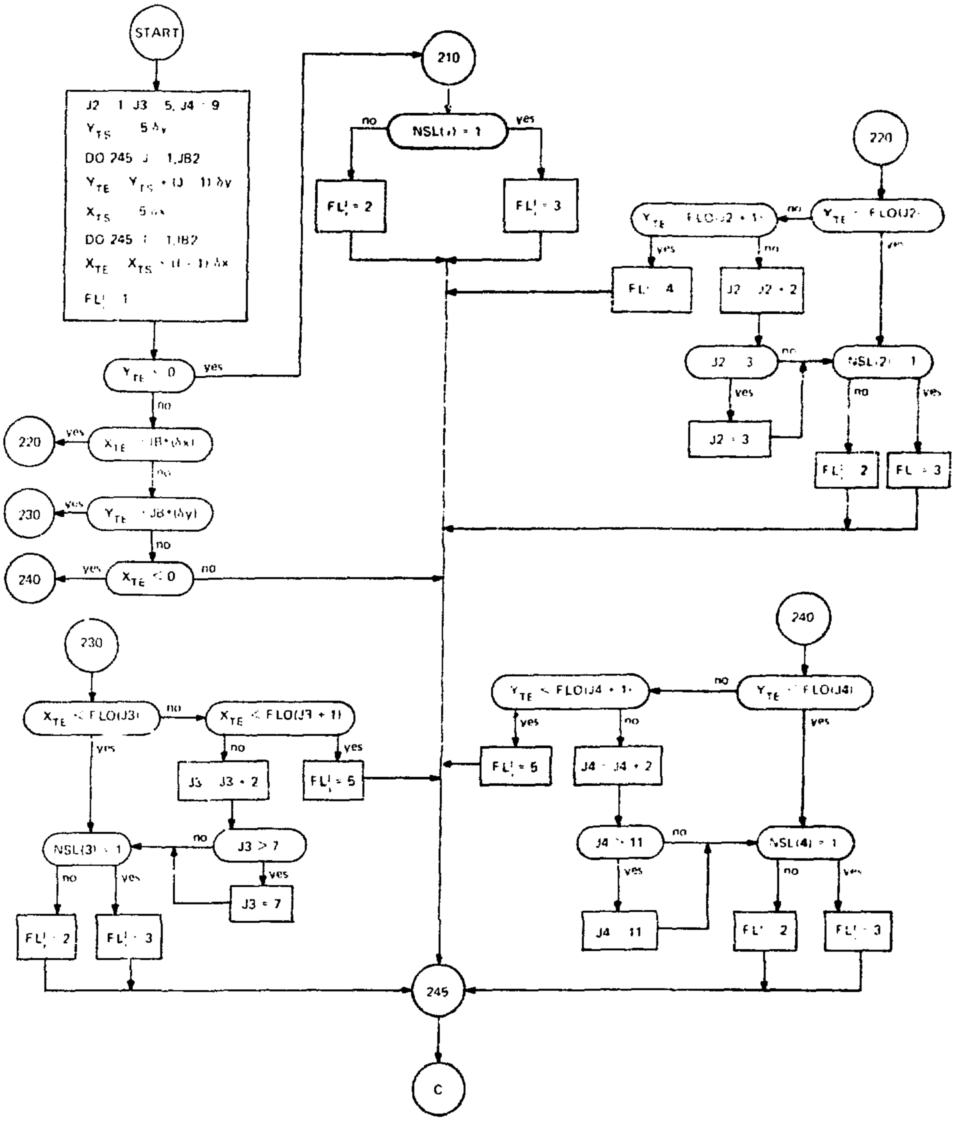
FLOW DIAGRAMS FOR RICE SUBROUTINES

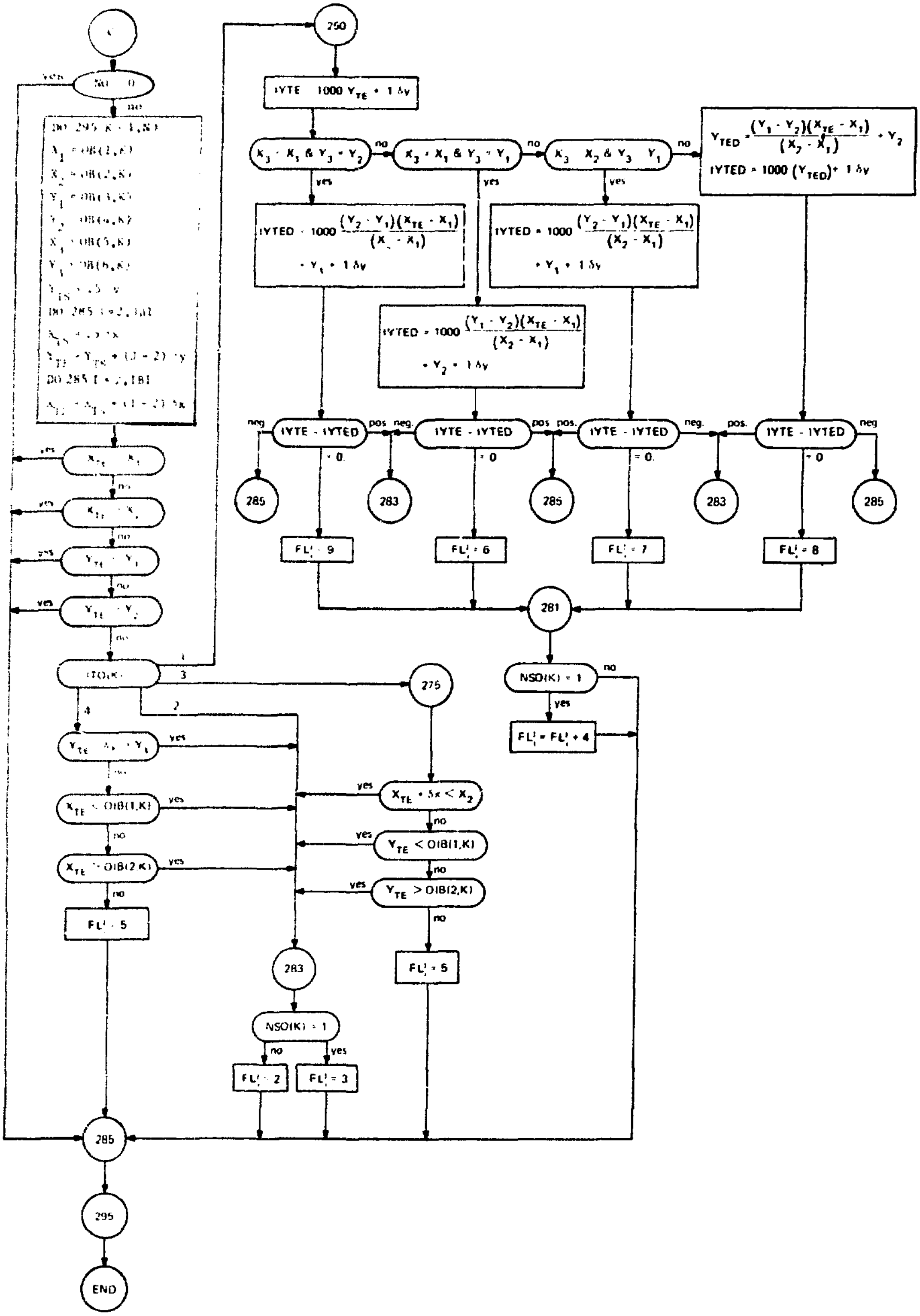
Flow diagrams for each subroutine of the RICE program are presented in this appendix except for the three subroutines VPLØT, CNPLØT, and CØPLØT which deal entirely with plotting. A FØRTRAN listing of these subroutines is given in Appendix C along with the listings of the other subroutines. The flow diagrams presented here are for RICE, PRØG, RTAPE5, FLIC, CVIC, TANVEL, ENERGY, and SPEC in that order.

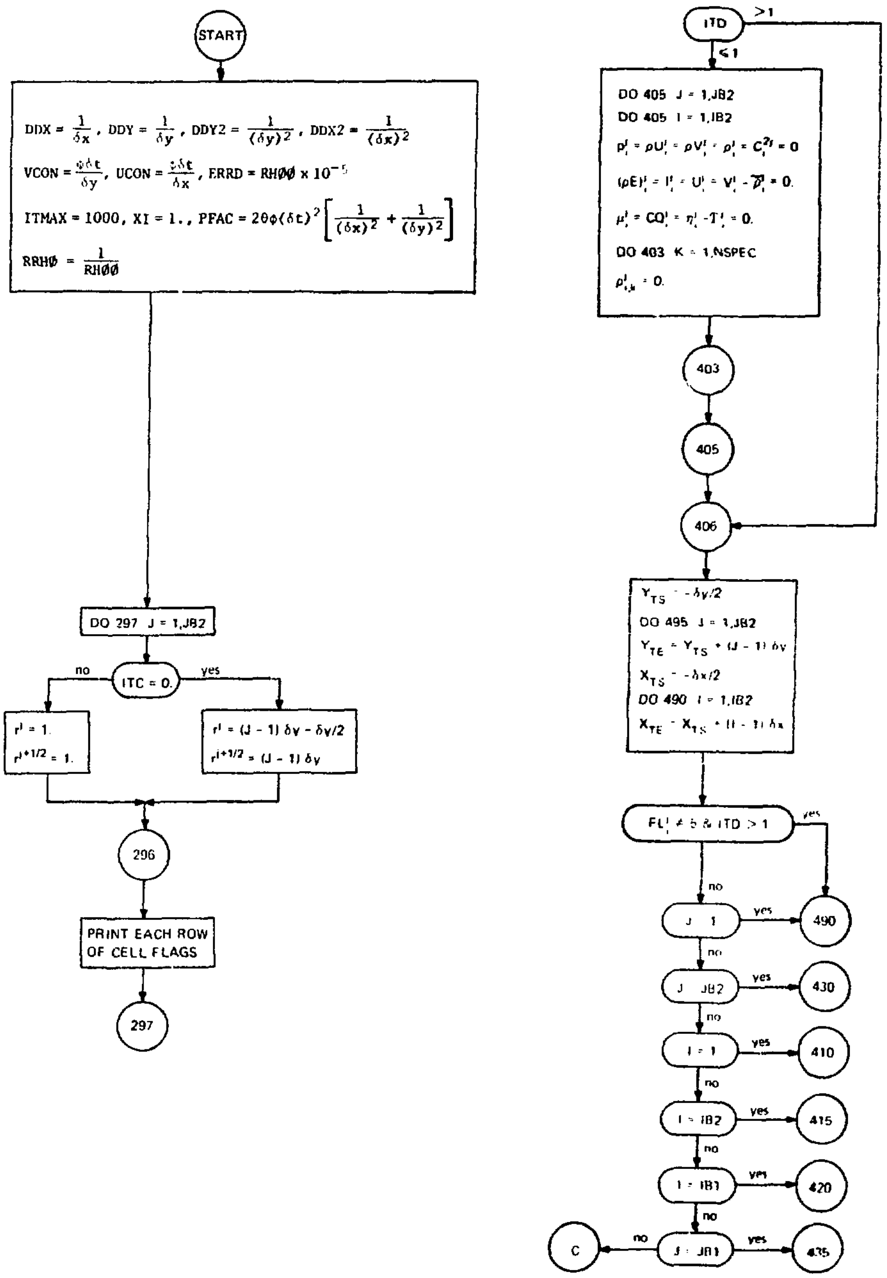


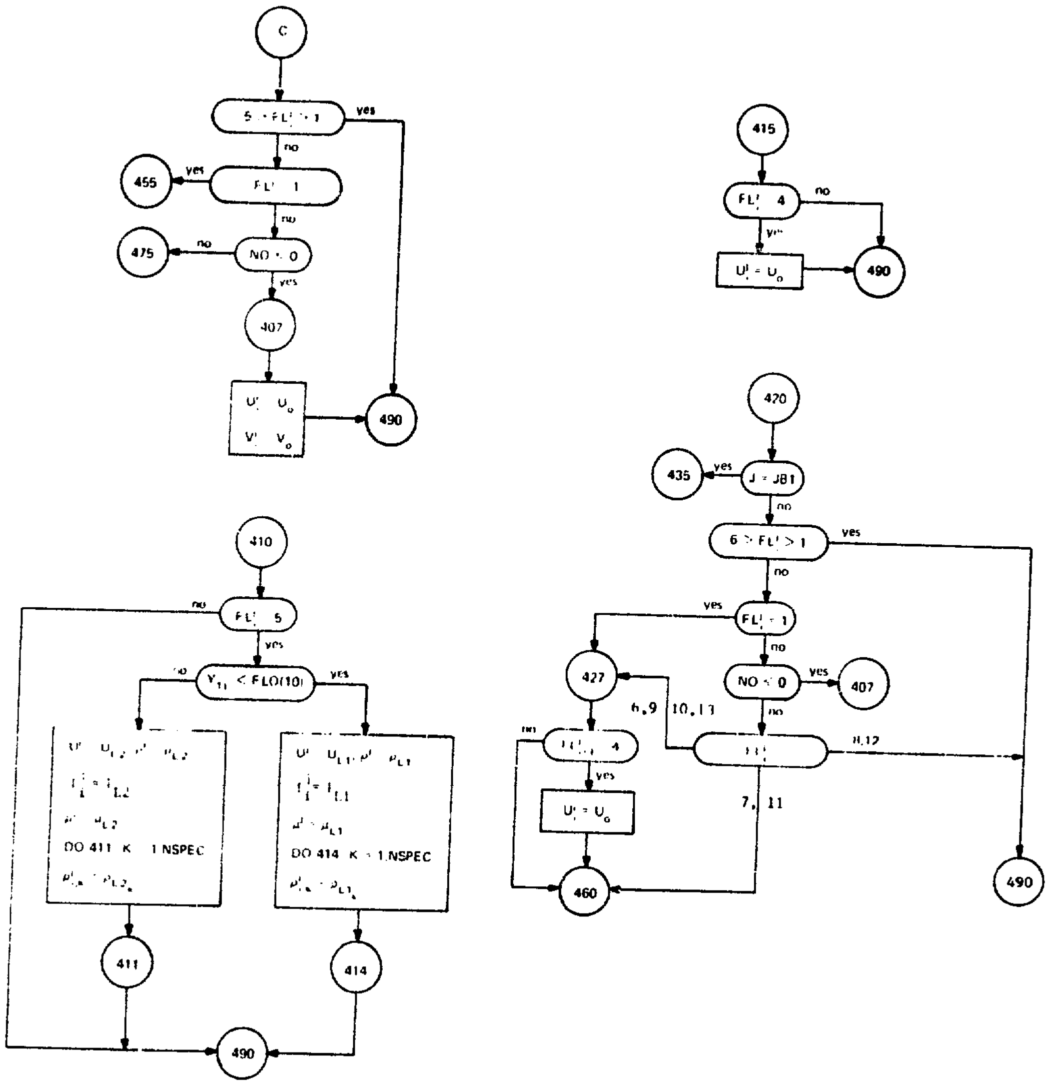

START
68
READ (5) NTDMP, IB2, JB2
PRINT 1, NTDMP, NTD
yes
no
70
READ (5)
READ (5)
READ (5) U,V,
yes
no
NSPEC
yes
no
73
READ (5)
EOF, 5
yes
yes
75
NTDMP = NFILE
no
END
68

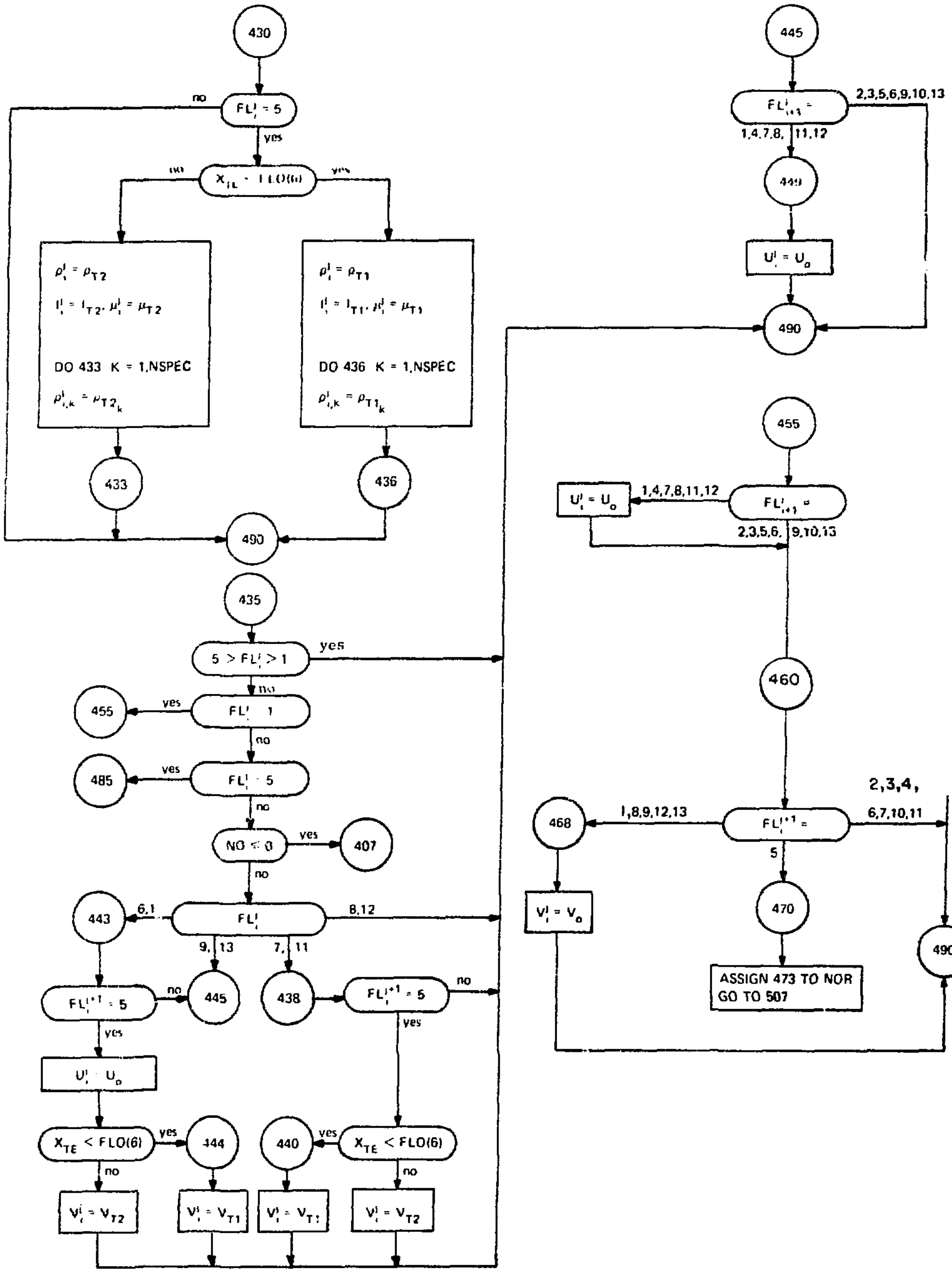
SUBROUTINE FLIC
(Sets cell flags)



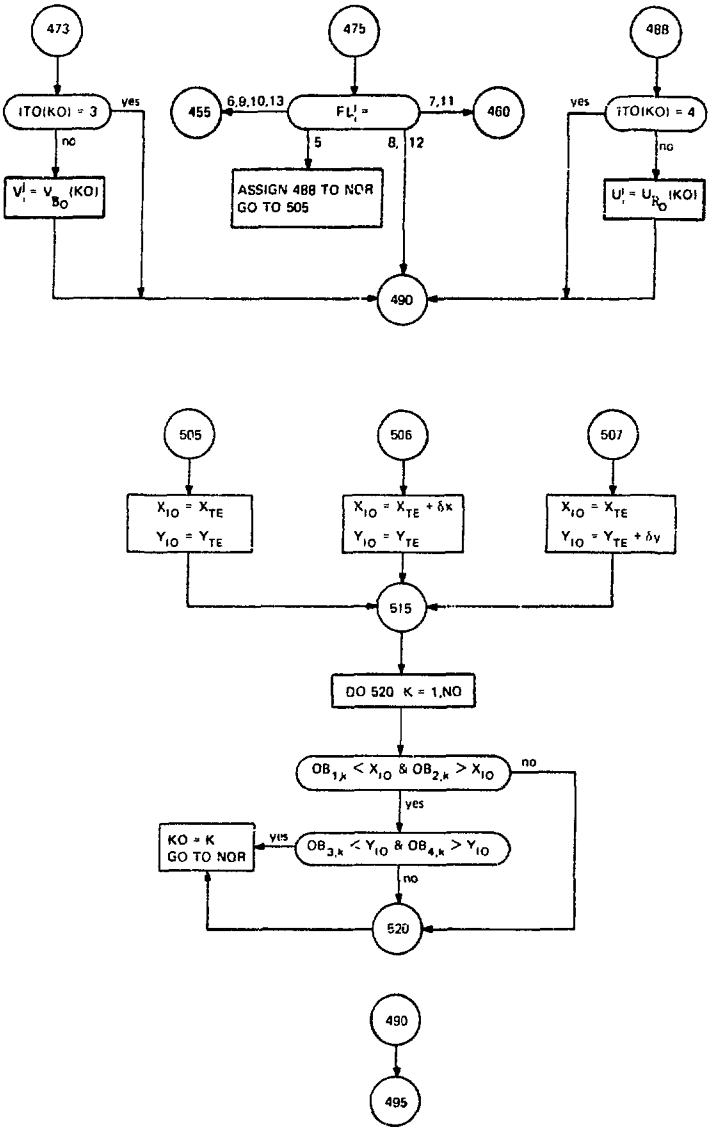

250
IYTE = 1000 Y_{TE} + 1 δy
X_3 = X_1 & Y_3 = Y_2
X_3 = X_1 & Y_3 = Y_1
X_3 = X_2 & Y_3 = Y_1
IYTED = 1000 $\frac{(Y_2 - Y_1)(X_{TE} - X_1)}{(X_2 - X_1)}$ + Y_1 + 1 δy
IYTED = 1000 $\frac{(Y_1 - Y_2)(X_{TE} - X_1)}{(X_2 - X_1)}$ + Y_2 + 1 δy
Y_{TED} = $\frac{(Y_1 - Y_2)(X_{TE} - X_1)}{(X_2 - X_1)}$ + Y_2
IYTED = 1000 (Y_{TED}) + 1 δy
IYTE - IYTED
neg
pos
= 0
285
283
281
NSO(K) = 1
yes
no
275
283
295
END
X_{TE} + δx < X_2
Y_{TE} < OIB(1,K)
Y_{TE} > OIB(2,K)
X_{TE} < OIB(1,K)
X_{TE} > OIB(2,K)
ITO(K)
DO 295 K = 1,N
DO 285 J = 2,181
DO 285 I = 2,181

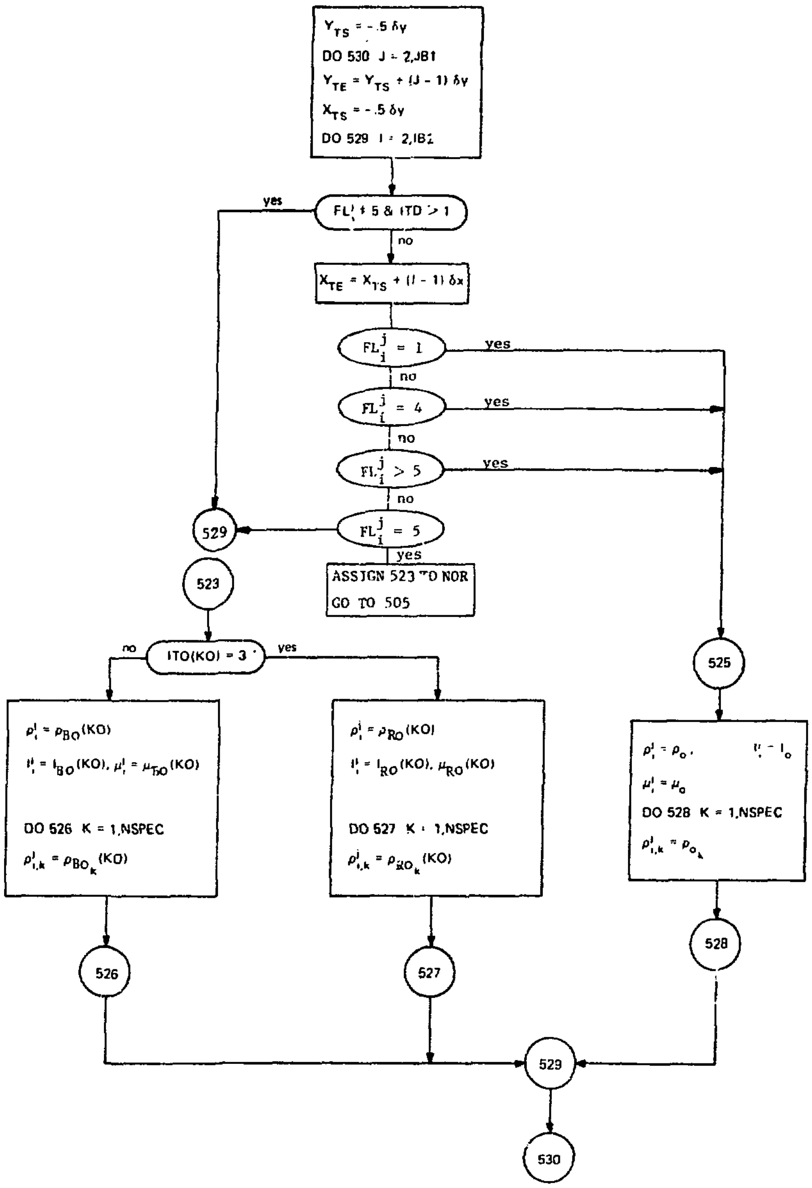

START
DDX = 1/δx , DDY = 1/δy , DDY2 = 1/(δy)² , DDX2 = 1/(δx)²
VCON = φδt/δy , UCON = φδt/δx , ERRD = RHØØ x 10⁻⁵
ITMAX = 1000, XI = 1., PFAC = 20φ(δt)² [1/(δx)² + 1/(δy)²]
RRHØ = 1/RHØØ
DO 297 J = 1,JB2
ITC = 0.
no
yes
r = 1.
r = 1.
r = (J - 1) δy - δy/2
r = (J - 1) δy
296
PRINT EACH ROW
OF CELL FLAGS
297
ITD
>1
≤1
DO 405 J = 1,JB2
DO 405 I = 1,IB2
DO 403 K = 1,NSPEC
403
405
406
DO 495 J = 1,JB2
DO 490 I = 1,IB2
yes
no
490
430
410
415
420
435
C


C
yes
no
455
yes
no
475
no
NO
yes
407
490
415
no
yes
490
420
435
yes
J = JB1
no
6 > FL > 1
yes
no
yes
427
NO
yes
407
no
6,9
10,13
8,12
no
yes
7, 11
460
490
410
no
yes
no
yes
DO 411 K = 1,NSPEC
DO 414 K = 1,NSPEC
411
414
490

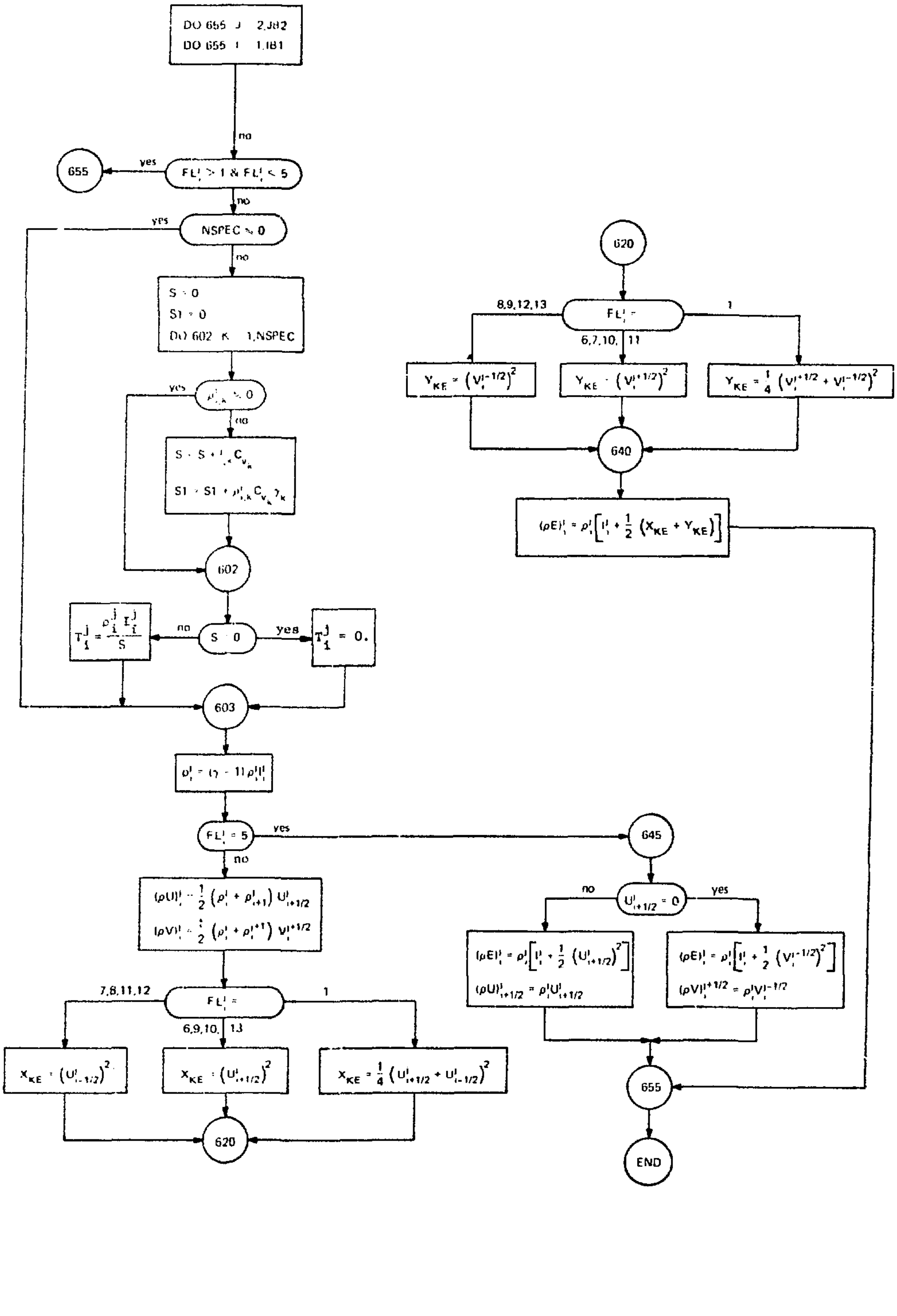

430
no
yes
DO 433 K = 1,NSPEC
DO 436 K = 1,NSPEC
433
436
490
435
5 > FL > 1
yes
no
455
485
NO ≤ 0
407
443
6,1
FL
8,12
9, 13
7, 11
445
438
no
yes
X_TE < FLO(6)
444
440
445
2,3,5,6,9,10,13
1,4,7,8,
11,12
449
U = U_o
490
455
1,4,7,8,11,12
2,3,5,6,
9,10,13
460
2,3,4,
468
1,8,9,12,13
6,7,10,11
5
470
490
V = V_o
ASSIGN 473 TO NOR
GO TO 507

CVIC continue (3)



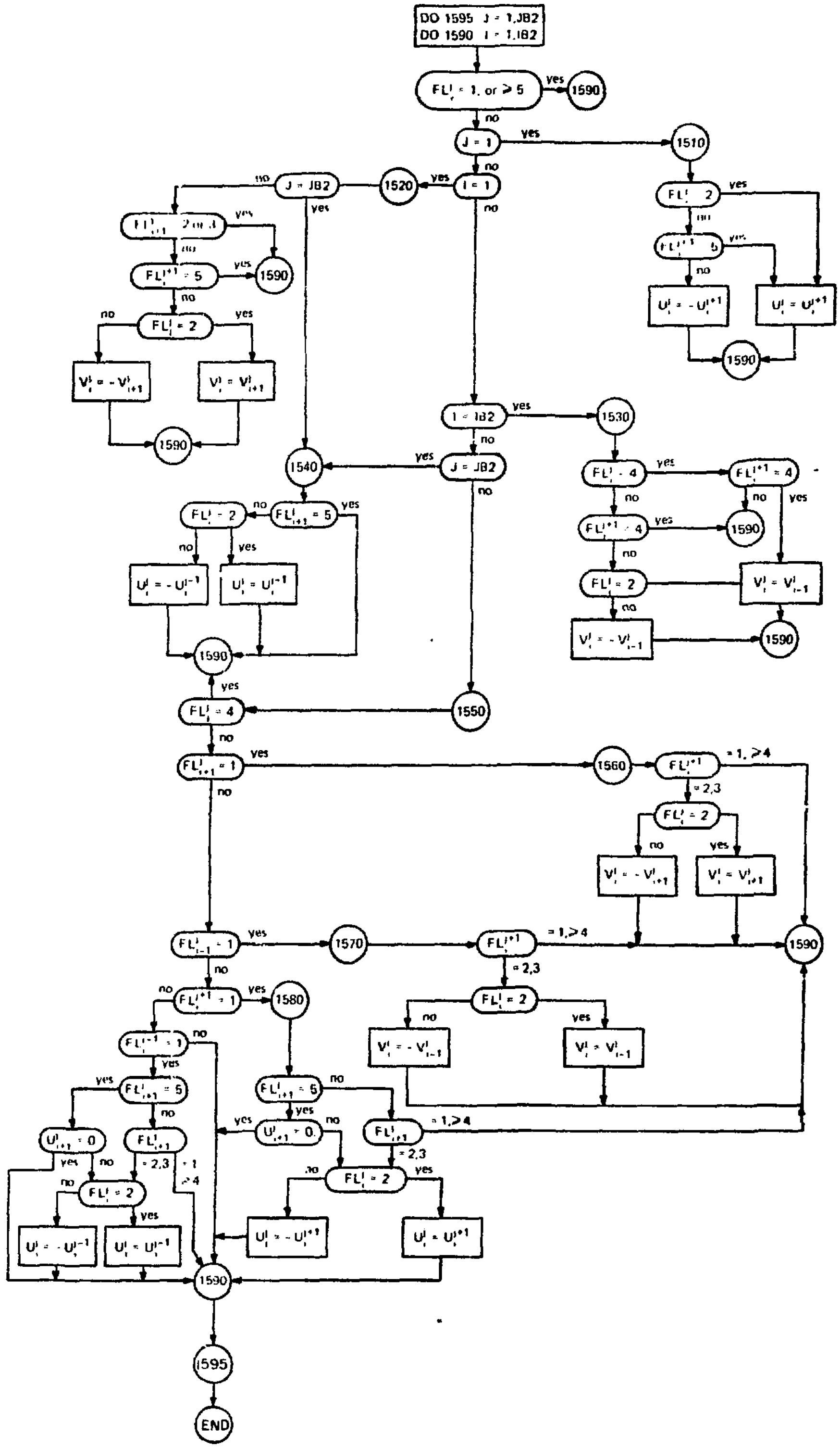


$Y_{TS} = -.5\,\delta y$
DO 530 J = 2,JB1
$Y_{TE} = Y_{TS} + (J-1)\,\delta y$
$X_{TS} = -.5\,\delta y$
DO 529 I = 2,IB1
yes
$FL_i^j \neq 5$ & ITD > 1
no
$X_{TE} = X_{TS} + (I-1)\,\delta x$
$FL_i^j = 1$
yes
no
$FL_i^j = 4$
yes
no
$FL_i^j > 5$
yes
no
529
$FL_i^j = 5$
yes
ASSIGN 523 TO NOR
GO TO 505
523
no
ITO(KO) = 3
yes
525
$\rho_i^j = \rho_{BO}(KO)$
$I_i^j = I_{BO}(KO)$, $\mu_i^j = \mu_{BO}(KO)$
DO 526 K = 1,NSPEC
$\rho_{i,k}^j = \rho_{BO_k}(KO)$
$\rho_i^j = \rho_{RO}(KO)$
$I_i^j = I_{RO}(KO)$, $\mu_{RO}(KO)$
DO 527 K = 1,NSPEC
$\rho_{i,k}^j = \rho_{RO_k}(KO)$
$\rho_i^j = \rho_o$, $I_i^j = I_o$
$\mu_i^j = \mu_o$
DO 528 K = 1,NSPEC
$\rho_{i,k}^j = \rho_{o_k}$
526
527
528
529
530

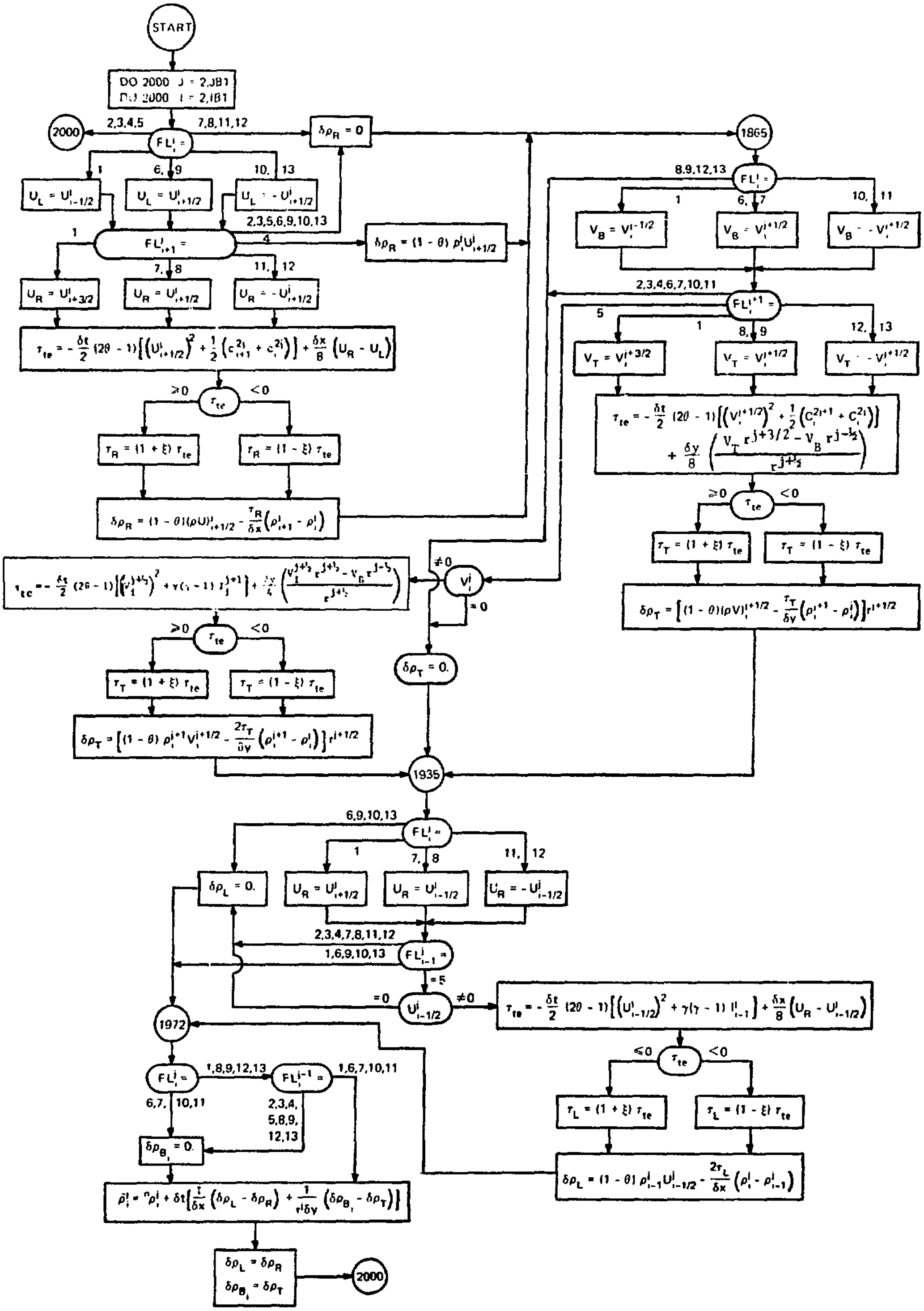

DO 655 J 2,J82
DO 655 I 1,IB1
no
yes
655
no
yes
NSPEC
no
S 0
S1 0
DO 602 K 1,NSPEC
yes
no
602
no
S 0
yes
603
yes
645
no
7,8,11,12
1
6,9,10, 13
620
620
8,9,12,13
1
6,7,10, 11
640
no
yes
655
END

SUBROUTINE TANVEL

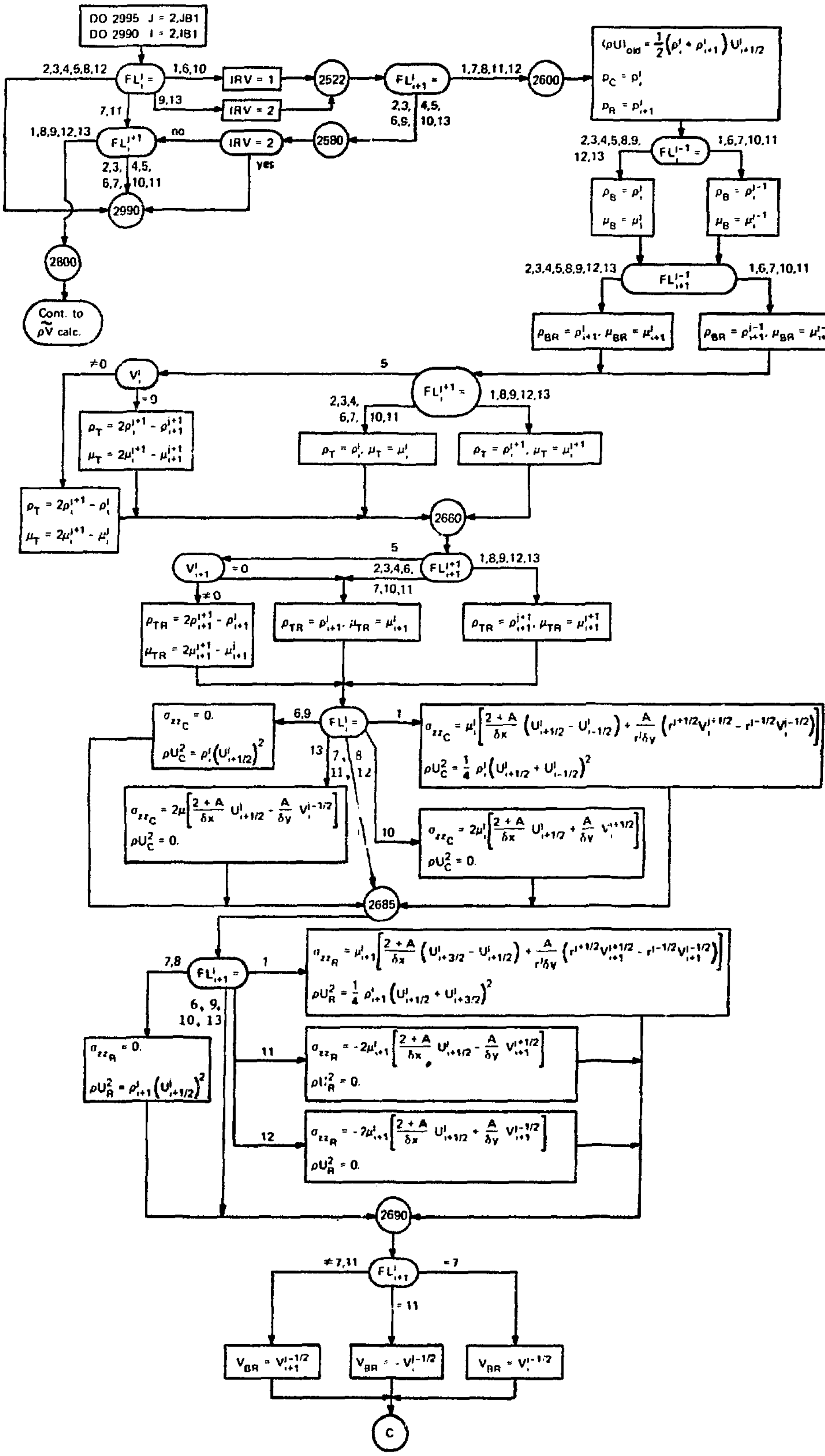
(Setting tangential velocities in fictitious and rectangular obstacle cells)

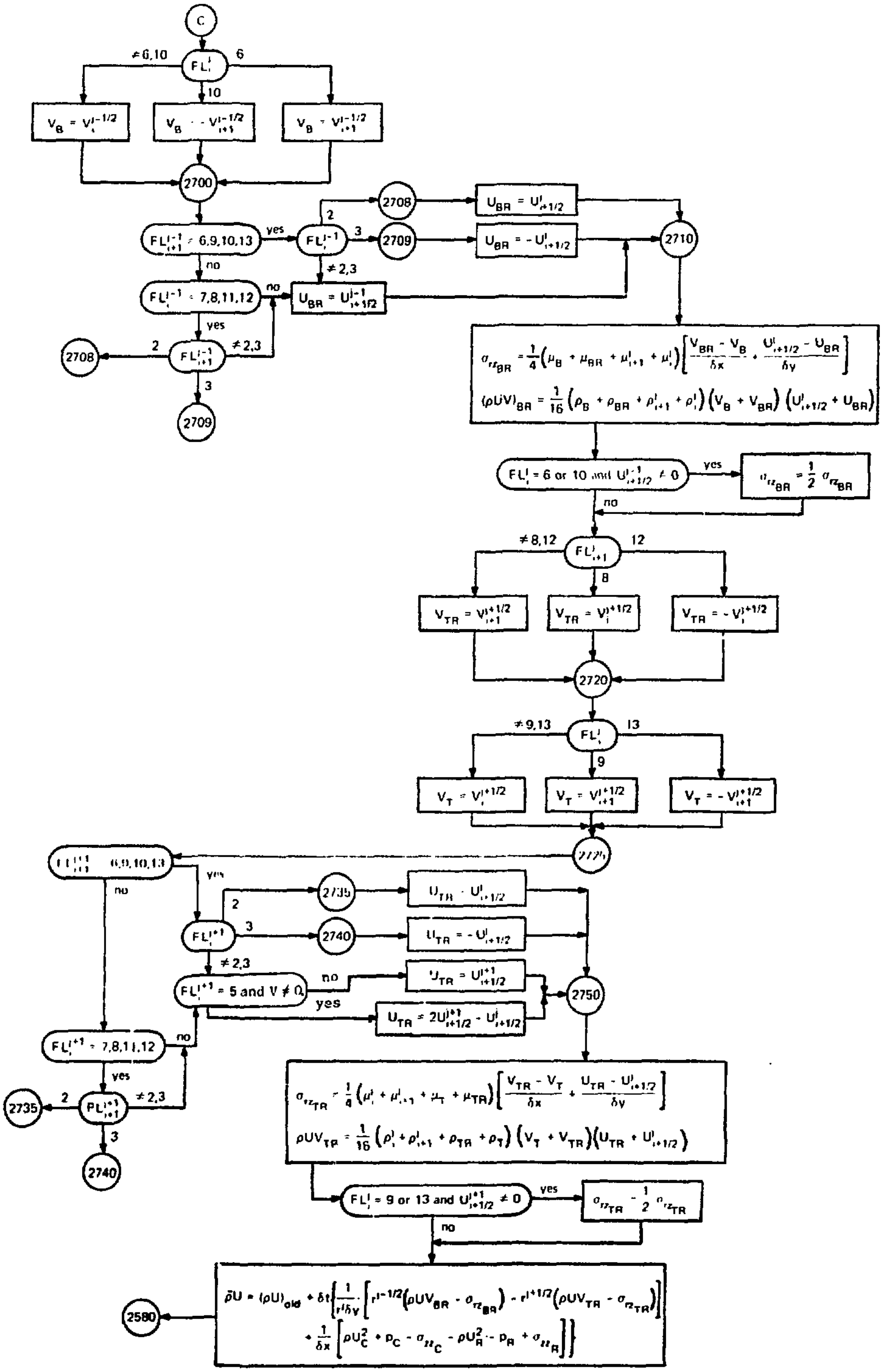


SUBROUTINE TILDE
($\tilde{\rho}$ Calculation)

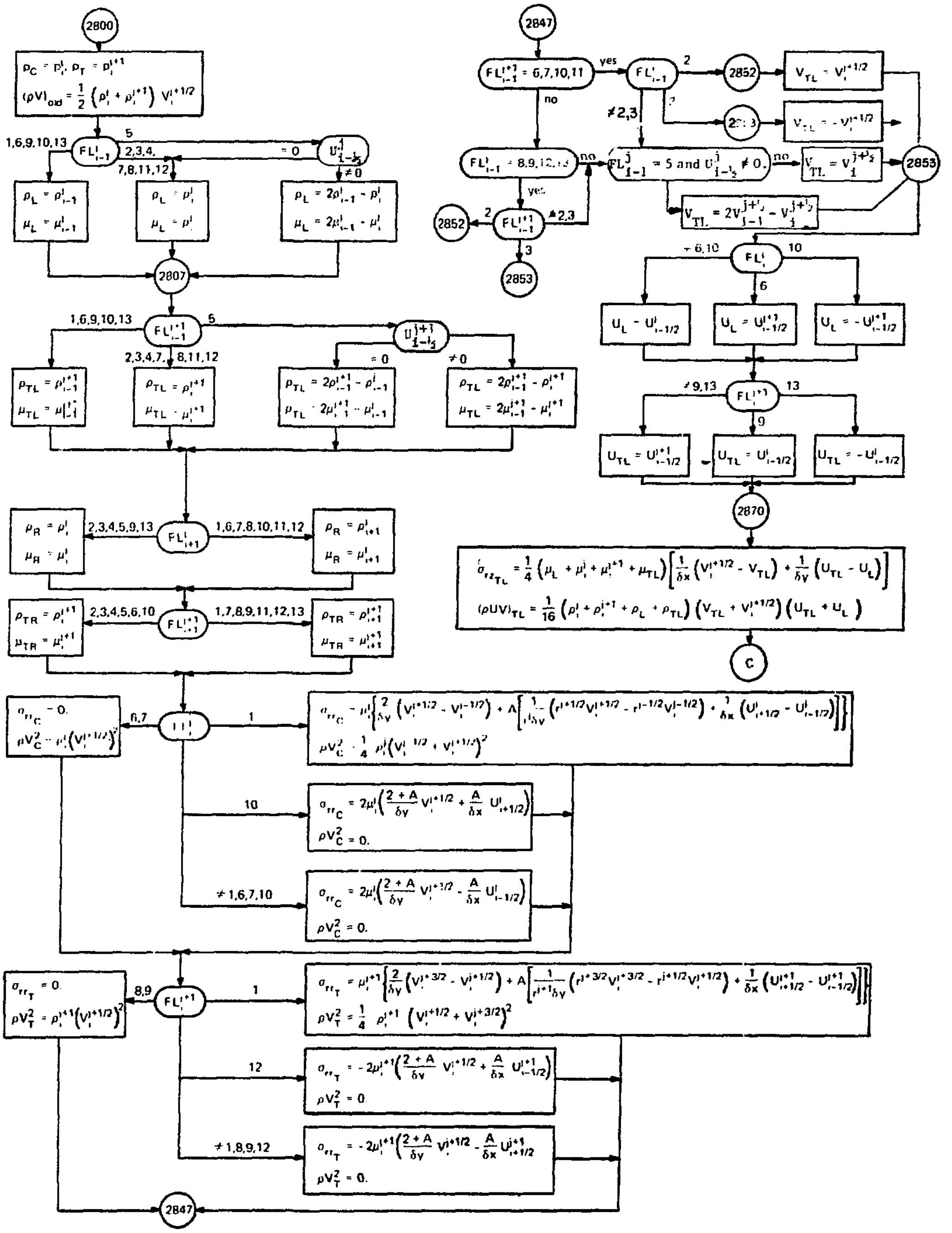


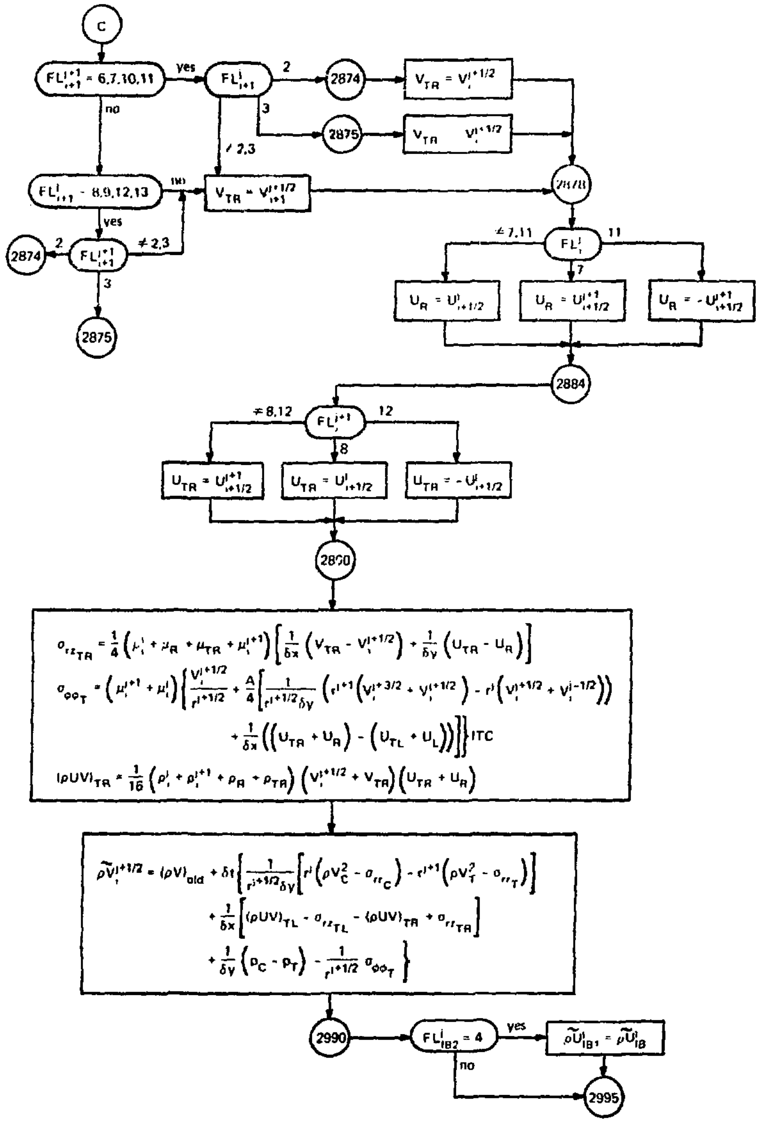
TILDE continue (1)
($\widetilde{\rho U}$ Calculation)



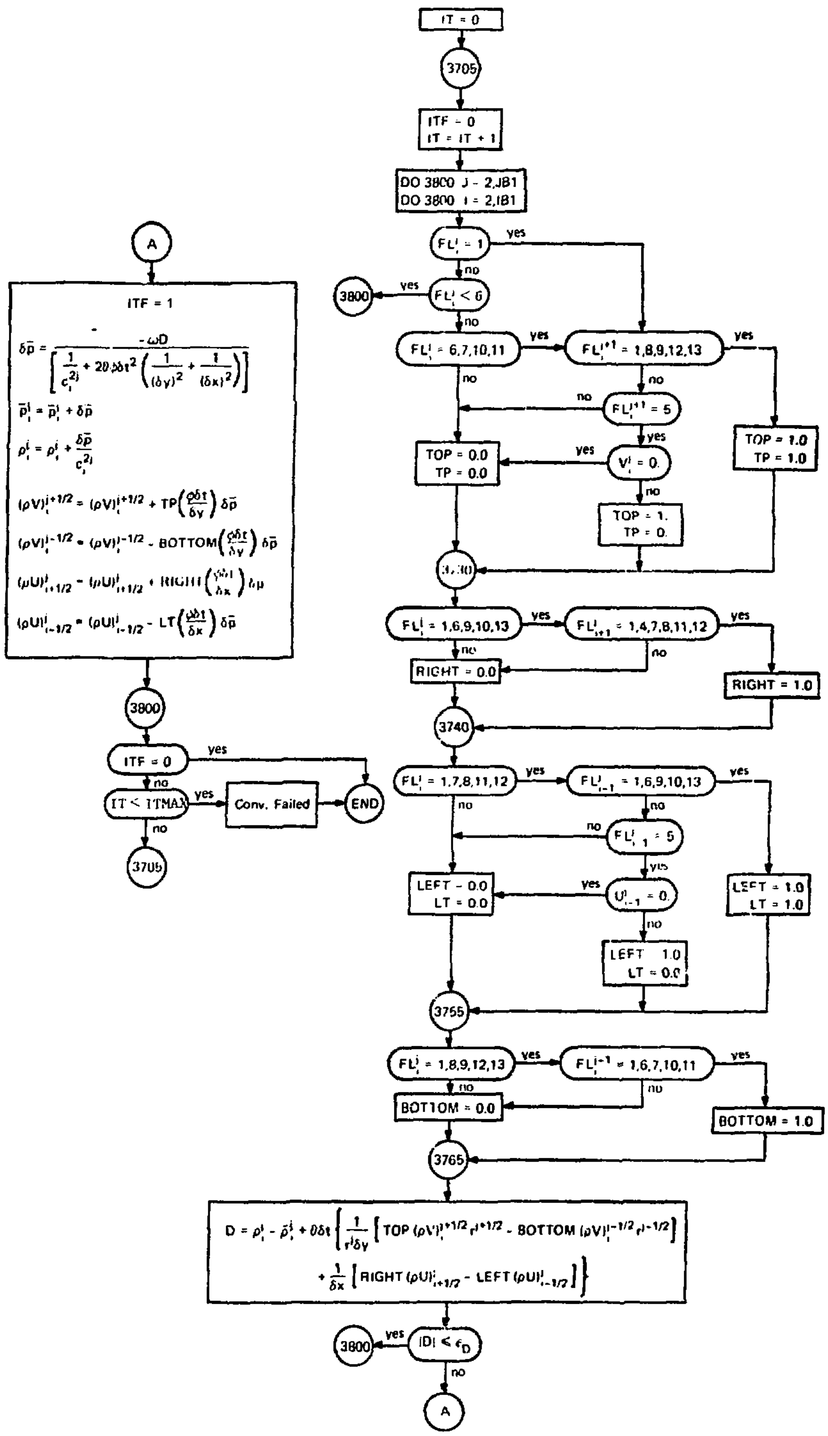

C
FL
≠6,10
6
10
2700
yes
no
2708
2709
2710
7,8,11,12
≠2,3
2
3
2720
≠8,12
12
8
≠9,13
13
9
2725
2735
2740
2750
5 and V ≠ 0
2580

TILDE continue (3)
($\rho\tilde{V}$ Calculation)

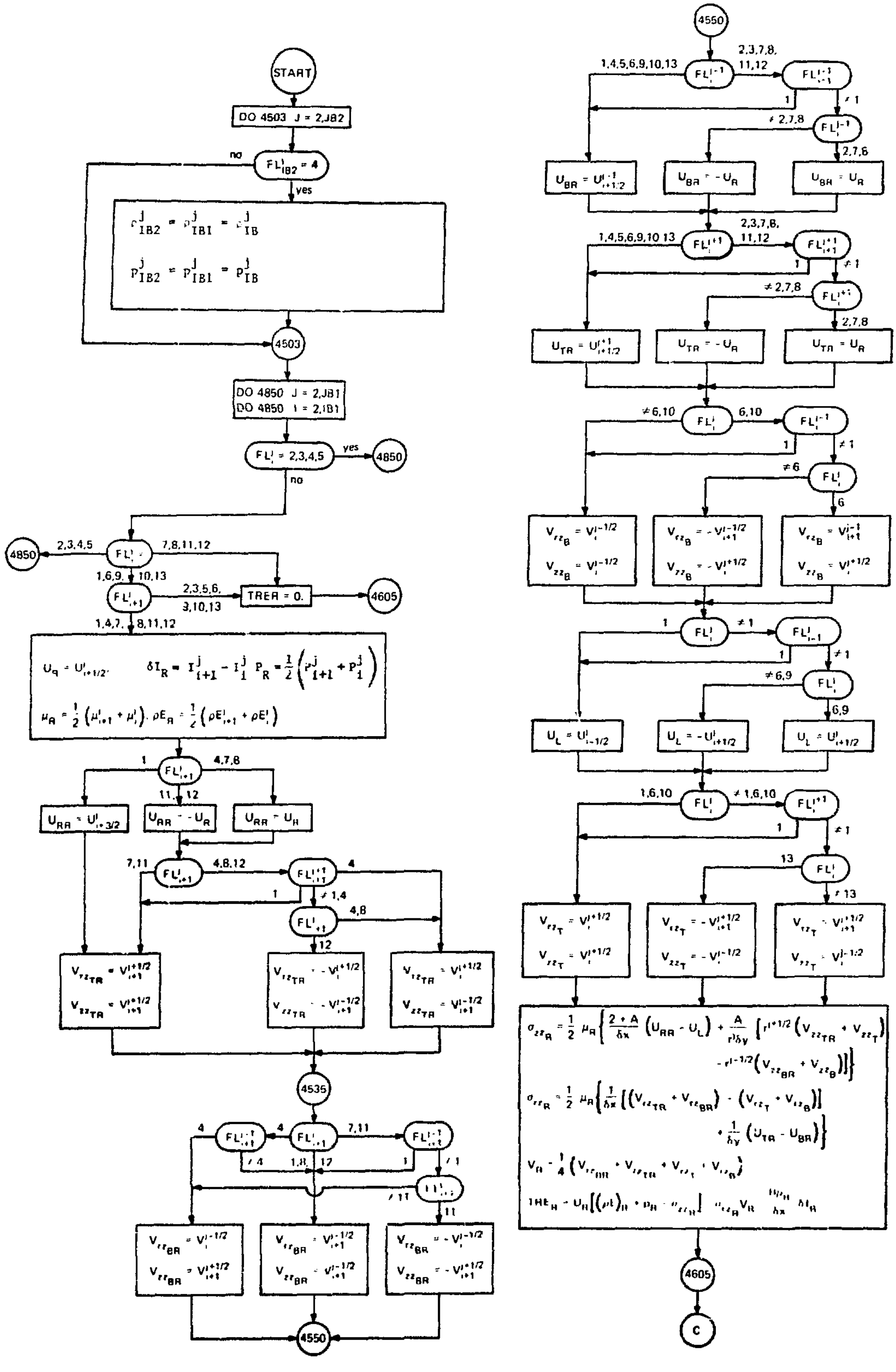


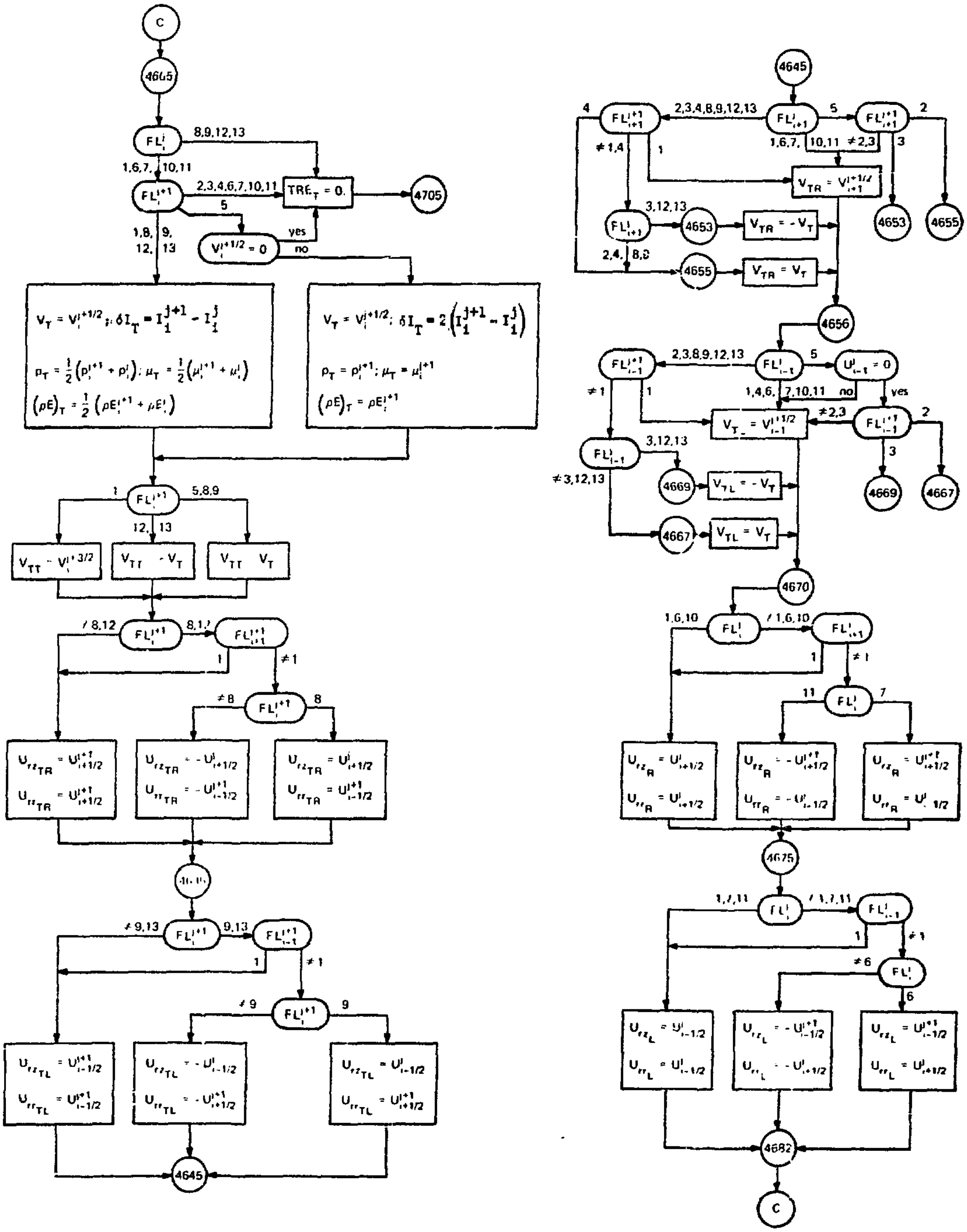


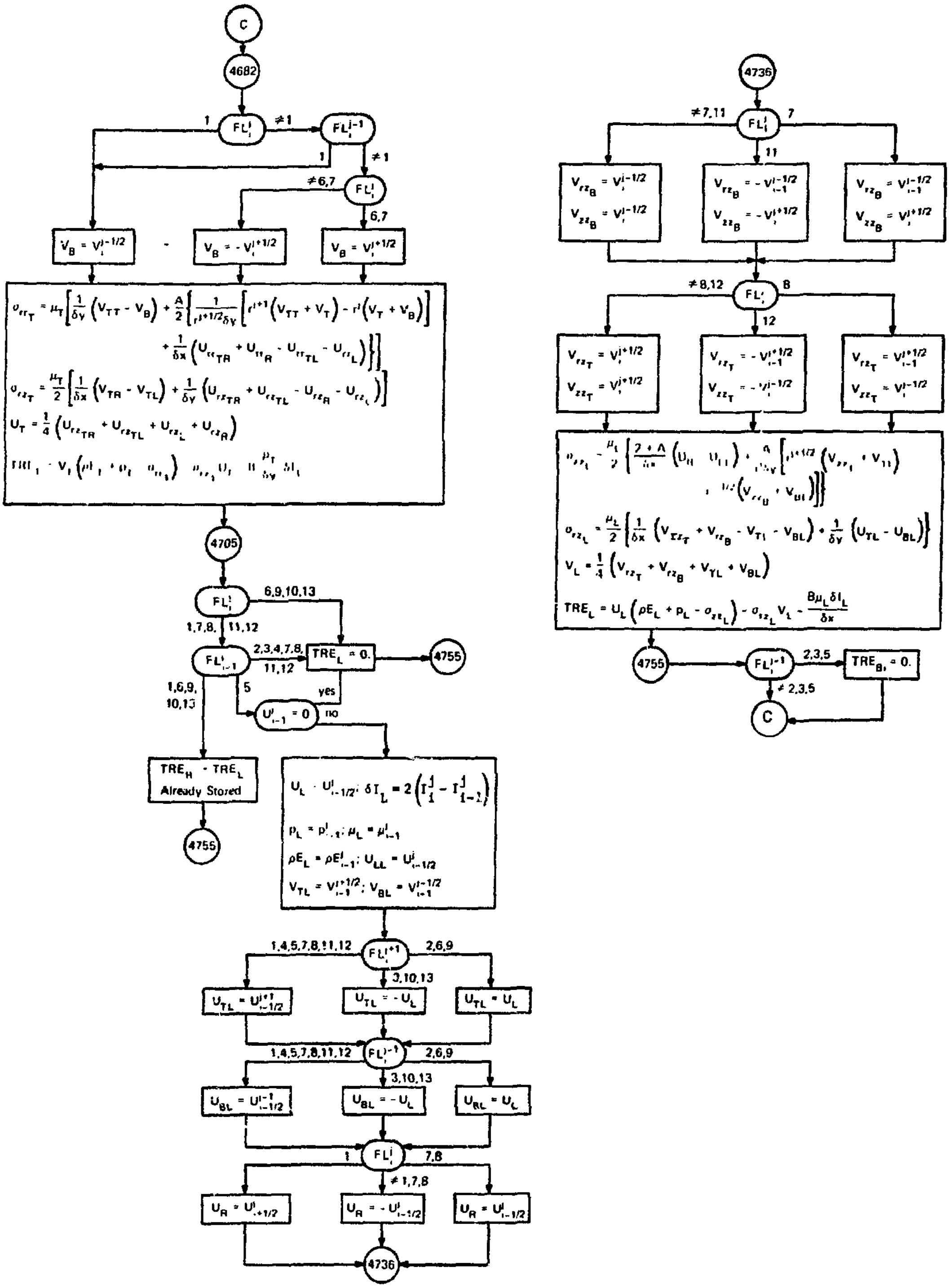
TILDE continue (5)
(Iteration Section)

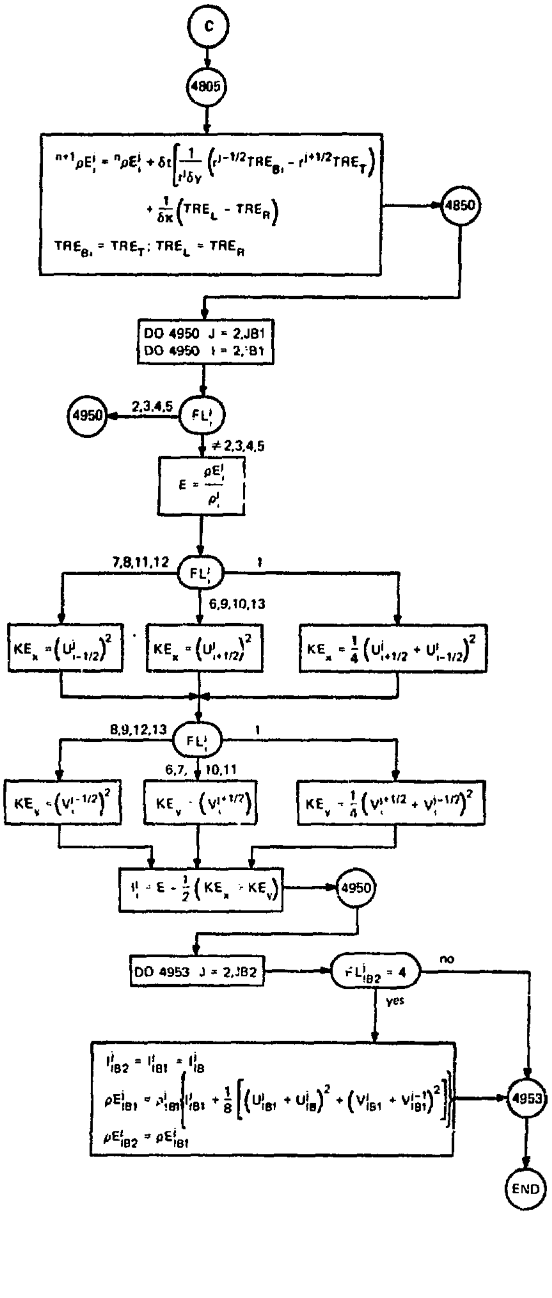


SUBROUTINE ENERGY
((ρE) & I Calculation)

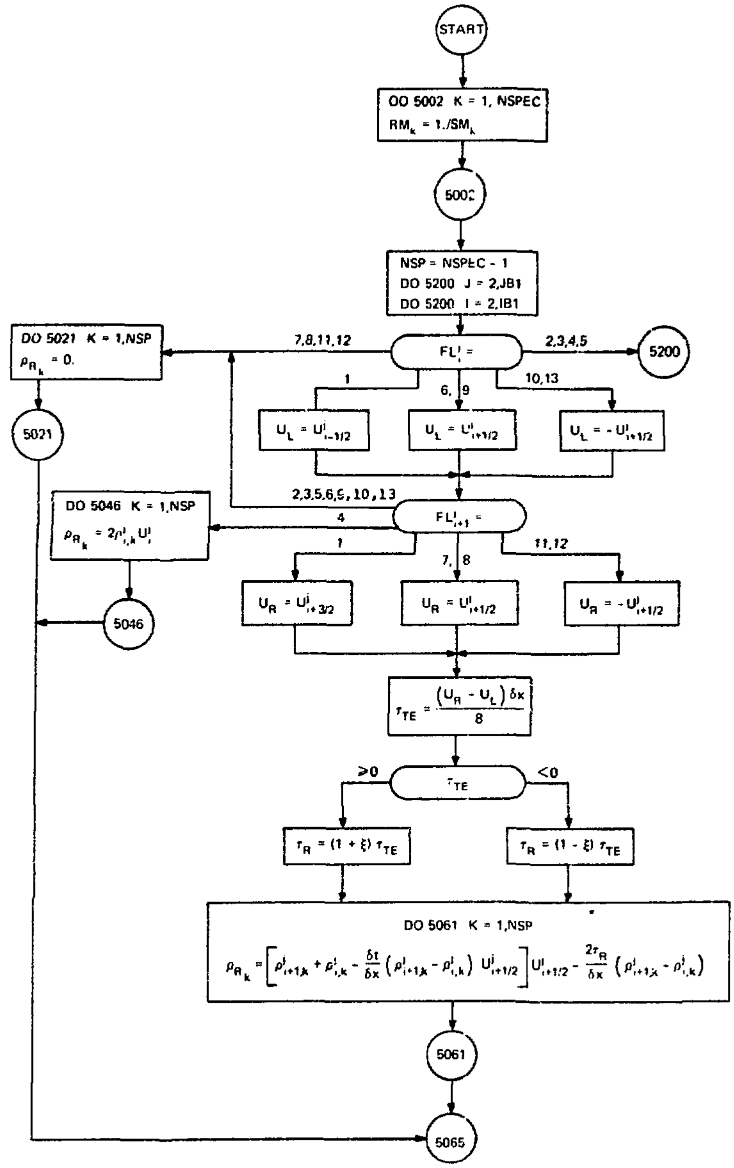


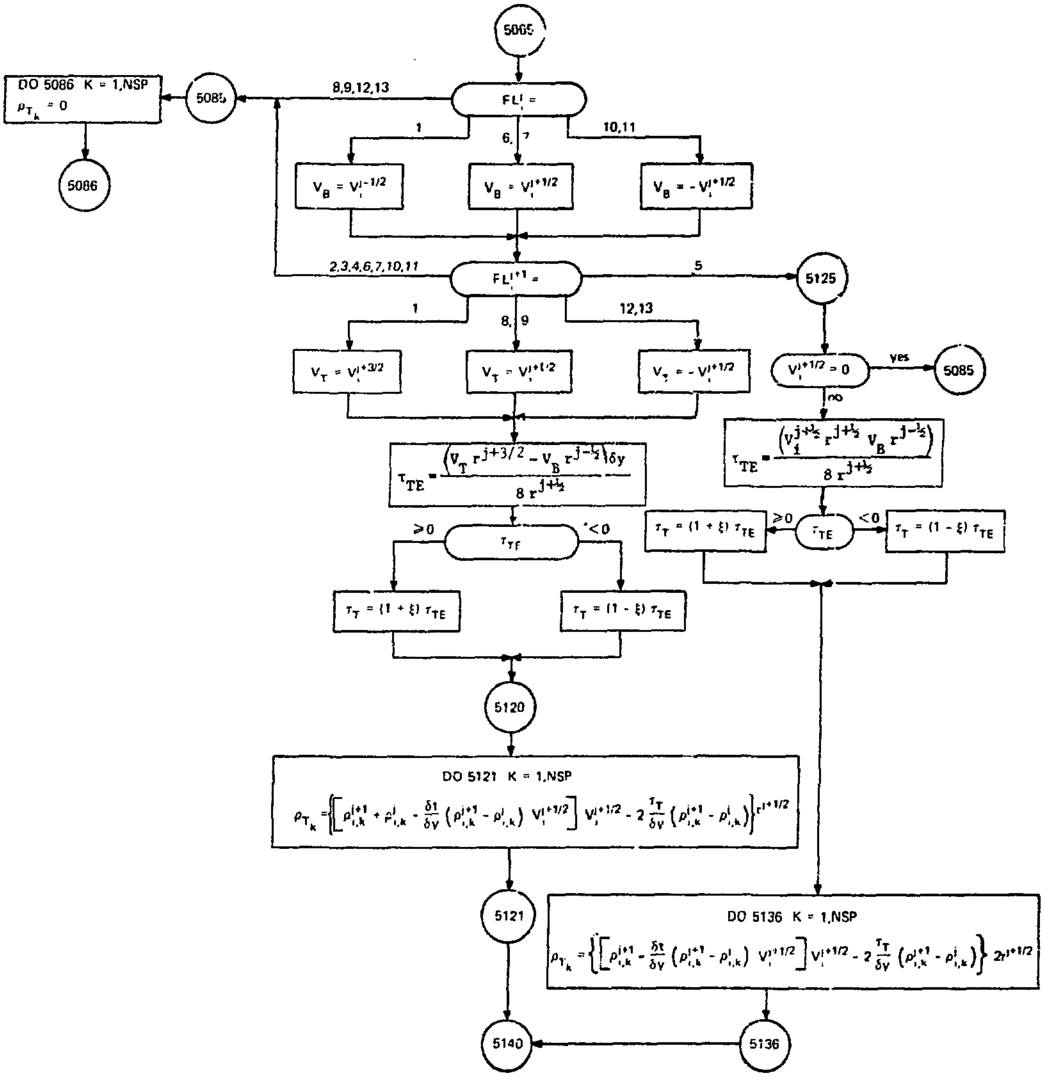


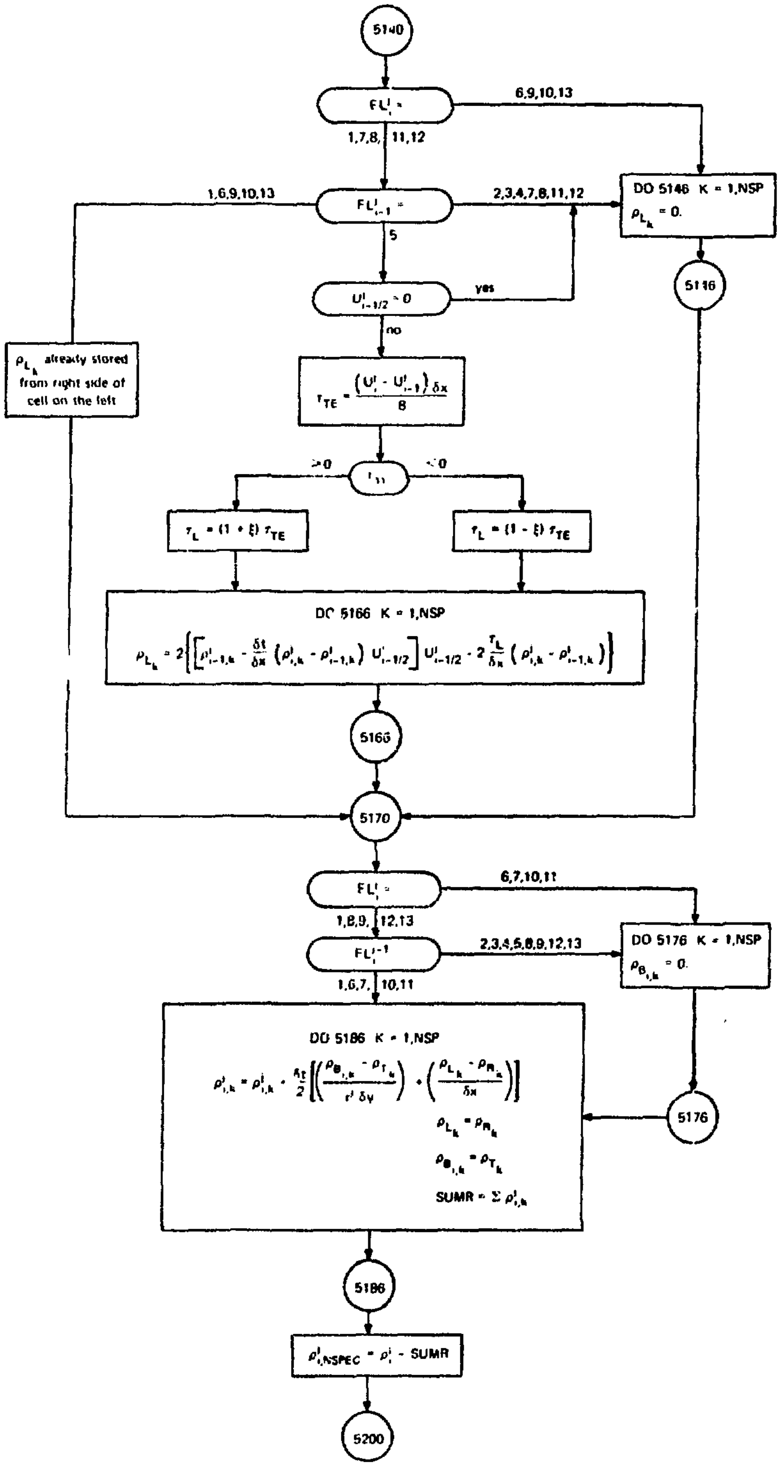

C
4682
4705
4755
4736
TRE$_L$ = 0.
TRE$_R$ - TRE$_L$ Already Stored
TRE$_{B}$ = 0.
yes
no


C
4805
TRE_B' = TRE_T; TRE_L = TRE_R
4850
DO 4950 J = 2,JB1
DO 4950 I = 2,IB1
FL
2,3,4,5
4950
≠2,3,4,5
7,8,11,12
6,9,10,13
1
8,9,12,13
6,7, 10,11
4950
DO 4953 J = 2,JB2
FL_IB2 = 4
no
yes
4953
END

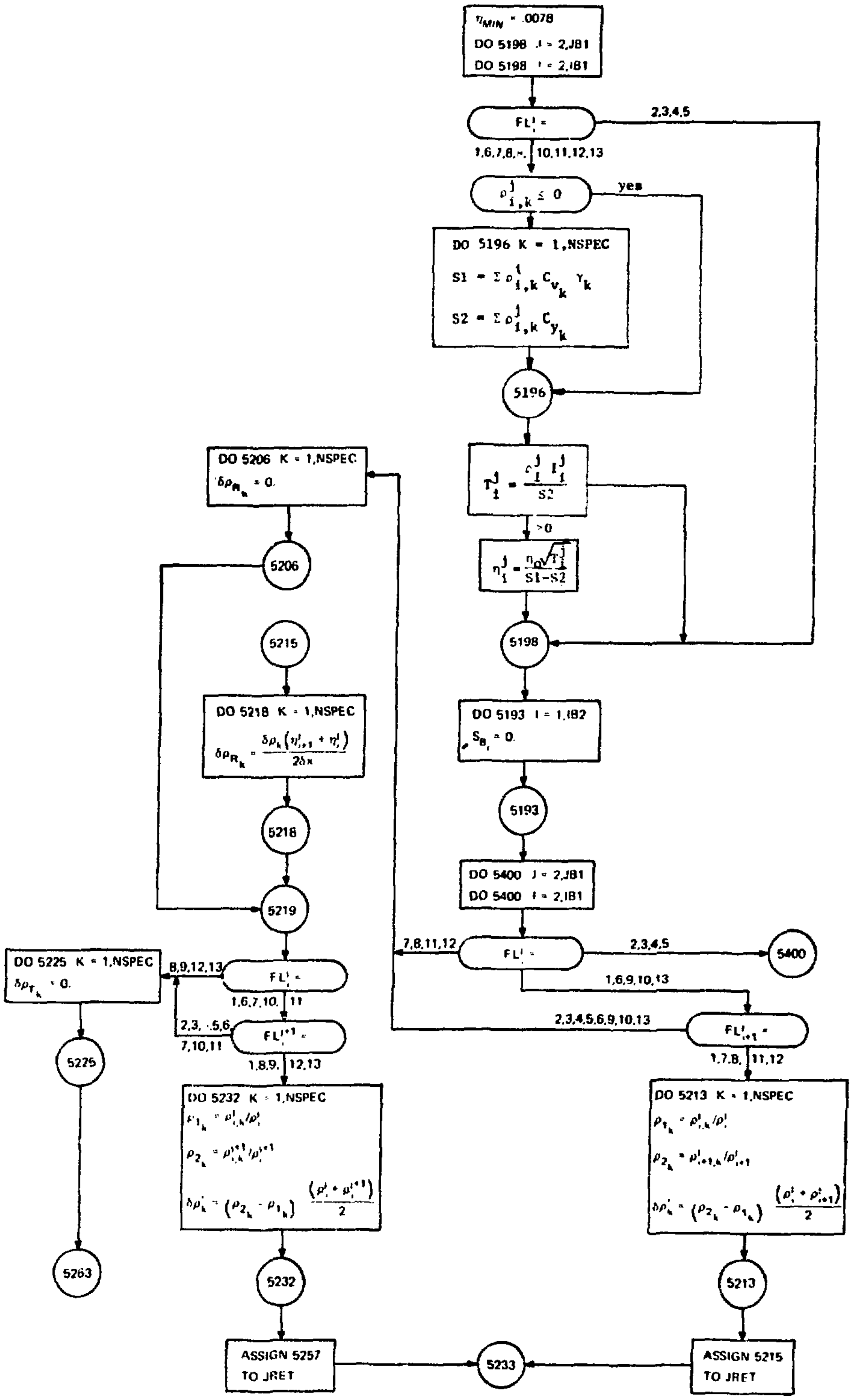
SUBROUTINE SPEC
(Species Convection Calculation)

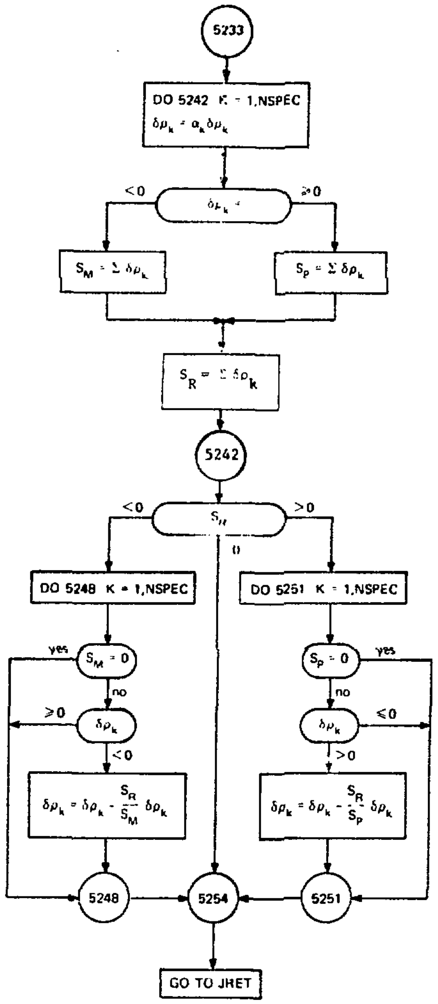


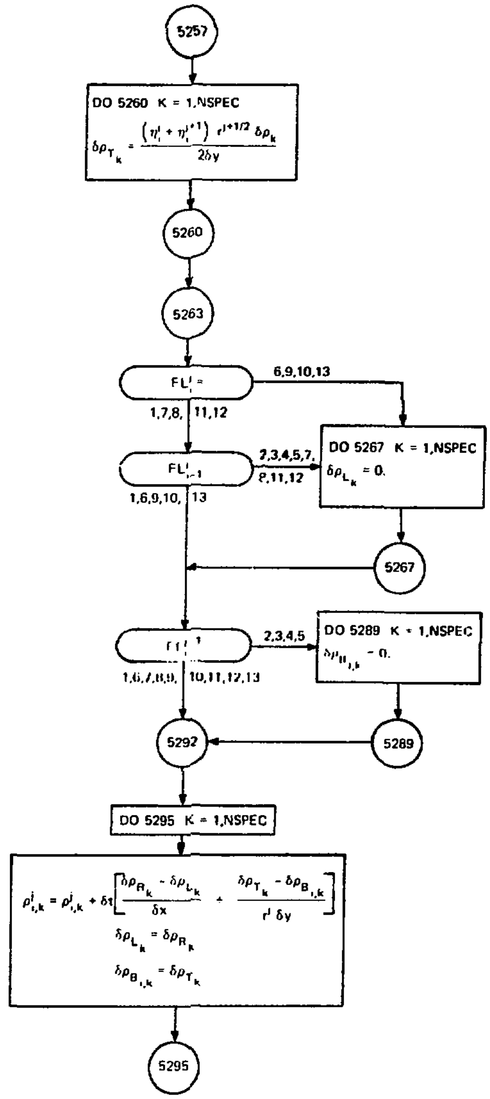

5065
FL =
8,9,12,13
5085
DO 5086 K = 1,NSP
5086
1
6, 7
10,11
2,3,4,6,7,10,11
5
5125
1
8, 9
12,13
yes
5085
no
≥0
<0
5120
DO 5121 K = 1,NSP
5121
DO 5136 K = 1,NSP
5140
5136


5140
FL_i^j =
6,9,10,13
1,7,8, 11,12
1,6,9,10,13
FL_{i-1}^j =
2,3,4,7,8,11,12
DO 5146 K = 1,NSP
ρ_{L_k} = 0.
5
U_{i-1/2}^j > 0
yes
5146
no
ρ_{L_k} already stored from right side of cell on the left
τ_TE = (U_i^j - U_{i-1}^j) δx / 8
≥0
<0
τ_L = (1 + ξ) τ_TE
τ_L = (1 - ξ) τ_TE
DO 5166 K = 1,NSP
5166
5170
FL_i^j =
6,7,10,11
1,8,9, 12,13
FL_i^{j-1}
2,3,4,5,8,9,12,13
DO 5176 K = 1,NSP
ρ_{B_{i,k}} = 0.
1,6,7, 10,11
DO 5186 K = 1,NSP
ρ_{L_k} = ρ_{R_k}
ρ_{B_{i,k}} = ρ_{T_k}
SUMR = Σ ρ_{i,k}^j
5176
5186
ρ_{i,NSPEC}^j = ρ_i^j - SUMR
5200

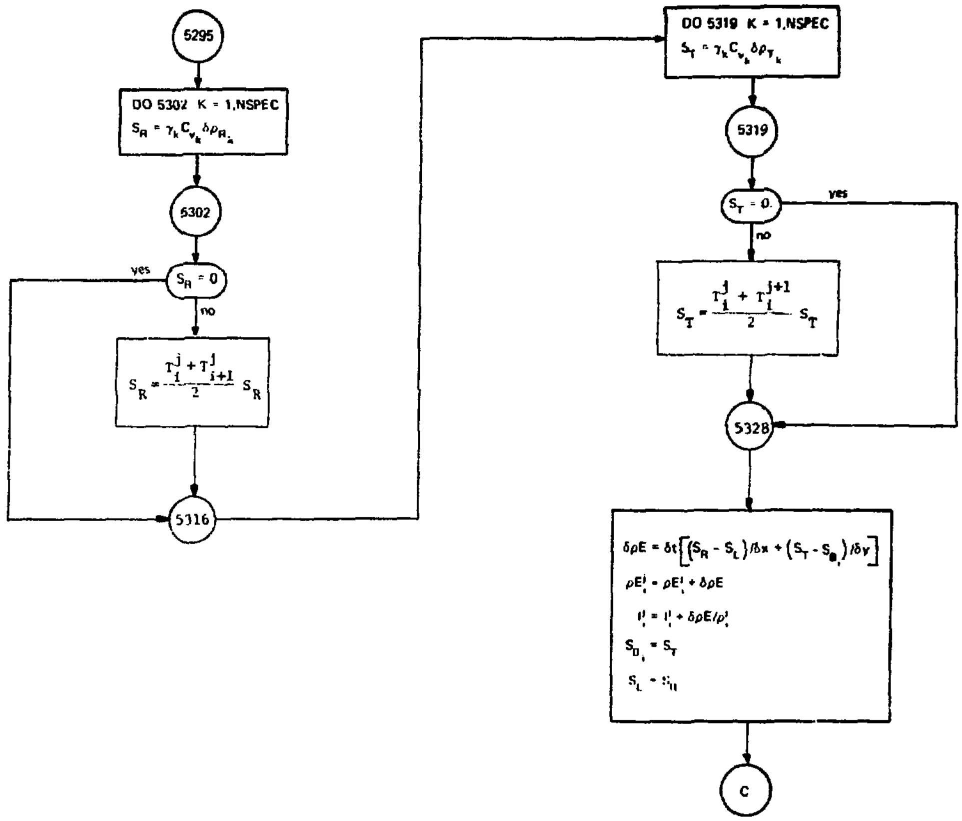
SPEC Continue (3)
(Species Diffusion Calculation)

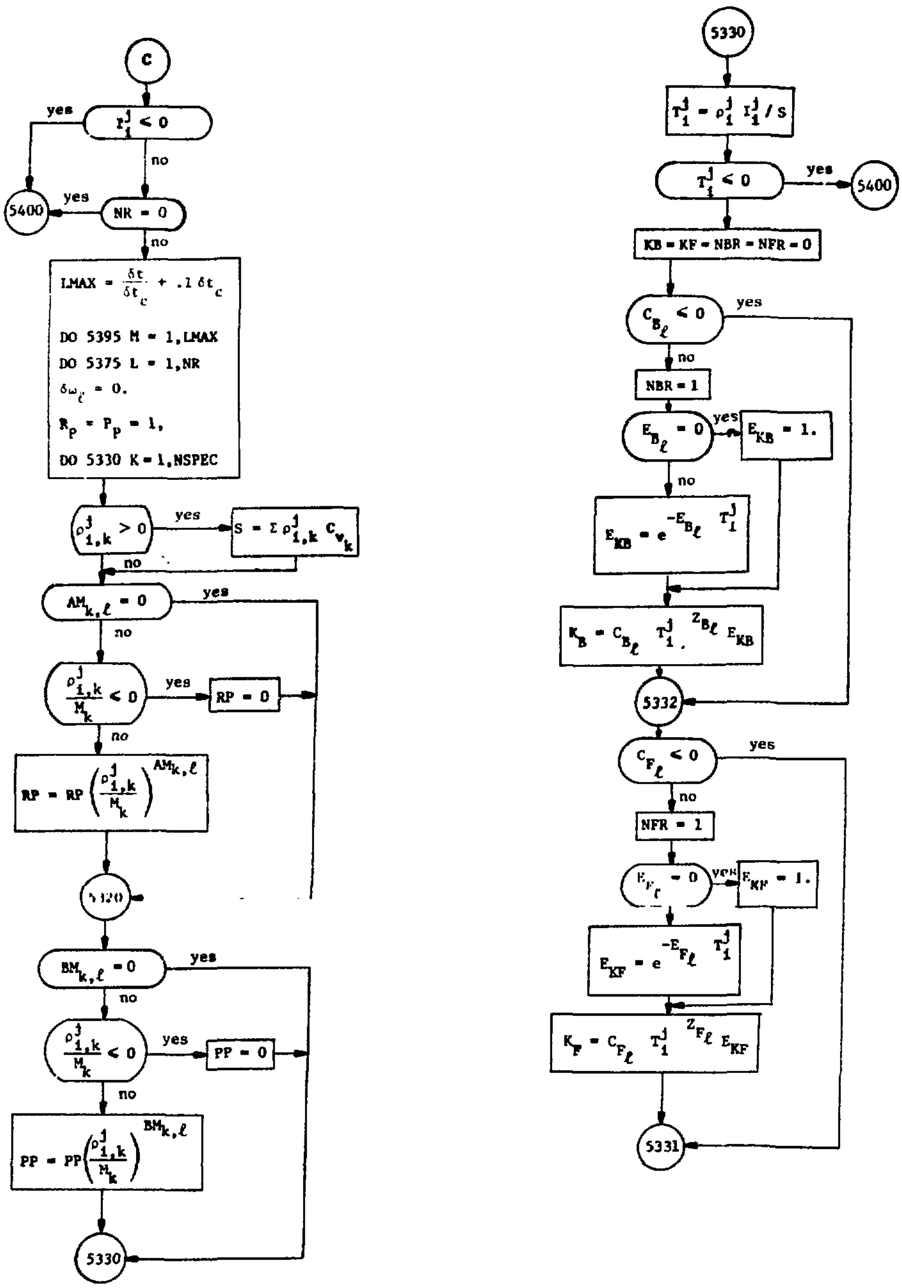


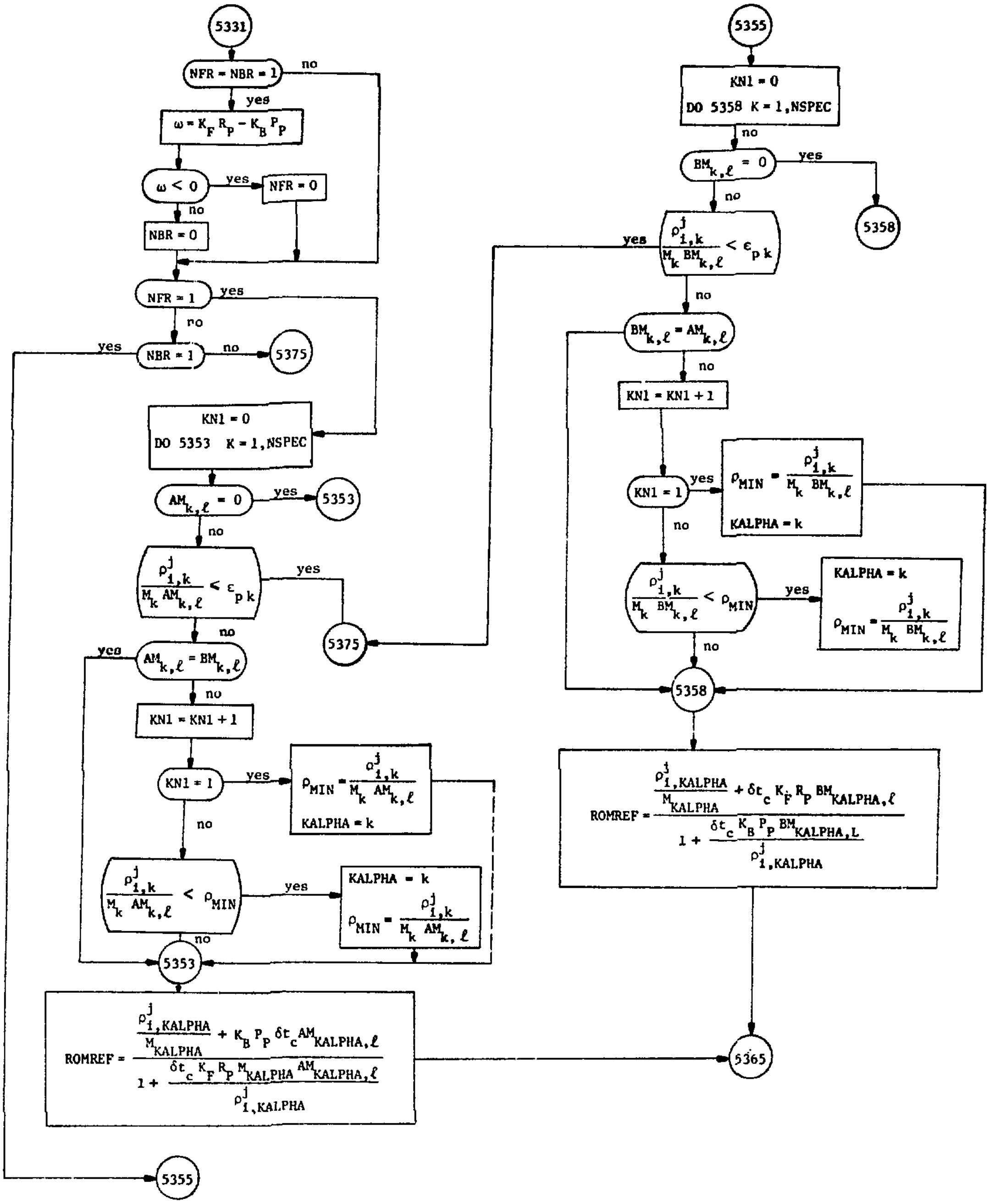

5233
DO 5242 K = 1,NSPEC
<0
≥0
$S_M = \Sigma\,\delta\rho_k$
$S_P = \Sigma\,\delta\rho_k$
$S_R = \Sigma\,\delta\rho_k$
5242
<0
>0
0
DO 5248 K = 1,NSPEC
DO 5251 K = 1,NSPEC
yes
$S_M = 0$
no
≥0
$\delta\rho_k$
<0
$\delta\rho_k = \delta\rho_k - \frac{S_R}{S_M}\,\delta\rho_k$
yes
$S_P = 0$
no
≤0
>0
$\delta\rho_k = \delta\rho_k - \frac{S_R}{S_P}\,\delta\rho_k$
5248
5254
5251
GO TO JRET

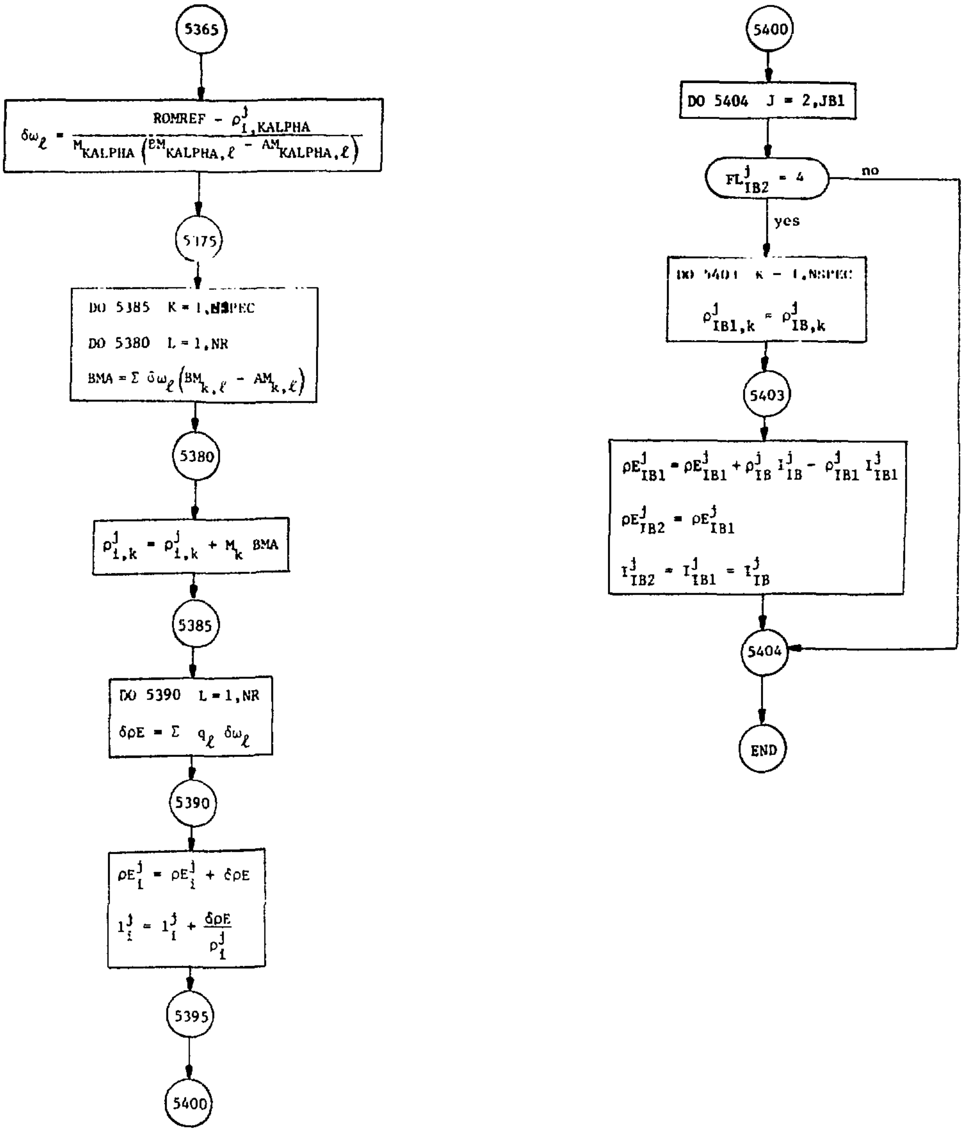

5257
DO 5260 K = 1,NSPEC
$\delta\rho_{T_k} = \frac{(\eta_i^j + \eta_i^{j+1})\, r^{j+1/2}\,\delta\rho_k}{2\delta y}$
5260
5263
6,9,10,13
1,7,8, 11,12
2,3,4,5,7, 8,11,12
DO 5267 K = 1,NSPEC
$\delta\rho_{L_k} = 0.$
1,6,9,10, 13
5267
2,3,4,5
DO 5289 K = 1,NSPEC
$\delta\rho_{B_{i,k}} = 0.$
1,6,7,8,9, 10,11,12,13
5292
5289
DO 5295 K = 1,NSPEC
$\rho_{i,k}^j = \rho_{i,k}^j + \delta t\left[\frac{\delta\rho_{R_k} - \delta\rho_{L_k}}{\delta x} + \frac{\delta\rho_{T_k} - \delta\rho_{B_{i,k}}}{r^j\,\delta y}\right]$
$\delta\rho_{L_k} = \delta\rho_{R_k}$
$\delta\rho_{B_{i,k}} = \delta\rho_{T_k}$
5295

SPEC Continue (5)
(ENTHALPY DIFFUSION)




C
$I_i^j < 0$
yes
no
5400
yes
NR = 0
no
LMAX = $\frac{\delta t}{\delta t_c}$ + .1 δt_c
DO 5395 M = 1,LMAX
DO 5375 L = 1,NR
$\delta\omega_\ell$ = 0.
R_p = P_p = 1.
DO 5330 K = 1,NSPEC
$\rho_{i,k}^j > 0$
yes
S = $\Sigma \rho_{i,k}^j C_{v_k}$
no
$AM_{k,\ell} = 0$
yes
no
$\frac{\rho_{i,k}^j}{M_k} < 0$
yes
RP = 0
no
$RP = RP\left(\frac{\rho_{i,k}^j}{M_k}\right)^{AM_{k,\ell}}$
5320
$BM_{k,\ell} = 0$
yes
no
$\frac{\rho_{i,k}^j}{M_k} < 0$
yes
PP = 0
no
$PP = PP\left(\frac{\rho_{i,k}^j}{M_k}\right)^{BM_{k,\ell}}$
5330
5330
$T_i^j = \rho_i^j I_i^j / S$
$T_i^j < 0$
yes
5400
KB = KF = NBR = NFR = 0
$C_{B_\ell} < 0$
yes
no
NBR = 1
$E_{B_\ell} = 0$
yes
E_{KB} = 1.
no
$E_{KB} = e^{-E_{B_\ell} T_i^j}$
$K_B = C_{B_\ell} T_i^{j\,Z_{B_\ell}} E_{KB}$
5332
$C_{F_\ell} < 0$
yes
no
NFR = 1
$E_{F_\ell} = 0$
yes
E_{KF} = 1.
$E_{KF} = e^{-E_{F_\ell} T_i^j}$
$K_F = C_{F_\ell} T_i^{j\,Z_{F_\ell}} E_{KF}$
5331


5331
NFR = NBR = 1
no
yes
ω = K_F R_P − K_B P_P
ω < 0
yes
NFR = 0
no
NBR = 0
NFR = 1
yes
no
NBR = 1
yes
no
5375
KN1 = 0
DO 5353 K = 1,NSPEC
AM_k,ℓ = 0
yes
5353
no
ρ^j_i,k / (M_k AM_k,ℓ) < ε_p k
yes
5375
no
AM_k,ℓ = BM_k,ℓ
yes
no
KN1 = KN1 + 1
KN1 = 1
yes
ρ_MIN = ρ^j_i,k / (M_k AM_k,ℓ)
KALPHA = k
no
ρ^j_i,k / (M_k AM_k,ℓ) < ρ_MIN
yes
KALPHA = k
ρ_MIN = ρ^j_i,k / (M_k AM_k,ℓ)
no
5353
ROMREF = [ρ^j_i,KALPHA / M_KALPHA + K_B P_P δt_c AM_KALPHA,ℓ] / [1 + δt_c K_F R_P M_KALPHA AM_KALPHA,ℓ / ρ^j_i,KALPHA]
5355
5355
KN1 = 0
DO 5358 K = 1,NSPEC
no
BM_k,ℓ = 0
yes
5358
no
ρ^j_i,k / (M_k BM_k,ℓ) < ε_p k
yes
no
BM_k,ℓ = AM_k,ℓ
no
KN1 = KN1 + 1
KN1 = 1
yes
ρ_MIN = ρ^j_i,k / (M_k BM_k,ℓ)
KALPHA = k
no
ρ^j_i,k / (M_k BM_k,ℓ) < ρ_MIN
yes
KALPHA = k
ρ_MIN = ρ^j_i,k / (M_k BM_k,ℓ)
no
5358
ROMREF = [ρ^j_i,KALPHA / M_KALPHA + δt_c K_F R_P BM_KALPHA,ℓ] / [1 + δt_c K_B P_P BM_KALPHA,L / ρ^j_i,KALPHA]
5365


5365
$\delta\omega_\ell = \frac{ROMREF - \rho^j_{I,KALPHA}}{M_{KALPHA}\left(BM_{KALPHA,\ell} - AM_{KALPHA,\ell}\right)}$
5375
DO 5385 K = 1,NSPEC
DO 5380 L = 1,NR
$BMA = \Sigma\ \delta\omega_\ell\left(BM_{k,\ell} - AM_{k,\ell}\right)$
5380
$\rho^j_{I,k} = \rho^j_{I,k} + M_k\ BMA$
5385
DO 5390 L = 1,NR
$\delta\rho E = \Sigma\ q_\ell\ \delta\omega_\ell$
5390
$\rho E^j_I = \rho E^j_I + \delta\rho E$
$I^j_I = I^j_I + \frac{\delta\rho E}{\rho^j_I}$
5395
5400
5400
DO 5404 J = 2,JB1
$FL^j_{IB2} = 4$
no
yes
DO 5401 K = 1,NSPEC
$\rho^j_{IB1,k} = \rho^j_{IB,k}$
5403
$\rho E^j_{IB1} = \rho E^j_{IB1} + \rho^j_{IB}\ I^j_{IB} - \rho^j_{IB1}\ I^j_{IB1}$
$\rho E^j_{IB2} = \rho E^j_{IB1}$
$I^j_{IB2} = I^j_{IB1} = I^j_{IB}$
5404
END

APPENDIX C

FØRTRAN IV LISTING OF THE RICE PROGRAM

The FØRTRAN IV listing of the main program and of each subroutine in the RICE program is given in this appendix.

LP-0396

```
RUN-LCM89                               74/08/29     09.40.08     T3OAFL 1AZ    PAGE NO. 1

            PROGRAM PREP(INP,FSET10=INP,FSET6,OUT,FSET9=OUT)
         44 FORMAT(*           LCM/CELLD/FL(*I3*,*I3*),U(*I3*,*I3*',V(*I3*,*I3/
           1*       1),RHO(*I3*,*I3*),SIE(*I3*,*I3*),P(*I3*,*I3/
           2*       2),MU(*I3*,*I3*),CSQ(*I3*,*I3*),RHOT(*I3*,*I3*),RHOU(*I3/
           3*       3,*I3*),RHOV(*I3*,*I3*),RE(*I3*,*I3*),ETA(*I3*,*I3*),CQ(*/
           4*       4*I3*,*I3*),FA(*I3*,*I3*,*I2*),RJ(*I3*),RJ2(*/
           5*       5*I3*),RFLB(*I3*),TFAB(*I3*,*I2*),TE(*I3*,*I3*)*)
         45 FORMAT(3I6)
         46 FORMAT(*        DIMENSION RHOUT(*I3*,*I3*),RHOVT(*I3*,*I3*),PB(*/
           1*       1*I3*,*I3*),DRHOB(*I3*,*I2*),STREB(*I3*)*)
   2        READ 45,IB2,JB2,NSFEC
  14        WRITE(6,44),IB2,JB2,IB2,JB2,IB2,JB2,IB2,JB2,
           1IB2,JB2,IB2,JB2,IB2,JB2,IB2,JB2,IB2,JB2,IB2,JB2,
           2IB2,JB2,IB2,JB2,IB2,JB2,IB2,JB2,IB2,JB2,NSFEC,JB2,JB2,IB2,
           3IB2,NSFEC,IB2,JB2
 134        WRITE(6,46)IB2,JB2,IB2,JB2,IB2,JB2,IB2,NSFEC,IB2
 162        END
```

```
        PROGRAM RICE(INP,OUT,FILM,FSET12=FILM,FSET5,FSET9=OUT,FSET10=INP)  RICE6      3
      C**** TAPE-5 DATA FOR STORAGE AND RESTART                             RCOM1      2
  2     COMMON/TAPED/ITD,NTD,NSDMP,NFILE,NWDMP,NCYDMP                       RCOM1      3
      C**** THE RCOM1 DATA BLOCK IS USED IN RICE(MAIN-PROGRAM),PROG AND RTAPE5 RCOM1   4
      C                                                                     RCOM1      5
      C**** THE RCOM2 DATA BLOCK IS COMMON TO RICE AND ALL SUBROUTINES      RCOM1      6
      C**** BOTH RCOM1 AND RCOM2 ARE LOADED USING *COMDECK FEATURE OF (UPDATE) RCOM1   7
  2     COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,        RCOM2      2
       1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC,ALA,BIE,THETA,     RCOM2      3
       2PHI,OMEGA,GAMMA,DXOU,ASQ                                            RCOM2      4
  2     COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SECREQ,DDX,DDY          RCOM2      5
       1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT                               RCOM2      6
  2     COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1,     RCOM2      7
       1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,     RCOM2      8
       2SIEL1,SIEL2                                                         RCOM2      9
  2     REAL MUO,MINT1,MINT2,MINL1,MINL2                                    RCOM2     10
  2     COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                           RCOM2     11
       1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),         RCOM2     12
       2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)              RCOM2     13
  2     COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20)  RCOM2     14
       1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),                 RCOM2     15
       2SIEBO(20),FARO(7,20),FABO(7,20)                                     RCOM2     16
  2     INTEGER AM,BM                                                       RCOM2     17
  2     REAL MINRO,MINBO                                                    RCOM2     18
      C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP) RCOM2  19
      C**** AND THE *READ FEATURE OF (UPDATE)                               RCOM2     20
      C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/      RCOM2     21
      C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC RCOM2    22
      C                                                                     RCOM2     23
  2          LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32                     RCOM2     24
       1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                               RCOM2     25
       2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62                   RCOM2     26
       3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(                    RCOM2     27
       4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                                RCOM2     28
       5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                             RCOM2     29
  2     DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                         RCOM2     30
       1 62, 32),DRHOB( 62, 7),STREB( 62)                                   RCOM2     31
  2     EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),          RCOM2     32
       1(TFAB,DRHOB)                                                        RCOM2     33
  2     INTEGER FL,CYCLE                                                    RCOM2     34
  2     REAL MU                                                             RCOM2     35
      1 FORMAT(24I3)                                                        RICE8      6
      2 FORMAT(2I3,11F6.2)                                                  RICE8      7
      3 FORMAT(1H036I3)                                                     RICE8      8
      4 FORMAT(12F6.2)                                                      RICE8      9
      5 FORMAT(10A8)                                                        RICE8     10
      6 FORMAT(*0UO=*1PE13.4*   VO=*1PE13.4*   VINT1=*1PE13.4*   VINT2=*1P RICE8     11
       1E13.4*   UINL1=*1PE13.4*   UINL2=*1PE13.4)                          RICE8     12
      7 FORMAT(1H1)                                                         RICE8     13
      8 FORMAT(1H 2I3,1P8E13.4)                                             RICE8     14
      9 FORMAT(*1CASE NO. *I3*   CP TIME=*F8.1)                             RICE8     15
     10 FORMAT(1H 1P12E11.3)                                                RICE8     16
     11 FORMAT(8I3,I6,15I3)                                                 RICE8     17
     12 FORMAT(14I2,2X,7F6.2)                                               RICE8     18
     13 FORMAT(*0   UINRO       RINRO       MINRO       SIERO*)             RICE8     19
```

```
   14 FORMAT(*0   VINBO      RINBO      MINBO      SIEBO*)               RICE8
   15 FORMAT(*0   FA(1)R     FA(2)R     FA(3)R     FA(4)R     FA(5)R    RICE8
     1 FA(6)R     FA(7)R*)                                             RICE8
   16 FORMAT(*0ITD=*I3*  NTD=*I3*  NSDMP=*I5*  NFILE=*I3*  NWDMP=*I5)  RICE8
   17 FORMAT(*0             RHOO=*1PE13.4*       RINT1=*1PE13.4*   RINT2 RICE8
     1=*1PE13.4*   RINL1=*1PE13.4*   RINL2=*1PE13.4)                   RICE8
   19 FORMAT(*0              MUO=*1PE13.4*       MINT1=*1PE13.4*   MINT2 RICE8
     1=*1PE13.4*   MINL1=*1PE13.4*   MINL2=*1PE13.4)                   RICE8
   20 FORMAT(*0IB2=*I3*  JB2=*I3*   NO=*I3*  NSL=*4I3*  LPR=*I3        RICE8
     1*  CYCLE=*I5*  ITC=*I2*  JPLOT=*5I2* NSPEC=*I2*  ISPEC=*7I2*  NR=* RICE8
     2I2)                                                              RICE8
   21 FORMAT(*0INFLOW DATA      --RIGHT--                    ----------- RICE8
     1-----TOP----------------------                --LEFT--*)        RICE8
   22 FORMAT(*0DX=*1PE10.3* DY=*1PE10.3* T=*1PE10.3* DT=*1PE10.3* TSTOP= RICE8
     1*1PE10.3* TPR=*1PE10.3* TPL=*1PE10.3* TPLD=*1PE10.3* DTC=*1PE10.3) RICE8
   23 FORMAT(*0                                      -- OBSTACLE DATA --*/) RICE8
   24 FORMAT(*0ITO NSO      OB(1)      OB(2)      OB(3)      OB(4)      RICE8
     1OB(5)      OB(6)      OIB(1)      OIB(2)*)                       RICE8
   25 FORMAT(*0             SIEO=*1PE13.4*       SIET1=*1PE13.4*   SIET2 RICE8
     1=*1PE13.4*   SIEL1=*1PE13.4*   SIEL2=*1PE13.4)                   RICE8
   26 FORMAT(*0ALA=*1PE10.3*  BIE=*1PE10.3*  THETA=*1PE10.3*  PHI=*     RICE8
     11PE10.3*  OMEGA=*1PE10.3*  GAMMA=*1PE10.3*  DXOU=*1PE10.3*  ASQ=*  RICE8
     21PE10.3 //)                                                      RICE8
   27 FORMAT(*0             ETAO=*1PE13.4)                             RICE8
   28 FORMAT(*0   FA(1)B     FA(2)B     FA(3)B     FA(4)B     FA(5)B    RICE8
     1 FA(6)B     FA(7)B*)                                             RICE8
   29 FORMAT(I4  *              FAO=*1PE13.4*     FAT1=*1PE13.4*    FAT2=*1P RICE8
     1E13.4*   FAL1=*1PE13.4*    FAL2=*1PE13.4)                        RICE8
   30 FORMAT(*0SPEC.NO*)                                               RICE8
   32 FORMAT(*0                                      -- SPECIES DATA --*/ ) RICE8
   33 FORMAT(*0MOLE WT(SM)=*1P10E13.4)                                 RICE8
   34 FORMAT(* EPSILON(EP)=*1P10E13.4)                                 RICE8
   35 FORMAT(*      (GAMA)=*1P10E13.4)                                 RICE8
   36 FORMAT(* SP.HEAT(CV)=*1P10E13.4)                                 RICE8
   37 FORMAT(* DIF.(ALPHA)=*1P10E13.4)                                 RICE8
   38 FORMAT(*1                        --SPECIES REACTION DATA--*)     RICE8
   39 FORMAT(*0             AM(7)                      BM(7)         Q RICE8
     1        CF        ZETAF          EF          CB        ZETAB     RICE8
     2   EB*)                                                          RICE8
   40 FORMAT(7I3,6X,7I3,1P7E12.4)                                      RICE8
   41 FORMAT(*0                                 --SCALING QUANTITIES--*) RICE8
   42 FORMAT(*0LENGTH (CM)=*1PE11.4*    VELOCITY (CM/SEC)=*1PE11.4*    D RICE8
     1ENSITY (GM/CC)=*1PE11.4*    MOLECULAR WT.(GM/MOLE)=*1PE11.4)     RICE8
   43 FORMAT(*0TEMPERATURE (DEG.K)=*1PE11.4*     TIME (SEC)=*1PE11.4*   RICE8
     1CONCENTRATION (MOLES/CC)=*1PE11.4  //)                           RICE8
C**** ITD=0--DOES NOT READ OR WRITE TAPE (5)                           RICE8
C**** ITD=1--DOES NOT READ TAPE (5) BUT WRITES TAPE(5)                 RICE8
C**** ITD=2--READS TAPE,DOES NOT WRITE TAPE                            RICE8
C**** ITD=3--READS AND WRITES TAPE (5)                                 RICE8
C**** NTD= NO. ON TAPE -5- TO READ DATA AT---------                    RICE8
C**** NSDMP= FREQUENCY AT WHICH DATA IS WRITTEN ON TAPE-5- (CYCLE-INTERV RICE8
C**** NFILE = NO. ON TAPE5 OF LAST DUMP TAKEN FROM A PREVIOUS RUN      RICE8
C                                                                      RICE8
3           CALL SECOND(TBEG)                                          RICE8
4           CALL GETQ(4LKTLM,NCREQ)                                    RICE8
```

```
  6          SECREQ=NCREQ*27.5E-9                                            RICE8
       C                                                                     RICE8
       C**** START OF -READ- INPUT DATA                                      RICE8
       C                                                                     RICE8
 11       51 READ 5,NAME                                                     RICE8
 17          READ 4,SLG,VEL,DEN,SMO,TEM                                      RICE8
 35          READ 1,ITD,NTD,NSDMP,NFILE,NWDMP                                RICE8
 53          READ 11,IB2,JB2,NO,(NSL(M),M=1,4),LPR,CYCLE,ITC,(JPLOT(M),M=1,5)  RICE8
            1,NSPEC,(ISPEC(K),K=1,7),NR                                      RICE8
105          NCYDMP=NSDMP $ NCASE=0                                          RICE8
       C                                                                     RICE8
       C**** OBTAINS FLUID VARIABLES FROM TAPE5 IF(ITD.GE.2)                 RICE8
107          IF(ITD.GE.2)CALL RTAPE5                                         RICE8
113          CALL ADV(3)                                                     RICE8
115          CALL LINCNT(64)                                                 RICE8
       C                                                                     RICE8
       C**** SPACES FILM 3-FRAMES,SETS FILM LINE COUNT AND PUTS JOB ID ON FILM  RICE8
117          CALL GETQ(4LKJBN,JNM)                                           RICE8
121          READ 4,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC                     RICE8
147          READ 4,ALA,BIE,THETA,PHI,OMEGA,GAMMA,DXOU,ASQ                   RICE8
173          READ4,(FLO(M),M=1,12)                                           RICE8
201          READ 4,UO,VO,VINT1,VINT2,UINL1,UINL2                            RICE8
221          READ 4,RHOO,RINT1,RINT2,RINL1,RINL2                             RICE8
237          READ        4,MUO, MINT1,MINT2,MINL1,MINL2                      RICE8
255          READ        4,SIEO,SIET1,SIET2,SIEL1,SIEL2                      RICE8
273          IF(NSPEC.LT.1)GO TO 46                                          RICE8
       C                                                                     RICE8
       C**** READS SPECIES DATA IF(NSPEC.GE.1),BYPASSES IF(NSPEC.LE.0)       RICE8
       C                                                                     RICE8
276          READ 4,ETAO                                                     RICE8
303          DO 52 K=1,7                                                     RICE8
305          READ 4,FAO(K),FAT1(K),FAT2(K),FAL1(K),FAL2(K)                   RICE8
327       52 CONTINUE                                                        RICE8
331          READ 4,(    SM(K),K=1,7)                                        RICE8
337          READ 4,(    EP(K),K=1,7)                                        RICE8
345          READ 4,( GAMA(K),K=1,7)                                         RICE8
353          READ 4,(    CV(K),K=1,7)                                        RICE8
361          READ 4,(ALPHA(K),K=1,7)                                         RICE8
367          IF(NR.LT.1)GO TO 46                                             RICE8
       C**** READ ONE DATA CARD FOR EACH (NR) REACTION * * * * * * * * * * * * *  RICE8
372          DO 49 L=1,NR                                                    RICE8
373          READ 12,(AM(K,L),K=1,7),(BM(K,L),K=1,7),Q(L),CF(L),ZETAF(L),EF(L),  RICE8
            1CB(L),ZETAB(L),EB(L)                                            RICE8
433       49 CONTINUE                                                        RICE8
436       46 CONTINUE                                                        RICE8
436          IF(NO.EQ.0)GO TO 53                                             RICE8
       C                                                                     RICE8
       C**** READS A SET OF FIVE CARDS FOR EACH OBSTACLE * * * * * * * * * * *  RICE8
       C                                                                     RICE8
437             DO 55 N=1,NO                                                 RICE8
441             READ 2,ITO(N),NSO(N),(OB(M,N),M=1,6),(OIB(M,N),M=1,2)        RICE8
462             READ 4,UINRO(N),RINRO(N),MINRO(N),SIERO(N)                   RICE8
502             READ 4,VINBO(N),RINBO(N),MINBO(N),SIEBO(N)                   RICE8
522             READ 4,(FARO(K,N),K=1,7)                                     RICE8
532             READ 4,(FARO(K,N),K=1,7)                                     RICE8
```

```
 542      55 CONTINUE                                                          RICE8
        C                                                                      RICE8
        C**** END OF READ INPUT DATA------PRINT ALL INPUT DATA READ FROM CARDS  RICE8
 545      53 KTAPE=9                                                           RICE8
        C**** PRINTS ALL INPUT DATA (TAPE-9),THEN COPIES ON FILM (TAPE-12) * * * RICE8
 546      54 WRITE(KTAPE,5)NAME                                                RICE8
 554         TI=SLG/VEL                                                        RICE8
 556         CON=DEN/SMO                                                       RICE8
 560         WRITE(KTAPE,41)                                                   RICE8
 564         WRITE(KTAPE,42)SLG,VEL,DEN,SMO                                    RICE8
 600         WRITE(KTAPE,43)TEM,TI,CON                                         RICE8
 612         WRITE(KTAPE,16)ITD,NTD,NSDMP,NFILE,NWDMP                          RICE8
 630         WRITE(KTAPE,20)IB2,JB2,NO,(NSL(K),K=1,4),LPR,CYCLE,ITC            RICE8
            1,(JPLOT(K),K=1,5),NSPEC,(ISPEC(K),K=1,7),NR                       RICE8
 662         WRITE(KTAPE,22)DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC               RICE8
 710         WRITE(KTAPE,26) ALA,BIE,THETA,PHI,OMEGA,GAMMA,DXOU,ASQ            RICE8
 734         WRITE(KTAPE,21)                                                   RICE8
 740         WRITE(KTAPE,10)(FLO(K),K=1,12)                                    RICE8
 746         WRITE(KTAPE, 6)UO,VO,VINT1,VINT2,UINL1,UINL2                      RICE8
 766         WRITE(KTAPE,17)RHOO,RINT1,RINT2,RINL1,RINL2                       RICE8
1004         WRITE(KTAPE,19) MUO,MINT1,MINT2,MINL1,MINL2                       RICE8
1022         WRITE(KTAPE,25)SIEO,SIET1,SIET2,SIEL1,SIEL2                       RICE8
1040         WRITE(KTAPE,32)                                                   RICE8
1044         IF(NSPEC.LT.1)GO TO 59                                            RICE8
        C                                                                      RICE8
        C**** PRINTS SPECIES DATA ONLY WHEN NSPEC IS GREATER THAN ZERO         RICE8
1047         WRITE(KTAPE,27)ETAO                                               RICE8
1054         WRITE(KTAPE,30)                                                   RICE8
1060         DO 58 K=1,7                                                       RICE8
1062         WRITE(KTAPE,29)K,FAO(K),FAT1(K),FAT2(K),FAL1(K),FAL2(K)           RICE8
1106      58 CONTINUE                                                          RICE8
1110         WRITE(KTAPE,33)(SM(K),K=1,7)                                      RICE8
1116         WRITE(KTAPE,34)(EP(K),K=1,7)                                      RICE8
1124         WRITE(KTAPE,35)(GAMA(K),K=1,7)                                    RICE8
1132         WRITE(KTAPE,36)(CV(K),K=1,7)                                      RICE8
1140         WRITE(KTAPE,37)(ALPHA(K),K=1,7)                                   RICE8
1146         IF(NR.LT.1)GO TO 59                                               RICE8
1151         WRITE(KTAPE,38)                                                   RICE8
1154         WRITE(KTAPE,39)                                                   RICE8
1160         DO 56 L=1,NR                                                      RICE8
1162         WRITE(KTAPE,40)(AM(K,L),K=1,7),(BM(K,L),K=1,7),Q(L),CF(L),ZETAF(L) RICE8
            1,EF(L),CB(L),ZETAB(L),EB(L)                                       RICE8
1222      56 CONTINUE                                                          RICE8
1225      59 CONTINUE                                                          RICE8
        C                                                                      RICE8
        C**** PRINTS OBSTACLE DATA ONLY IF COMPUTING REGION CONTAINS OBSTACLES RICE8
1225         IF(KTAPE.EQ.9)NSDMP=NSDMP+CYCLE                                   RICE8
1231         WRITE(KTAPE,23)                                                   RICE8
1235         IF(NO.EQ.0) GO TO 60                                              RICE8
1236         DO 57 N=1,NO                                                      RICE8
1240         WRITE(KTAPE,24)                                                   RICE8
1243         WRITE(KTAPE,8)ITO(N),NSO(N),(OB(M,N),M=1,6),(OIB(M,N),M=1,2)      RICE8
1265         WRITE(KTAPE,13)                                                   RICE8
1271         WRITE(KTAPE,10)UINRO(N),RINRO(N),MINRO(N),SIERO(N)                RICE8
1311         WRITE(KTAPE,14)                                                   RICE8
```

```
1315          WRITE(KTAPE,10)VINBO(N),RINBO(N),MINBO(N),SIEBO(N)          RICE8
1335          WRITE(KTAPE,15)                                             RICE8
1341          WRITE(KTAPE,10)(FARO(K,N),K=1,7)                            RICE8
1351          WRITE(KTAPE,28)                                             RICE8
1355          WRITE(KTAPE,10)(FABO(K,N),K=1,7)                            RICE8
1365       57 CONTINUE                                                    RICE8
1370       60 CONTINUE                                                    RICE8
1370          IF(KTAPE.NE.9)GO TO 62                                      RICE8
1372          CALL ADV(1)                                                 RICE8
1374          KTAPE=12 $ GO TO 54                                         RICE8
1376       62 CONTINUE                                                    RICE8
1376          IB=IB2-2                                                    RICE8
1400          IB1=IB2-1                                                   RICE8
1401          JB=JB2-2                                                    RICE8
1403          JB1=JB2-1                                                   RICE8
      C**** CONVERTS ALL GRID DATA TO 4020 NUMBERS IN ALL PLOT ROUTINES  . . . RICE8
1404          CALL VPLOT(0)                                               RICE8
1405          CALL CNPLOT(0)                                              RICE8
1407          CALL COPLOT(0)                                              RICE8
      C**** TRANSFERS TO CONTROL PROGRAM FOR TIME DEPENDENT CALC.  . . . . . . . RICE8
1411          CALL PROG                                                   RICE8
1412          CALL SECOND(TEND)                                           RICE8
1414          NCASE=NCASE+1 $ TCASE=TEND-TBEG                             RICE8
1420          WRITE(9,9)NCASE,TCASE                                       RICE8
1427          GO TO 51 $ END                                              RICE8
      C                                                                   RICE8
      C                                                                   RICE8
      C**** END OF THE MAIN PROGRAM ****************************************** RICE8
      C                                                                   RICE8
      C                                                                   RICE8
```

```
          SUBROUTINE PROG                                                   RICE8     215
    C**** TAPE-5 DATA FOR STORAGE AND RESTART                               RCOM1       2
 1        COMMON/TAPED/ITD,NTD,NSDMP,NFILE,NWDMP,NCYDMP                     RCOM1       3
    C**** THE RCOM1 DATA BLOCK IS USED IN RICE(MAIN-PROGRAM),PROG AND RTAPE5 RCOM1      4
    C                                                                       RCOM1       5
    C**** THE RCOM2 DATA BLOCK IS COMMON TO RICE AND ALL SUBROUTINES        RCOM1       6
    C**** BOTH RCOM1 AND RCOM2 ARE LOADED USING *COMDECK FEATURE OF (UPDATE) RCOM1      7
 1        COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,      RCOM2       2
         1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC,ALA,BIE,THETA,   RCOM2       3
         2PHI,OMEGA,GAMMA,DXOU,ASQ                                          RCOM2       4
 1        COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SCCREQ,DDX,DDY        RCOM2       5
         1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT                             RCOM2       6
 1        COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1,   RCOM2       7
         1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,   RCOM2       8
         2SIEL1,SIEL2                                                       RCOM2       9
 1        REAL MUO,MINT1,MINT2,MINL1,MINL2                                  RCOM2      10
 1        COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                         RCOM2      11
         1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),       RCOM2      12
         2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)            RCOM2      13
 1        COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20) RCOM2     14
         1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),               RCOM2      15
         2SIEBO(20),FARO(7,20),FABO(7,20)                                   RCOM2      16
 1        INTEGER AM,BM                                                     RCOM2      17
 1        REAL MINRO,MINBO                                                  RCOM2      18
    C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP) RCOM2     19
    C**** AND THE *READ FEATURE OF (UPDATE)                                 RCOM2      20
    C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/        RCOM2      21
    C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC  RCOM2      22
    C                                                                       RCOM2      23
 1          LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32                      RCOM2      24
         1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                             RCOM2      25
         2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62                 RCOM2      26
         3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(                  RCOM2      27
         4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                              RCOM2      28
         5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                           RCOM2      29
 1        DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                       RCOM2      30
         1 62, 32),DRHOB( 62, 7),STREB( 62)                                 RCOM2      31
 1        EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),        RCOM2      32
         1(TFAB,DRHOB)                                                      RCOM2      33
 1        INTEGER FL,CYCLE                                                  RCOM2      34
 1        REAL MU                                                           RCOM2      35
        1 FORMAT(24I3)                                                      RICE8     218
        3 FORMAT(1H036I3)                                                   RICE8     219
        4 FORMAT(1H 1P10E13.5)                                              RICE8     220
        5 FORMAT(1H0,1P8E15.4)                                              RICE8     221
        7 FORMAT(2I5,7(2X,1PE12.5))                                         RICE8     222
        8 FORMAT(*0TAPE 5 DUMP NO.=*I5*   CYCLE=*I5*   TIME=*F10.4)         RICE8     223
        9 FORMAT(*1CYCLE=*I5*  TIME=*F8.3,10A8)                              RICE8     224
       30 FORMAT(*  I  J  FL       U          V          RHO          PB    RICE8     225
         1   SIE         TEMP.          MU*)                                RICE8     226
       31 FORMAT(1H 3I3,1P10E12.4)                                          RICE8     227
       32 FORMAT(*      SPECIES DENSITIES*)                                 RICE8     228
       33 FORMAT(*  I  J  FL       A          B          C           D      RICE8     229
         1      E          F          G*)                                   RICE8     230
       41 FORMAT(* ITER=*I4*    TIME=*F7.2*    CYCLE=*I4*    CP=*1PE10.2    RICE8     231
```

```
          1*     GRINDS=*1PE12.5)                                            RICE6     232
  1        TPR1=TPR+TIME                                                     RICE8     233
  3        TPLO=TPL+TIME                                                     RICE8     234
  4        TPLOD=TPLD+TIME                                                   RICE8     235
  6        TRDISK=2100.                                                      RICE8     236
 10        CALL FLIC                                                         RICE8     237
     C**** FLIC---SETS ALL CELL FLAGS                                        RICE8     238
     C                                                                       RICE8     239
     C**** CVIC---SETS UP INITIAL CONDITIONS FOR FLUID VARIABLES FROM INPUT  RICE8     240
     C**** DATA                                                              RICE8     241
 11        CALL CVIC                                                         RICE8     242
 12        CALL SECOND(TCP)                                                  RICE8     243
     C*****START OF TIME STEP----------END OF I.C.SETUP ******************** RICE8     244
     C                                                                       RICE8     245
     C                                                                       RICE8     246
     C                                                                       RICE8     247
 14   1000 CONTINUE                                                          RICE8     248
 14        JC=0                                                              RICE8     249
 15        DO 1025 J=2,JB1                                                   RICE8     250
 17        JC=JC+IB2                                                         RICE8     251
 20        DO 1025 I=2,IB1                                                   RICE8     252
 22        IJ=I+JC                                                           RICE8     253
 23        IMJ=IJ-1                                                          RICE8     254
 24        IJM=IJ-IB2                                                        RICE8     255
 26        IF(FL(IJ).GT.1.AND.FL(IJ).LT.6)GO TO 1025                         RICE8     256
 40        IF(NSPEC.LE.0)GO TO 1004                                          RICE8     257
 41        SUM=0.                                                            RICE8     258
 42        SUM1=0.                                                           RICE8     259
 43        DO 1003 K=1,NSPEC                                                 RICE8     260
 44        KZ=IB2*JB2*(K-1)                                                  RICE8     261
 47        IJK=IJ+KZ                                                         RICE8     262
 51        IF(FA(IJK  ).LE.0.)GO TO 1003                                     RICE8     263
 55        SUM=SUM+FA(IJK  )*CV(K)                                           RICE8     264
 61        SUM1=SUM1+FA(IJK  )*CV(K)*GAMA(K)                                 RICE8     265
 63   1003 CONTINUE                                                          RICE8     266
 66        IF(SUM.NE.0.)GAMMA=SUM1/SUM                                       RICE8     267
 70        IF(SUM.LE.0.)TE(IJ)=0.                                            RICE8     268
 75        IF(SUM.GT.0.)TE(IJ)=RHO(IJ)*SIE(IJ)/SUM                           RICE8     269
105   1004 CONTINUE                                                          RICE8     270
106        CSQ(IJ)=GAMMA*(GAMMA-1.)*SIE(IJ)+ASQ                              RICE8     271
     C**** CALC.C**2,P AND MU AT END OF TIME STEP ALL DATA TIME UPDATED * * * RICE8    272
     C                                                                       RICE8     273
113        MU(IJ)=MU0*RHO(IJ)*RRHO                                           RICE8     274
120        IF(ASQ.GT.0.)GO TO 1025                                           RICE8     275
122        P(IJ)=(GAMMA-1.)*RHO(IJ)*SIE(IJ)                                  RICE8     276
          1+ASQ*(RHO(IJ)-RHOO)                                               RICE8     277
135        IF(P(IJ ).LT.0.)P(IJ )=0.                                         RICE8     278
141   1025 CONTINUE                                                          RICE8     279
146        DO 1030 J=2,JB1                                                   RICE8     280
150        IF(FL(IB2,J).NE.4)GO TO 1030                                      RICE8     281
155        CSQ(IB2,J)=CSQ(IB1,J)=CSQ(IB,J)                                   RICE8     282
167        MU(IB2,J)=MU(IB1,J)=MU(IB,J)                                      RICE8     283
200        P(IB2,J)=P(IB1,J)=P(IB,J)                                         RICE8     284
212        TE(IB2,J)=TE(IB1,J)=TE(IB,J)                                      RICE8     285
223   1030 CONTINUE                                                          RICE8     286
```

```
      C*****SETS TANGENTIAL VEL.IN FICTIOUS AND RECTANGULAR OBSTACLE CELLS     RICE6
226         CALL TANVEL                                                        RICE8
      C                                                                        RICE8
      C                                                                        RICE8
      C**************************** PRINT SECTION (2000) ******************** RICE8
      C*****LPR=-2  OMITS GRAPH IDENTIFICATION,OMITS PRINTED DATA ON FILM      RICE8
      C*****LPR=-2  OMITS VPLOT AND CNPLOT  (USE FOR MOVIE OR COLOR PLOT)      RICE8
      C*****LPR=-1  OMITS GRAPH IDENTIFICATION,OMITS PRINTED DATA ON FILM      RICE8
      C*****LPR=-1  OMITS VPLOT                                                RICE8
      C*****LPR= 0  OMITS GRAPH IDENTIFICATION,OMITS PRINTED DATA ON FILM      RICE8
      C*****LPR= 0  OMITS CNPLOT AND COPLOT                                    RICE8
      C*****LPR= 1  ALLOWS ALL PLOTTING AND PRINTING ON FILM,AND PRINTING ON P RICE8
      C*****LPR= 2  ALLOWS ALL PLOTTING AND PRINTING ON FILM,AND PRINTING ON P RICE8
      C*****LPR= 2  OMITS COPLOT                                               RICE8
      C*****LPR= 4  BYPASSES ALL PLOTTING,PRINTS NUM.DATA AT SPECIFIED TIMES   RICE8
      C*****LPR= 5  BYPASSES ALL PRINTING,PLOTS DATA AND GRAPHS                RICE8
227         IF(LPR.EQ.-2)GO TO 2360                                            RICE8
231         IF(LPR.LT.0)GO TO 2167                                             RICE8
232         IF(LPR.EQ.4)GO TO 2170                                             RICE8
234         IF(CYCLE.LE.1)GO TO 2068                                           RICE8
236         IF(TIME+.000001.LT.TPLO)GO TO 2170                                 RICE8
242    2068 CONTINUE                                                           RICE8
      C**** VECTOR PLOT SECTION                                                RICE8
      C**** IF PLOTTING SPECIFIED BY LPR,THEN PLOT ARE MADE WHEN TPLO.LE.TIME  RICE8
      C                                                                        RICE8
242         CALL ADV(1)                                                        RICE8
244         CALL VPLOT(1)                                                      RICE8
246         DO 2165 J=2,JB1                                                    RICE8
250         DO 2165 I=2,IB                                                     RICE8
251         XX1=(I-1.5)*DX                                                     RICE8
254         YY1=(J-1.5)*DY                                                     RICE8
256         IF(FL(I,J).EQ.1) GO TO 2163                                        RICE8
264         IF(FL(I,J).EQ.5) GO TO 2165                                        RICE8
270         IF(FL(I,J).GT.1.AND.FL(I,J).LT.5) GO TO 2165                       RICE8
301         NFL=FL(I,J)-5 $ GO TO (2153,2155,2157,2159,2162,2162,2162,2162)NFL RICE8
321    2153 XX2=XX1+   U(I  ,J)*DXOU $ YY2=YY1+   V(I  ,J)*DXOU $ GO TO 2164   RICE8
335    2155 XX2=XX1+   U(I-1,J)*DXOU $ YY2=YY1+   V(I  ,J)*DXOU $ GO TO 2164   RICE8
351    2157 XX2=XX1+   U(I-1,J)*DXOU $ YY2=YY1+   V(I,J-1)*DXOU $ GO TO 2164   RICE8
365    2159 XX2=XX1+   U(I  ,J)*DXOU $ YY2=YY1+   V(I,J-1)*DXOU $ GO TO 2164   RICE9
404    2163 XX2=XX1+.5*(U(I-1,J)+U(I,J))*DXOU                                  RICE8
412         YY2=YY1+.5*(V(I,J-1)+V(I,J))*DXOU                                  RICE8
420         GO TO 2164                                                         RICE8
421    2162 XX2=XX1 $ YY2=YY1                                                  RICE8
424    2164 CALL VPLOT(2,XX1,YY1,XX2,YY2)                                      RICE8
430    2165 CONTINUE                                                           RICE8
      C**** END OF VECTOR PLOT SECTION--------START OF CONTOUR PLOT SECTION* * RICE8
      C                                                                        RICE8
      C**** PLOTS---RHO,P,SIE,T,M---FOR ALL REAL FLUID CELLS AS SPECIFIED BY   RICE8
      C**** JPLOT(L),EACH ARRAY STORED IN DUMMY ARRAY--CQ--TO TRANSFER TO PLOT RICE8
      C**** ROUTINES                                                           RICE8
      C                                                                        RICE8
435    2167 CONTINUE                                                           RICE8
435         IF(LPR.EQ.0)GO TO 2170                                             RICE8
436         IF(CYCLE.LE.1)GO TO 2169                                           RICE8
441         IF(TIME+.000001.LT.TPLO)GO TO 2170                                 RICE8
```

```
 444          TPLO=TPLO+TPL                                                   RICE8
 446    2169 CONTINUE                                                         RICE8
 446          DO 2325 L=1,5                                                   RICE8
 450          IF(JPLOT(L).NE.1) GO TO 2325                                    RICE8
 452          DO 2320 J=1,JB2                                                 RICE8
 454          DO 2320 I=1,IB2                                                 RICE8
 455          IF(FL(I,J).GT.1.AND.FL(I,J).LT.5) GO TO 2321                    RICE8
 470          GO TO (2307,2309,2311,2313,2315)L                               RICE8
 501    2307 CQ(I,J)=RHO(I,J) $ GO TO 2320                                    RICE8
 510    2309 CQ(I,J)= PB(I,J) $ GO TO 2320                                    RICE8
 517    2311 CQ(I,J)=SIE(I,J) $ GO TO 2320                                    RICE8
 526    2313 CQ(I,J)=TE(I,J)                                                  RICE8
 534          GO TO 2320                                                      RICE8
 535    2315 IF(FL(I,J).GT.1.AND.FL(I,J).LT.6) GO TO 2321                     RICE8
 551          IF(FL(I,J).EQ.1)GO TO 2314                                      RICE8
 556          NFL=FL(I,J)-5 $ GO TO(2316,2317,2318,2319,2316,2317,2318,2319)NFL RICE8
 574    2314 CQ(I,J)=SQRT(((U(I,J)+U(I-1,J))**2+(V(I,J)+V(I,J-1))**2)/       RICE8
             1CSQ(I,J))*.5 $ GO TO 2320                                       RICE8
 625    2316 CQ(I,J)=SQRT((U(I,J)**2+V(I,J)**2)/CSQ(I,J)) $ GO TO 2320        RICE8
 647    2317 CQ(I,J)=SQRT((U(I-1,J)**2+V(I,J)**2)/CSQ(I,J)) $ GO TO 2320      RICE8
 671    2318 CQ(I,J)=SQRT((U(I-1,J)**2+V(I,J-1)**2)/CSQ(I,J)) $ GO TO 2320    RICE8
 713    2319 CQ(I,J)=SQRT((U(I,J)**2+V(I,J-1)**2)/CSQ(I,J)) $ GO TO 2320      RICE8
 735    2321 CQ(I,J)=0.                                                       RICE8
 741    2320 CONTINUE                                                         RICE8
 746          CALL CNPLOT(1)                                                  RICE8
 750          CALL CNPLOT(2,L)                                                RICE8
 752    2325 CONTINUE                                                         RICE8
 754          IF(NSPEC.LT.1) GO TO 2170                                       RICE8
 757    2330 DO 2355 K=1,NSPEC                                                RICE8
 761          IF(ISPEC(K).NE.1) GO TO 2355                                    RICE8
 763          DO 2350 J=1,JB2                                                 RICE8
 765          DO 2350 I=1,IB2                                                 RICE8
 766          IF(FL(I,J).GT.1.AND.FL(I,J).LT.5) GO TO 2350                    RICE8
1001          CQ(I,J)=FA(I,J,K)                                               RICE8
1007          IF(CQ(I,J).LT.0.)CQ(I,J)=0.                                     RICE8
1017    2350 CONTINUE                                                         RICE8
1024          CALL CNPLOT(1)                                                  RICE8
1026          CALL CNPLOT(3,K)                                                RICE8
1030    2355 CONTINUE                                                         RICE8
        C**** END OF CONTOUR PLOT SECTION-----START OF COLOR PLOT SECTION* * * RICE8
        C                                                                     RICE8
        C**** PLOTS SPECIFIED SPECIES (FA) DENSITIES ON ONE FRAME             RICE8
        C**** USES DIFFERENT SYMBOLS FOR BLACK AND WHITE. COPLOT SUBROUTINE MUST RICE8
        C**** BE CHANGED FOR COLOR-USING ALL DOTS AND DIFFERENT COLORS        RICE8
        C                                                                     RICE8
1033    2360 IF(LPR.EQ.0)GO TO 2170                                           RICE8
1034          IF(LPR.EQ.2)GO TO 2170                                          RICE8
1036    2365 CALL ADV(1)                                                      RICE8
1040          CALL COPLOT(1)                                                  RICE8
1042          DO 2375 K=1,NSPEC                                               RICE8
1044          IF(ISPEC(K).NE.1) GO TO 2375                                    RICE8
1046          DO 2370 J=1,JB2                                                 RICE8
1050          DO 2370 I=1,IB2                                                 RICE8
1051          IF(FA(I,J,K).LT.0.)GO TO 2368                                   RICE8
1057          IF(FL(I,J).GT.1.AND.FL(I,J).LT.6) GO TO 2368                    RICE8
```

```
1071         IF(FA(I,J,K).LT.EP(K))GO TO 2368                             RICE8
1105         PTCL=10000./((IB*JB)*RRHO                                    RICE8
1112         CQ(I,J)=FA(I,J,K)*PTCL+.5                                    RICE8
1116         GO TO 2370                                                   RICE8
1117    2368 CQ(I,J)=0.                                                   RICE8
1123    2370 CONTINUE                                                     RICE8
1130         CALL COPLOT(2,K)                                             RICE8
1132    2375 CONTINUE                                                     RICE8
1135    2170 CONTINUE                                                     RICE8
1135         CALL SECOND(TNOW)                                            RICE8
        C**** RELEASES DISK DATA TO TAPE EVERY 35 MINUTES  * * * * * * * * * * * * RICE8
1137         IF(TNOW-TBEG.LT.TRDISK) GO TO 2174                           RICE8
1143         TRDISK=TRDISK+2100.                                          RICE8
1144         CALL DATAREL(5LFSET5)                                        RICE8
1146    2174 CONTINUE                                                     RICE8
1146         KTAPE=9                                                      RICE8
        C**** WRITES FLUID VARIABLES U,V,RHO,P,I,T,MU (TAPE-9) ALSO ON FILM* * * RICE8
1147         IF(LPR.EQ.5)GO TO 2266                                       RICE8
1151         IF(SECREQ-20..LE.TNOW-TBEG)GO TO 2248                        RICE8
1156         IF(CYCLE.LT.0)GO TO 2248                                     RICE8
1160         IF(TIME+.000001.LT.TPRI)GO TO 2266                           RICE8
1163         TPRI=TPRI+TPR                                                RICE8
1165    2248 CONTINUE                                                     RICE8
1165    2250 CALL ADV(2)                                                  RICE8
1167    2252 WRITE(KTAPE,9)CYCLE,TIME,NAME                                RICE8
1201         WRITE(KTAPE,30)                                              RICE8
1205         IJC=0                                                        RICE8
1206         DO 2265 J=1,JB2                                              RICE8
1210         DO 2265 I=1,IB2                                              RICE8
1211         IJC=IJC+1                                                    RICE8
1212         IF(IJC.LT.56)GO TO 2255                                      RICE8
1214         IJC=0                 $ WRITE(KTAPE,9)CYCLE,TIME,NAME        RICE8
1226         WRITE(KTAPE,30)                                              RICE8
1232    2255 WRITE(KTAPE,31)I,J,FL(I,J),U(I,J),V(I,J),RHO(I,J),PB(I,J)    RICE8
            1,SIE(I,J),TE(I,J),MU(I,J)                                    RICE8
1332    2265 CONTINUE                                                     RICE8
1337         IF(NSPEC.LT.1)GO TO 2264                                     RICE8
1342         CALL ADV(1)                                                  RICE8
1343         WRITE(KTAPE,9)CYCLE,TIME,NAME                                RICE8
1355         WRITE(KTAPE,32)                                              RICE8
1361         WRITE(KTAPE,33)                                              RICE8
1365         IJC=0                                                        RICE8
        C**** WRITES SPECIES DENSITIES (TAPE-9) ALSO ON FILM (TAPE-12) * * * * * RICE8
1366         DO 2263 J=1,JB2                                              RICE8
1370         DO 2263 I=1,IB2                                              RICE8
1371         IF(IJC.LT.56)GO TO 2262                                      RICE8
1373         IJC=0                                                        RICE8
1374         WRITE(KTAPE,9)CYCLE,TIME,NAME                                RICE8
1405         WRITE(KTAPE,32) $ WRITE(KTAPE,33)                            RICE8
1415    2262 WRITE(KTAPE,31)I,J,FL(I,J),(FA(I,J,K),K=1,NSPEC)             RICE8
1451         IJC=IJC+1                                                    RICE8
1453    2263 CONTINUE                                                     RICE8
1460    2264 CONTINUE                                                     RICE8
1460    2266 IF(LPR.EQ.4)GO TO 2267                                       RICE8
1462         IF(LPR.LE.0)GO TO 2267                                       RICE8
```

```
1463          IF(SECREQ-20..LE.TNOW-TBEG)GO TO 2268
1470          IF(CYCLE.LE.1)GO TO 2268
1473          IF(TIME+.000001.LT.TPLOD)GO TO 2267
1476          TPLOD=TPLOD+TPLD
1500     2268 IF(KTAPE.EQ.12)GO TO 2267
1502          CALL ADV(1)
1504          KTAPE=12 $ GO TO 2252
1506     2267 CONTINUE
1506          IF(ITD.EQ.1.OR.ITD.EQ.3)GO TO 2270
1516          GO TO 2280
1516     2270 CONTINUE
1516          IF(SECREQ-20..LE.TNOW-TBEG)GO TO 2274
1523          IF(CYCLE.EQ.NSDMP)GO TO 2274 $ GO TO 2280
1526     2274 WRITE(9,3)NWDMP,CYCLE,TIME
         C
         C**** WRITES DATA ON TAPE5 FOR RESTART IF ITD = 1 OR 3
1540          NSDMP=NSDMP+NCYDMP
1542          WRITE(5) NWDMP,IB2,JB2
1553          WRITE(5)((U(I,J),V(I,J),RHO(I,J),SIE(I,J),I=1,IB2),J=1,JB2)
1621          IF(NSPEC.LT.1)GO TO 2278
1624          WRITE(5)(((FA(I,J,K),I=1,IB2),J=1,JB2),K=1,NSPEC)
1647     2278 CONTINUE
1647          IF(SECREQ-20..LE.TNOW-TBEG) CALL EXIT
1655          NWDMP=NWDMP+1
1657     2280 CONTINUE
1657          IF(SECREQ-20..LE.TNOW-TBEG)CALL EXIT
         C*******************END OF CYCLE,PRINT AND PLOT FINISHED**************
         C**** START OF NEW CYCLE----ALL QUANTITIES CORRESPOND IN TIME-----------
         C**** STOPS CALC. 20 SEC. BEFORE REQUESTED TIME ON JOB CARD USED TO
         C**** ALLOW TIME FOR OUTPUT
         C
         C
1665          IF(TIME+.000001.GE.TSTOP)GO TO 5000
         C**** CALCULATES THE NECESSARY TILDE QUANTITIES--RHOVT,RHOUT AND RHOT **
1671          CALL TILDE
         C    REGION 4000 DETERMINES NEW CELL BOUNDARY VELOCITIES
         C
         C**** VELOCITIES ARE DETERMINED FROM RHOU,RHOV,AND RHO
         C
1672          JC=0
1673     4000 DO 4400 J=2,JB1
1675          JC=JC+IB2
1676          DO 4400 I=2,IB1
1700          IJ=I+JC
1701          IPJ=IJ+1
1702          IJP=IJ+IB2
1704          NFL=FL(IJ)
1707          GO TO (4050,4200,4200,4200,4200,4050,4200,4200,4050,4050,4200,4200
             1,4050)NFL
1727     4050 NFL=FL(IPJ)
1732          GO TO (4100,4200,4200,4150,4200,4200,4100,4100,4200,4200,4100,4100
             1,4200)NFL
1752     4100 U(IJ)=2.*RHOU (IJ)/(RHO(IJ)+RHO(IPJ)) $ GO TO 4200
1763     4150 U(IJ)=RHOU(IJ)/RHO(IJ) $ GO TO 4200
1770     4200 NFL=FL(IJ)
```

```
1773        GO TO (4250,4400,4400,4400,4400,4250,4250,4400,4400,4250,4250,4400 RICE6
           1,4400)NFL                                                          RICE8
2013   4250 NFL=FL(IJP)                                                        RICE8
2016        GO TO (4300,4400,4400,4400,4400,4400,4400,4300,4300,4400,4400,4300 RICE8
           1,4300)NFL                                                          RICE8
2036   4300 V(IJ)=2.*RHOV(IJ)/(RHO(IJ)+RHO(IJP))                               RICE8
2046   4400 CONTINUE                                                           RICE8
      C**** CALCULATES TOTAL AND SPECIFIC INTERNAL ENERGY            * * * * * RICE8
2053        CALL ENERGY                                                        RICE8
2054        IF(NSPEC.LE.1)GO TO 4405                                           RICE8
      C**** CALCULATES SPECIES DENSITIES FROM CONVECTION,DIFFUSION,CHEM.REACT. RICE8
2057        CALL SPEC                                                          RICE8
2060   4405 CONTINUE                                                           RICE8
      C                                                                        RICE8
      C**** MASS,MOMENTUM,ENERGY AND CHEMISTRY CALC.UPDATED TO TIME+DT         RICE8
      C**** P,C**2 AND MU CAN BE UPDATED AND CYCLE WILL BE COMPLETE,PRINTING   RICE8
      C**** AND PLOTTING CAN BE DONE                                           RICE8
      C                                                                        RICE8
2060        TIME=TIME+DT                                                       RICE8
2062        CYCLE=CYCLE+1                                                      RICE8
2064        TCPOLD=TCP                                                         RICE8
2065        CALL SECOND(TCP)                                                   RICE8
2067        CPT=TCP-TBEG                                                       RICE8
2071        CPTG=TCP-TCPOLD                                                    RICE8
2072        GRINDS=CPTG/(IB*JB)                                                RICE8
2077        PRINT 41,IT,TIME,CYCLE,CPT,GRINDS                                  RICE8
2115        GO TO 1000                                                         RICE8
2116   5000 CONTINUE                                                           RICE8
2116        RETURN                                                             RICE8
2117        END                                                                RICE8
      C                                                                        RICE8
      C                                                                        RICE8
      C**** END OF THE SUBROUTINE PROG *************************************** RICE8
      C                                                                        RICE8
      C                                                                        RICE8
```

```
            SUBROUTINE RTAPE5                                                    RICE8      542
        C**** TAPE-5 DATA FOR STORAGE AND RESTART                                RCOM1        2
     1         COMMON/TAPED/ITD,NTD,NSDMP,NFILE,NWDMP,NCYDMP                     RCOM1        3
        C**** THE RCOM1 DATA BLOCK IS USED IN RICE(MAIN-PROGRAM),PROG AND RTAPE5 RCOM1        4
        C                                                                        RCOM1        5
        C**** THE RCOM2 DATA BLOCK IS COMMON TO RICE AND ALL SUBROUTINES         RCOM1        6
        C**** BOTH RCOM1 AND RCOM2 ARE LOADED USING *COMDECK FEATURE OF (UPDATE) RCOM1        7
     1         COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,      RCOM2        2
              1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC,ALA,BIE,THETA,   RCOM2        3
              2PHI,OMEGA,GAMMA,DXOU,ASQ                                          RCOM2        4
     1         COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SECREQ,DDX,DDY        RCOM2        5
              1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT                             RCOM2        6
     1         COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1,   RCOM2        7
              1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,   RCOM2        8
              2SIEL1,SIEL2                                                       RCOM2        9
     1         REAL MUO,MINT1,MINT2,MINL1,MINL2                                  RCOM2       10
     1         COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                         RCOM2       11
              1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),       RCOM2       12
              2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)            RCOM2       13
     1         COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20) RCOM2      14
              1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),               RCOM2       15
              2SIEBO(20),FARO(7,20),FABO(7,20)                                   RCOM2       16
     1         INTEGER AM,BM                                                     RCOM2       17
     1         REAL MINRO,MINBO                                                  RCOM2       18
        C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP) RCOM2       19
        C**** AND THE *READ FEATURE OF (UPDATE)                                  RCOM2       20
        C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/         RCOM2       21
        C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC   RCOM2       22
        C                                                                        RCOM2       23
     1            LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32                     RCOM2       24
             1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                              RCOM2       25
             2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62                  RCOM2       26
             3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(                   RCOM2       27
             4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                               RCOM2       28
             5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                            RCOM2       29
     1         DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                       RCOM2       30
             1 62, 32),DRHOB( 62, 7),STREB( 62)                                  RCOM2       31
     1         EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),        RCOM2       32
             1(TFAB,DRHOB)                                                       RCOM2       33
     1         INTEGER FL,CYCLE                                                  RCOM2       34
     1         REAL MU                                                           RCOM2       35
             1 FORMAT(24I3)                                                      RICE8      545
     1      68 READ(5) NTDMP,IB2,JB2                                             RICE8      546
    12         WRITE(9,1)NTDMP,NTD                                               RICE8      547
    22         IF(NTD.EQ.NTDMP)GO TO 70                                          RICE8      548
    24         READ (5)                                                          RICE8      549
    27         READ (5)                                                          RICE8      550
    32         IF(NTDMP.EQ.NFILE)GO TO 75                                        RICE8      551
    34         IF(EOF,5)75,68                                                    RICE8      552
    37      70 READ (5)((U(I,J),V(I,J),RHO(I,J),SIE(I,J),I=1,IB2),J=1,JB2)       RICE8      553
   105         IF(NSPEC.LT.1)GO TO 73                                            RICE8      554
   110         READ (5)(((FA(I,J,K),I=1,IB2),J=1,JB2),K=1,NSPEC)                 RICE8      555
   133      73 CONTINUE                                                          RICE8      556
   133         IF(NTDMP.NE.NFILE)GO TO 68                                        RICE8      557
   135      75 CONTINUE                                                          RICE8      558
```

```
135         RETURN $ END                                                    RICE8
      C                                                                     RICE8
      C                                                                     RICE8
      C**** END OF THE SUBROUTINE RTAPES ************************************ RICE8
      C                                                                     RICE8
      C                                                                     RICE8
```

```
          SUBROUTINE TILDE                                                    RICE8    565
 1        COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,         RCOM2      2
         1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC,ALA,BIE,THETA,     RCOM2      3
         2PHI,OMEGA,GAMMA,DXOU,ASQ                                            RCOM2      4
 1        COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SECREC,DDX,DDY          RCOM2      5
         1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT                               RCOM2      6
 1        COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1,     RCOM2      7
         1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,     RCOM2      8
         2SIEL1,SIEL2                                                         RCOM2      9
 1        REAL MUO,MINT1,MINT2,MINL1,MINL2                                    RCOM2     10
 1        COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                           RCOM2     11
         1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),         RCOM2     12
         2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)              RCOM2     13
 1        COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20) RCOM2     14
         1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),                 RCOM2     15
         2SIEBO(20),FARO(7,20),FABO(7,20)                                     RCOM2     16
 1        INTEGER AM,BM                                                       RCOM2     17
 1        REAL MINRO,MINBO                                                    RCOM2     18
   C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP)   RCOM2     19
   C**** AND THE *READ FEATURE OF (UPDATE)                                    RCOM2     20
   C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/           RCOM2     21
   C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC     RCOM2     22
   C                                                                          RCOM2     23
 1            LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32                      RCOM2     24
         1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                               RCOM2     25
         2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62                   RCOM2     26
         3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(                    RCOM2     27
         4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                                RCOM2     28
         5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                             RCOM2     29
 1        DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                         RCOM2     30
         1 62, 32),DRHOB( 62, 7),STREB( 62)                                   RCOM2     31
 1        EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),          RCOM2     32
         1(TFAB,DRHOB)                                                        RCOM2     33
 1        INTEGER FL,CYCLE                                                    RCOM2     34
 1        REAL MU                                                             RCOM2     35
 1        JC=0                                                                RICE8    567
   C                                                                          RICE8    568
   C**** START OF CALC.FOR MASS EQ.,INTERMEDIATE SOLUTION (RHOT)              RICE8    569
   C                                                                          RICE8    570
 2        DO 2000 J=2,JB1                                                     RICE8    571
 4        JC=JC+IB2                                                           RICE8    572
 5        DO 2000 I=2,IB1                                                     RICE8    573
 7        IJ=I+JC                                                             RICE8    574
10        IPJ=IJ+1                                                            RICE8    575
11        IMJ=IJ-1                                                            RICE8    576
12        IJP=IJ+IB2                                                          RICE8    577
14        IJM=IJ-IB2                                                          RICE8    578
14        IMJM=IJM-1                                                          RICE8    579
15        IPJM=IJM+1                                                          RICE8    580
16        IMJP=IJP-1                                                          RICE8    581
20        IPJP=IJP+1                                                          RICE8    582
21        NFL=FL(IJ) $ GO TO(1805,2000,2000,2000,2000,1810,1820,1820,1810,    RICE8    583
         11815,1820,1820,1815)NFL                                             RICE8    584
44   1805 UL=U(IMJ) $ GO TO 1825                                              RICE8    585
50   1810 UL=U(IJ)   $ GO TO 1825                                             RICE8    586
```

```
 54   1815 UL=-U(IJ)  $ GO TO 1825                                          RICE8
 60   1820 RFLR=0. $ GO TO 1865                                             RICE8
 62   1825 NFL=FL(IPJ) $ GO TO(1830,1820,1820,1860,1820,1820,1835,1835,1820  RICE8
          1,1820,1840,1840,1820)NFL                                         RICE8
105   1830 UR=U(IPJ) $ GO TO 1845                                           RICE8
111   1835 UR=U(IJ)    $ GO TO 1845                                         RICE8
     C**** TAUTE=MASS DIFFUSION TRUNCATION ERROR COEF.            . . . . . . . . RICE8
     C                                                                      RICE8
115   1840 UR=-U(IJ)                                                        RICE8
120   1845 TAUTE=-0.5*DT*(2.*THETA-1.)*(U(IJ)**2+0.5*(CSQ(IPJ)+CSQ(IJ)))    RICE8
          1+0.125*DX*(UR-UL)                                                RICE8
141         IF(TAUTE.LT.0.)GO TO 1850                                       RICE8
143         TAUR=(1.0+XI)*TAUTE $ GO TO 1855                                RICE8
146   1850 TAUR=(1.0-XI)*TAUTE                                              RICE8
     C**** RFLR=MASS FLUX ACROSS RIGHT CELL BOUNDARY               . . . . . . . . RICE8
     C                                                                      RICE8
151   1855 RFLR=(1.0-THETA)*RHOU(IJ)-DDX*TAUR*(RHO(IPJ)-RHO(IJ))            RICE8
164         GO TO 1865                                                      RICE8
164   1860 RFLR=(1.0-THETA)*RHO(IJ)*U(IJ)                                   RICE8
172   1865 IF(FL(IJ).EQ.1) GO TO 1870                                       RICE8
176         NFL=FL(IJ)-5 $ GO TO(1875,1875,1887,1887,1880,1880,1887,1887)NFL RICE8
213   1870 VB=V(IJM) $ GO TO 1885                                           RICE8
217   1875 VB=V(IJ)    $ GO TO 1885                                         RICE8
223   1880 VB=-V(IJ)                                                        RICE8
226   1885 NFL=FL(IJP) $ GO TO(1890,1887,1887,1887,1920,1887,1887,1895,1895  RICE8
          1,1887,1887,1900,1900)NFL                                         RICE8
251   1887 RFLT=0. $ GO TO 1935                                             RICE8
253   1890 VT=V(IJP) $ GO TO 1905                                           RICE8
257   1895 VT=V(IJ)    $ GO TO 1905                                         RICE8
263   1900 VT=-V(IJ)                                                        RICE8
266   1905 TAUTE=-0.5*DT*(2.*THETA-1.)*(V(IJ)**2+0.5*(CSQ(IJP)+CSQ(IJ)))    RICE8
          1   +.125*DY*(VT*RJ2(J+1)-VB*RJ2(J-1))/RJ2(J)                      RICE8
314         IF(TAUTE.LT.0.)GO TO 1910                                       RICE8
316         TAUT=(1.0+XI)*TAUTE $ GO TO 1915                                RICE8
321   1910 TAUT=(1.0-XI)*TAUTE                                              RICE8
     C**** RFLT=MASS FLUX ACROSS TOP OF CELL BOUNDARY              . . . . . . . . RICE8
     C                                                                      RICE8
324   1915 RFLT=((1.0-THETA)*RHOV(IJ)-DDY*TAUT*(RHO(IJP)-RHO(IJ)))*RJ2(J)   RICE8
341         GO TO 1935                                                      RICE8
341   1920 IF(V(IJ).EQ.0.)GO TO 1887                                        RICE8
344         TAUTE=-.5*DT*(2.0*THETA-1.0)*(V(IJ)**2+GAMMA*(GAMMA-1.)*SIE(IJP)) RICE8
          1   +.25*DY*(V(IJ)*RJ2(J)-VB*RJ2(J-1))/RJ2(J)                      RICE8
371         IF(TAUTE.LT.0.)GO TO 1925                                       RICE8
373         TAUT=(1.+XI)*TAUTE $ GO TO 1930                                 RICE8
376   1925 TAUT=(1.-XI)*TAUTE                                               RICE8
401   1930 RFLT=((1.-THETA)*RHO(IJP)*V(IJ)-2.*DDY*TAUT*(RHO(IJP)-RHO(IJ)))*RJ RICE8
          12(J)                                                             RICE8
422   1935 IF(FL(IJ).EQ.1)GO TO 1940                                        RICE8
426         NFL=FL(IJ)-5 $ GO TO(1960,1945,1945,1960,1960,1950,1950,1960)NFL RICE8
443   1940 UR=U(IJ)    $ GO TO 1955                                         RICE8
447   1945 UR=U(IMJ) $ GO TO 1955                                           RICE8
453   1950 UR=-U(IMJ)                                                       RICE8
456   1955 NFL=FL(IMJ) $ GO TO(1972,1960,1960,1960,1962,1972,1960,1960,1972  RICE8
          1,1972,1960,1960,1972)NFL                                         RICE8
501   1960 RFLL=0. $ GO TO 1972                                             RICE8
```

```
503    1962 IF(U(IMJ).EQ.0.)GO TO 1960                                          RICE8
506         TAUTE=-.5*DT*(2.0*THETA-1.0)*(U(IMJ)**2+GAMMA*(GAMMA-1.)*SIE(IMJ)) RICE8
           1+0.125*DX*(UR-U(IMJ))                                               RICE8
526         IF(TAUTE.LT.0.)GO TO 1965                                           RICE8
530         TAUL=(1.+XI)*TAUTE $ GO TO 1970                                     RICE8
533    1965 TAUL=(1.-XI)*TAUTE                                                  RICE8
536    1970 RFLL=(1.-THETA)*RHO(IMJ)*U(IMJ)-2.*DDX*TAUL*(RHO(IJ)-RHO(IMJ))      RICE8
553    1972 IF(FL(IJ).EQ.1)GO TO 1980                                           RICE8
557         NFL=FL(IJ)-5 $ GO TO(1975,1975,1980,1980,1975,1975,1980,1980)NFL    RICE8
574    1975 RFLB(I)=0. $ GO TO 1985                                             RICE8
577    1980 NFL=FL(IJM) $ GO TO(1985,1975,1975,1975,1975,1985,1985,1975,1975    RICE8
           1,1985,1985,1975,1975)NFL                                            RICE8
624    1985 RHOT(IJ)=RHO(IJ)+DT*(DDX*(RFLL-RFLR)+DDY/RJ(J)*(RFLB(I)-RFLT))      RICE8
637         RFLL=RFLR                                                           RICE8
641         RFLB(I)=RFLT                                                        RICE8
644    2000 CONTINUE                                                            RICE8
     C**** INTERMEDIATE MASS DENSITY CALC.COMPLETE                  * * * * * * * * RICE8
     C                                                                          RICE8
     C**** START OF CALC.FOR MOMENTUM EQ.,INTERMEDIATE SOLUTION (RHOUT-RHOVT)    RICE8
     C                                                                          RICE8
651         JC=0                                                                RICE8
652         DO 2995 J=2,JB1                                                     RICE8
653         JC=JC+IB2                                                           RICE8
654         IBJ=JC+IB                                                           RICE8
655         IB1J=IBJ+1                                                          RICE8
657         IB2J=IBJ+2                                                          RICE8
661         DO 2990 I=2,IB1                                                     RICE8
662         IJ=I+JC                                                             RICE8
663         IF(FL(IJ).GT.1.AND.FL(IJ).LE.5)GO TO 2990                           RICE8
676         IPJ=IJ+1                                                            RICE8
676         IMJ=IJ-1                                                            RICE8
677         IJP=IJ+IB2                                                          RICE8
701         IJM=IJ-IB2                                                          RICE8
701         IMJM=IJM-1                                                          RICE8
702         IPJM=IJM+1                                                          RICE8
703         IMJP=IJP-1                                                          RICE8
705         IPJP=IJP+1                                                          RICE8
705         IF(FL(IJ).EQ.1)GO TO 2520                                           RICE8
710         NFL=FL(IJ)-5 $ GO TO(2520,2582,2990,2550,2520,2582,2990,2550)NFL    RICE8
     C**** DETERMINES WHETHER TO CALC.OR SKIP RHOUT(I,J)                        RICE8
     C**** CALC.IF FL(I,J)=1,6,9,10 OR 13 AND IF(FL(I+1,J)=1,7,8,11 OR 12       RICE8
     C                                                                          RICE8
726    2520 IRV=1                                                               RICE8
727    2522 IF(FL(IPJ).EQ.1)GO TO 2600                                          RICE8
733         IF(FL(IPJ).LT.6)GO TO 2580                                          RICE8
736         NFL=FL(IPJ)-5                                                       RICE8
741         GO TO(2580,2600,2600,2580,2580,2600,2600,2580)NFL                   RICE8
754    2550 IRV=2 $ GO TO 2522                                                  RICE8
     C**** DETERMINES WHETHER TO CALC.OR SKIP RHOVT(IJ)                         RICE8
     C**** CALC.IF FL(I,J)=1,6,7,10 OR 11 AND IF (FL(I,J+1)=1,8,9,12 OR 13      RICE8
     C                                                                          RICE8
756    2580 IF(IRV.EQ.2)GO TO 2990                                              RICE8
760    2582 IF(FL(IJP).EQ.1)GO TO 2800                                          RICE8
764         IF(FL(IJP).LT.6)GO TO 2990                                          RICE8
767         NFL=FL(IJP)-5                                                       RICE8
```

```
 772        GO TO(2990,2990,2800,2800,2990,2990,2800,2800)NFL                  RICE8
      C**** CALC.RHOUT,FLUID ON BOTH SIDES OF  U                               RICE8
      C                                                                        RICE8
 1005  2600 CONTINUE                                                           RICE8
 1007       PC=P(IJ)                                                           RICE8
 1011       RUOLD=.5*U(IJ)*(RHO(IJ)+RHO(IPJ))                                  RICE8
 1017       PR=P(IPJ)                                                          RICE8
 1021       NFL=FL(IJM)                                                        RICE8
 1024       GO TO(2615,2612,2612,2612,2612,2615,2615,2612,2612,2615,2615,2612, RICE8
           12612)NFL                                                           RICE8
 1044  2612 RHOB=RHO(IJ) $ VISB=MU(IJ) $ GO TO 2625                            RICE8
 1052  2615 RHOB=RHO(IJM) $ VISB=MU(IJM) $ GO TO 2625                          RICE8
 1060  2625 NFL=FL(IPJM) $ GO TO(2635,2630,2630,2630,2630,2635,2635,2630,      RICE8
           12630,2635,2635,2630,2630)NFL                                       RICE8
 1103  2630 RHOBR=RHO(IPJ) $ VISBR=MU(IPJ) $ GO TO 2640                        RICE8
 1111  2635 RHOBR=RHO(IPJM) $ VISBR=MU(IPJM) $ GO TO 2640                      RICE8
 1117  2640 CONTINUE                                                           RICE8
 1117       NFL=FL(IJP) $ GO TO(2655,2642,2642,2642,2642,2642,2642,2655,       RICE8
           12655,2642,2642,2655,2655)NFL                                       RICE8
 1142  2642 IF(FL(IJP).NE.5)GO TO 2650                                         RICE8
 1146       IF(V(IJ).EQ.0.)GO TO 2648                                          RICE8
 1150       RTOP=2.*RHO(IJP)-RHO(IJ) $ VIST=2.*MU(IJP)-MU(IJ) $GOTO 2660       RICE8
 1163  2648 RTOP=2.*RHO(IJP)-RHO(IPJP) $ VIST=2.*MU(IJP)-MU(IPJP)             RICE8
 1177       GO TO 2660                                                         RICE8
 1177  2650 RTOP=RHO(IJ) $ VIST=MU(IJ) $ GO TO 2660                            RICE8
 1205  2655 RTOP=RHO(IJP) $ VIST=MU(IJP) $ GO TO 2660                          RICE8
 1213  2660 NFL=FL(IPJP) $ GO TO(2670,2665,2665,2665,2665,2665,2665,2670,      RICE8
           12670,2665,2665,2670,2670)NFL                                       RICE8
 1236  2665 IF(FL(IPJP).NE.5)GO TO 2668                                        RICE8
 1242       IF(V(IPJ).EQ.0.)GO TO 2668                                         RICE8
 1244       RHOTR=2.*RHO(IPJP)-RHO(IPJ) $ VISTR=2.*MU(IPJP)- MU(IPJ)           RICE8
 1257       GO TO 2675                                                         RICE8
 1257  2668 RHOTR=RHO(IPJ)                                                     RICE8
 1262       VISTR=MU(IPJ) $ GO TO 2675                                         RICE8
 1265  2670 RHOTR=RHO(IPJP)                                                    RICE8
 1270       VISTR=MU(IPJP) $ GO TO 2675                                        RICE8
 1273  2675 CONTINUE                                                           RICE8
 1273       NFL=FL(IJ) $ GO TO(2677,2685,2685,2685,2685,2678,2685,2685,        RICE8
           12678,2679,2685,2685,2680)NFL                                       RICE8
 1322  2677 SZZC=MU(IJ)*((2.+ALA)*(U(IJ)-U(IMJ))*DDX+ALA*DDY/RJ(J)*            RICE8
           1(RJ2(J)*V(IJ)-RJ2(J-1)*V(IJM)))                                    RICE8
 1342       RU2C=.25*RHO(IJ)*(U(IJ)+U(IMJ))**2                                 RICE8
 1350       GO TO 2685                                                         RICE8
 1352  2678 SZZC=0.                                                           RICE8
 1353       RU2C=RHO(IJ)*U(IJ)**2                                              RICE8
 1357       GO TO 2685                                                         RICE8
 1360  2679 SZZC=2.*MU(IJ)*((2.+ALA)*U(IJ)*DDX+ALA*DDY*V(IJ))                  RICE8
 1372       RU2C=0.                                                            RICE8
 1373       GO TO 2685                                                         RICE8
 1374  2680 SZZC=2.*MU(IJ)*((2.+ALA)*U(IJ)*DDX-ALA*DDY*V(IJM))                 RICE8
 1407       RU2C=0.                                                            RICE8
 1411       GO TO 2685                                                         RICE8
 1411  2685 NFL=FL(IPJ) $ GO TO(2686,2690,2690,2690,2690,2690,2687,2687,       RICE8
           12690,2690,2688,2689,2690)NFL                                       RICE8
 1440  2686 SZZR=MU(IPJ)*((2.+ALA)*DDX*(U(IPJ)-U(IJ))+ALA*DDY/RJ(J)*           RICE8
```

```
                1(RJ2(J)*V(IPJ)-RJ2(J-1)*V(IPJM)))                            RICE6
1460            RU2R=.25*RHO(IPJ)*(U(IPJ)+U(IJ))**2                            RICE8
1466            GO TO 2690                                                     RICE8
1467      2687  SZZR=0.                                                        RICE8
1470            RU2R=RHO(IPJ)*U(IJ)**2                                         RICE8
1475            GO TO 2690                                                     RICE8
1476      2688  SZZR=2.*MU(IPJ)*(-(2.+ALA)*DDX*U(IJ)+ALA*DDY*V(IPJ))           RICE8
1511            RU2R=0.                                                        RICE8
1512            GO TO 2690                                                     RICE8
1513      2689  SZZR=2.*MU(IPJ)*(-(2.+ALA)*DDX*U(IJ)-ALA*DDY*V(IPJM))          RICE8
1527            RU2R=0.                                                        RICE8
1531            GO TO 2690                                                     RICE8
1531      2690  IF(FL(IPJ).EQ.7)GO TO 2695                                     RICE8
1535            IF(FL(IPJ).EQ.11)GO TO 2693                                    RICE8
1540            VBR=V(IPJM) $ GO TO  2696                                      RICE8
1543      2693  VBR=-V(IJM) $ GO TO 2696                                       RICE8
1547      2695  VBR=V(IJM) $ GO TO 2696                                        RICE8
1553      2696  IF(FL(IJ).EQ.6)GO TO 2699                                      RICE8
1557            IF(FL(IJ).EQ.10)GO TO 2698                                     RICE8
1562            VB=V(IJM) $ GO TO 2700                                         RICE8
1565      2698  VB=-V(IPJM) $ GO TO 27..J                                      RICE8
1571      2699  VB= V(IPJM) $ GO TO 2700                                       RICE8
1575      2700  IF(FL(IPJM).EQ.6)GO TO 2701                                    RICE8
1601            IF(FL(IPJM).EQ.9)GO TO 2701                                    RICE8
1604            IF(FL(IPJM).EQ.10)GO TO 2701                                   RICE8
1607            IF(FL(IPJM).EQ.13)GO TO 2701                                   RICE8
1612            IF(FL(IJM).EQ. 7)GO TO 2703                                    RICE8
1615            IF(FL(IJM).EQ. 8)GO TO 2703                                    RICE8
1620            IF(FL(IJM).EQ.11)GO TO 2703                                    RICE8
1623            IF(FL(IJM).EQ.12)GO TO 2703 $GO TO 2702                        RICE8
1627      2701  IF(FL(IJM).EQ. 2)GO TO 2708                                    RICE8
1633            IF(FL(IJM).EQ. 3)GO TO 2709                                    RICE8
1636      2702  UBR=U(IJM) $ GO TO 2710                                        RICE8
1642      2703  IF(FL(IPJM).EQ.2)GO TO 2708                                    RICE8
1646            IF(FL(IPJM).EQ.3)GO TO 2709 $ GO TO 2702                       RICE8
1651      2708  UBR= U(IJ) $ GO TO 2710                                        RICE8
1655      2709  UBR=-U(IJ) $ GO TO 2710                                        RICE8
1663      2710  SRZBR=.25*(VISB+VISBR+MU(IPJ)+MU(IJ))*((VBR-VB)*DDX            RICE8
               1+(U(IJ)-UBR)*DDY)                                              RICE8
1700            RUVBR=(RHO(IJ)+RHO(IPJ)+RHOB+RHOBR)*(VB+VBR)*(U(IJ)+UBR)*.0625 RICE8
1714            IF(FL(IJ).EQ. 6.AND.U(IJM).NE.0.)SRZBR=SRZBR*.5               RICE8
1730            IF(FL(IJ).EQ.10.AND.U(IJM).NE.0.)SRZBR=SRZBR*.5               RICE8
1743            IF(FL(IPJ).EQ.8)GO TO 2715                                     RICE8
1747            IF(FL(IPJ).EQ.12)GO TO 2712                                    RICE8
1752            VTR=V(IPJ) $ GO TO 2720                                        RICE8
1754      2712  VTR=-V(IJ) $ GO TO 2720                                        RICE8
1760      2715  VTR=V(IJ) $ GO TO 2720                                         RICE8
1764      2720  IF(FL(IJ).EQ.9)GO TO 2723                                      RICE8
1770            IF(FL(IJ).EQ.13)GO TO2722                                      RICE8
1773            VT=V(IJ) $ GO TO 2725                                          RICE8
1775      2722  VT=-V(IPJ) $ GO TO 2725                                        RICE8
2001      2723  VT=V(IPJ) $ GO TO 2725                                         RICE8
2005      2725  IF(FL(IPJP).EQ. 6)GO TO 2727                                   RICE8
2011            IF(FL(IPJP).EQ. 9)GO TO 2727                                   RICE8
2014            IF(FL(IPJP).EQ.10)GO TO 2727                                   RICE8
```

```
2017         IF(FL(IPJP).EQ.13)GO TO 2727
2022         IF(FL(IJP).EQ. 7)GO TO 2729
2025         IF(FL(IJP).EQ. 8)GO TO 2729
2030         IF(FL(IJP).EQ.11)GO TO 2729
2033         IF(FL(IJP).EQ.12)GO TO 2729 $GO TO 2728
2037    2727 IF(FL(IJP).EQ. 2)GO TO 2735
2043         IF(FL(IJP).EQ. 3)GO TO 2740
2046    2728 IF(FL(IJP).EQ.5.AND.V(IJ).NE.0.)UTR=2.*U(IJP)-U(IJ)
2065         IF(FL(IJP).NE.5)UTR=U(IJP)
2073         GO TO 2750
2074    2729 IF(FL(IPJP).EQ.2)GO TO 2735
2100         IF(FL(IPJP).EQ.3)GO TO 2740 $ GO TO 2728
2103    2735 UTR= U(IJ) $ GO TO 2750
2107    2740 UTR=-U(IJ) $ GO TO 2750
2113    2750 CONTINUE
2115         SRZTR=.25*(MU(IJ)+MU(IPJ)+VIST+VISTR)*( (VTR -VT)*DDX+
            1(UTR-U(IJ))*DDY)
2132         RUVTR=(RHO(IJ)+RHO(IPJ)+RHOTR+RTOP)*(VT+VTR)*(U(IJ)+UTR)*.0625
2145         IF(FL(IJ).EQ. 9.AND.U(IJP).NE.0.)SRZTR=SRZTR*.5
2161         IF(FL(IJ).EQ.13.AND.U(IJP).NE.0.)SRZTR=SRZTR*.5
2175         T1=(RJ2(J-1)*(RUVBR-SRZBR)-RJ2(J)*(RUVTR-SRZTR))/(RJ(J)*DY)
2210         T2=(RU2C+ PC-SZZC-RU2R-PR+SZZR)*DDX
2217         RHOUT(IJ)=RUOLD+DT*(T1+T2)
      C**** INTERMEDIATE  --  AXIAL MOMENTUM CALC.COMPLETE   * * * * * * * *
      C
      C**** FINISHED RHOU(I,J) CALC.------START RHOV(I,J) CALC.
      C
2224         GO TO 2580
2224    2800 CONTINUE
2226         PC=P(IJ)
2230         RVOLD=.5*V(IJ)*(RHO(IJ)+RHO(IJP))
2236         PT=P(IJP)
2240         NFL=FL(IMJ) $ GO TO(2801,2805,2805,2805,2803,2801,2805,2805,
            12801,2801,2805,2805,2801)NFL
2263    2801 RHOL=RHO(IMJ) $ VISL=MU(IMJ) $ GO TO 2807
2271    2803 IF(U(IMJ).EQ.0) GO TO 2805
2274         RHOL=2.*RHO(IMJ)-RHO(IJ) $ VISL=2.*MU(IMJ)-MU(IJ) $GOTO 2807
2307    2805 RHOL=RHO(IJ) $ VISL=MU(IJ) $ GO TO 2807
2315    2807 NFL=FL(IMJP) $ GO TO(2808,2810,2810,2810,2809,2808,2810 ,2810,
            12808,2808,2810,2810,2808)NFL
2340    2808 RHOTL=RHO(IMJP) $ VISTL=MU(IMJP) $ GO TO 2815
2346    2809 IF(U(IMJP).EQ.0)GO TO 2813
2351         RHOTL=2.*RHO(IMJP)-RHO(IJP) $ VISTL=2.*MU(IMJP)-MU(IJP)
2363         GO TO 2815
2364    2810 RHOTL=RHO(IJP) $ VISTL=MU(IJP) $ GO TO 2815
2372    2813 RHOTL=2.*RHO(IMJP)-RHO(IMJ) $ VISTL=2.*MU(IMJP)-MU(IMJ)
2406         GO TO 2815
2406    2815 CONTINUE
2406         NFL=FL(IPJ) $ GO TO(2818,2817,2817,2817,2817,2818,2818,2818,
            12817,2818,2818,2818,2817)NFL
2431    2817 RHOR=RHO(IJ) $ VISR=MU(IJ) $ GO TO 2820
2437    2818 RHOR=RHO(IPJ) $VISR=MU(IPJ) $ GO TO 2820
2445    2820 NFL=FL(IPJP) $ GO TO(2822,2821,2821,2821,2821,2821,2822,2822,
            12822,2821,2822,2822,2822)NFL
2470    2821 RHOTR=RHO(IJP) $ VISTR=MU(IJP) $ GO TO 2825
```

```
2476   2822 RHOTR=RHO(IPJP) $ VISTR=MU(IPJP) $ GO TO 2825                RICE8
2504   2825 CONTINUE                                                     RICE8
2504        IF(FL(IJ).NE.1)GO TO 2828                                    RICE8
2510        RV2C=.25*RHO(IJ)*(V(IJ)+V(IJM))**2                           RICE8
2516        SRRC=MU(IJ)*(2.*DDY*(V(IJ)-V(IJM))+ALA*(DDY/RJ(J)*(RJ2(J)*   RICE8
           1V(IJ)-RJ2(J-1)*V(IJM))+DDX*(U(IJ)-U(IMJ))))$ GO TO 2838      RICE8
2547   2828 IF(FL(IJ).EQ.6)GO TO 2830                                    RICE8
2553        IF(FL(IJ).EQ.7)GO TO 2830                                    RICE8
2556        IF(FL(IJ).EQ.10)GO TO 2833                                   RICE8
2561        RV2C=0.                                                      RICE8
2561        SRRC=2.*MU(IJ)*((2.+ALA)*DDY*V(IJ)-ALA*DDX*U(IMJ))$GO TO 2838 RICE8
2600   2830 SRRC=0.                                                      RICE8
2601        RV2C=RHO(IJ)*V(IJ)**2                                        RICE8
2605        GO TO 2838                                                   RICE8
2606   2833 SRRC=2.*MU(IJ)*((2.+ALA)*DDY*V(IJ)+ALA*DDX*U(IJ))            RICE8
2620        RV2C=0.                                                      RICE8
2621   2838 IF(FL(IJP).EQ.1)GO TO 2840                                   RICE8
2625        IF(FL(IJP).EQ.8)GO TO 2843                                   RICE8
2630        IF(FL(IJP).EQ.9)GO TO 2843                                   RICE8
2633        IF(FL(IJP).EQ.12)GO TO 2845                                  RICE8
2636        RV2T=0.                                                      RICE8
2636        SRRT=-2.*MU(IJP)*((2.+ALA)*DDY*V(IJ)-ALA*DDX*U(IJP))$GOTO2847 RICE8
2657   2840 SRRT=MU(IJP)*(2.*DDY*(V(IJP)-V(IJ))+ALA*(DDY/RJ(J+1)*        RICE8
           1(RJ2(J+1)*V(IJP)-RJ2(J)*V(IJ))+DDX*(U(IJP)-U(IMJP))))        RICE8
2703        RV2T=.25*RHO(IJP)*(V(IJ)+V(IJP))**2                          RICE8
2711        GO TO 2847                                                   RICE8
2712   2843 SRRT=0.                                                      RICE8
2713        RV2T=RHO(IJP)*V(IJ)**2                                       RICE8
2720        GO TO 2847                                                   RICE8
2720   2845 SRRT=2.*MU(IJP)*(-(2.+ALA)*DDY*V(IJ)-ALA*DDX*U(IMJP))        RICE8
2735        RV2T=0.                                                      RICE8
2736   2847 IF(FL(IMJP).EQ. 6)GO TO 2848                                 RICE8
2742        IF(FL(IMJP).EQ. 7)GO TO 2848                                 RICE8
2745        IF(FL(IMJP).EQ.10)GO TO 2848                                 RICE8
2750        IF(FL(IMJP).EQ.11)GO TO 2848                                 RICE8
2753        IF(FL(IMJ).EQ. 8)GO TO 2850                                  RICE8
2756        IF(FL(IMJ).EQ. 9)GO TO 2850                                  RICE8
2761        IF(FL(IMJ).EQ.12)GO TO 2850                                  RICE8
2764        IF(FL(IMJ).EQ.13)GO TO 2850 $ GO TO 2849                     RICE8
2770   2848 IF(FL(IMJ).EQ. 2)GO TO 2852                                  RICE8
2774        IF(FL(IMJ).EQ. 3)GO TO 2853                                  RICE8
2777   2849 IF(FL(IMJ).EQ.5.AND.U(IMJ).NE.0.)VTL=2.*V(IMJ)-V(IJ)         RICE8
3016        VTL=V(IMJ)                                                   RICE8
3021        GO TO 2855                                                   RICE8
3022   2850 IF(FL(IMJP).EQ.2)GO TO 2852                                  RICE8
3026        IF(FL(IMJP).EQ.3)GO TO 2853 $ GO TO 2849                     RICE8
3031   2852 VTL= V(IJ) $ GO TO 2855                                      RICE8
3035   2853 VTL=-V(IJ) $ GO TO 2855                                      RICE8
3041   2855 IF(FL(IJ).EQ.6)GO TO 2858                                    RICE8
3045        IF(FL(IJ).EQ.10)GO TO 2860                                   RICE8
3050        UL=U(IMJ) $ GO TO 2863                                       RICE8
3053   2858 UL=U(IMJP) $ GO TO 2863                                      RICE8
3057   2860 UL=-U(IMJP) $ GO TO 2863                                     RICE8
3063   2863 IF(FL(IJP).EQ.9)GO TO 2865                                   RICE8
3067        IF(FL(IJP).EQ.13)GO TO 2858                                  RICE8
```

```
3072          UTL=U(IMJP) $ GO TO 2870                                          RICE8
3075     2865 UTL=U(IMJ) $ GO TO 2870                                           RICE8
3101     2868 UTL=-U(IMJ) $ GO TO 2870                                          RICE8
3107     2870 SRZTL=.25*(VISL+MU(IJ)+MU(IJP)+VISTL)*((V(IJ)-VTL)*DDX+           RICE8
             1(UTL-UL)*DDY)                                                     RICE8
3124          RUVTL=(RHO(IJ)+RHO(IJP)+RHOL+RHOTL)*(VTL+V(IJ))*(UTL+UL)*.0625    RICE8
3140          IF(FL(IPJP).EQ. 6)GO TO 2871                                      RICE8
3144          IF(FL(IPJP).EQ. 7)GO TO 2871                                      RICE8
3147          IF(FL(IPJP).EQ.10)GO TO 2871                                      RICE8
3152          IF(FL(IPJP).EQ.11)GO TO 2871                                      RICE8
3155          IF(FL(IPJ).EQ. 8)GO TO 2873                                       RICE8
3161          IF(FL(IPJ).EQ. 9)GO TO 2873                                       RICE8
3164          IF(FL(IPJ).EQ.12)GO TO 2873                                       RICE8
3167          IF(FL(IPJ).EQ.13)GO TO 2873 $GO TO 2872                           RICE8
3172     2871 IF(FL(IPJ).EQ. 2)GO TO 2874                                       RICE8
3176          IF(FL(IPJ).EQ. 3)GO TO 2875                                       RICE8
3201     2872 VTR=V(IPJ) $ GO TO 2878                                           RICE8
3205     2873 IF(FL(IPJP).EQ.2)GO TO 2874                                       RICE8
3211          IF(FL(IPJP).EQ.3)GO TO 2875 $ GO TO 2872                          RICE8
3214     2874 VTR= V(IJ) $ GO TO 2878                                           RICE8
3220     2875 VTR=-V(IJ) $ GO TO 2878                                           RICE8
3224     2878 IF(FL(IJ).EQ.7)GO TO 2880                                         RICE8
3230          IF(FL(IJ).EQ.11)GO TO 2882                                        RICE8
3233          UR=U(IJ) $ GO TO 2884                                             RICE8
3235     2880 UR=U(IJP) $ GO TO 2884                                            RICE8
3241     2882 UR=-U(IJP) $ GO TO 2884                                           RICE8
3245     2884 IF(FL(IJP).EQ.8)GO TO 2886                                        RICE8
3251          IF(FL(IJP).EQ.12)GO TO 2888                                       RICE8
3254          UTR=U(IJP) $ GO TO 2890                                           RICE8
3256     2886 UTR=U(IJ) $ GO TO 2890                                            RICE8
3262     2888 UTR=-U(IJ) $ GO TO 2890                                           RICE8
3275     2890 SRZTR=.25*(MU(IJ)+VISR+VISTR+MU(IJP))*((VTR-V(IJ))*DDX+(UTR-UR)*  RICE8
             1DDY)                                                              RICE8
3310          RUVTR=(RHO(IJ)+RHO(IJP)+RHOR+RHOTR)*(V(IJ)+VTR)*(UTR+UR)*.0625    RICE8
3321          SOOT=(MU(IJP)+MU(IJ))*(V(IJ)/RJ2(J)+.25*ALA*(DDY/RJ2(J)*          RICE8
             1(RJ(J+1)*(V(IJP)+V(IJ))-RJ(J)*(V(IJ)+V(IJM)))+                    RICE8
             1DDX*((UTR+UR)-(UTL+UL))))*ITC                                     RICE8
3352          RHOVT(IJ)=RVOLD+DT*((RJ(J)*(RV2C -SRRC)-RJ(J+1)*(RV2T-SRRT))/     RICE8
             1(RJ2(J)*DY)+(RUVTL-SRZTL-RUV TR+SRZTR)*DDX+(PC-PT)*DDY-           RICE8
             2SOOT/RJ2(J))                                                      RICE8
     C**** INTERMEDIATE  --  RADIAL MOMENTUM CALC.COMPLETE         * * * * * * * * RICE8
3404     2990 CONTINUE                                                          RICE8
3407          IF(FL(IB2J).EQ.4)RHOUT(IB1J)=RHOUT(IBJ)                           RICE8
3416     2995 CONTINUE                                                          RICE8
     C                                                                          RICE8
     C**** END OF TILDE CALC.---ARRAYS OF RHOT,RHOUT AND RHOVT STORED           RICE8
     C                                                                          RICE8
     C**** START OF ITERATION TO OBTAIN FINAL VALUES OF RHOU,RHOV AND RHO       RICE8
3421          IT=0                                                              RICE8
3422     3705 ITF=0                                                             RICE8
3423          IT=IT+1                                                           RICE8
3425          JC=0                                                              RICE8
3426          DO 3800 J=2,JB1                                                   RICE8
3427          JC=JC+IB2                                                         RICE8
3430          DO 3800 I=2,IB1                                                   RICE8
```

```
3432         IJ=I+JC                                                                       RICE8     972
3433         IPJ=IJ+1                                                                      RICE8     973
3434         IMJ=IJ-1                                                                      RICE8     974
3435         IJP=IJ+182                                                                    RICE8     975
3437         IJM=IJ-182                                                                    RICE8     976
3437         IF(FL(IJ ).GT.1.AND.FL(IJ ).LT.6)GO TO 3800                                   RICE8     977
3451         DCSQ=1./CSQ(IJ)                                                               RICE8     978
3454         IF(FL(IJ).EQ.1)GO TO 3710                                                     RICE8     979
3460         NFL=FL(IJ)-5                                                                  RICE8     980
3462         GO TO(3710,3710,3725,3725,3710,3710,3725,3725)NFL                             RICE8     981
3475    3710 NFL=FL(IJP)                                                                   RICE8     982
3500         GO TO(3715,3725,3725,3725,3717,3725,3725,3715,3715,3725,3725,3715,            RICE8     983
            13715)NFL                                                                      RICE8     984
3520    3715 TOP=1.0                                                                       RICE8     985
3521         TP=1.0                                                                        RICE8     986
3523         GO TO 3730                                                                    RICE8     987
3524    3717 IF(V(IJ).EQ.0.)GO TO 3725                                                     RICE8     988
3527         TOP=1.0                                                                       RICE8     989
3530         TP=0.                                                                         RICE8     990
3531         GO TO 3730                                                                    RICE8     991
3531    3725 TOP=0.                                                                        RICE8     992
3532         TP=0.                                                                         RICE8     993
3533    3730 IF(FL(IJ).EQ.1)GO TO 3733                                                     RICE8     994
3537         NFL=FL(IJ)-5                                                                  RICE8     995
3541         GO TO(3733,3738,3738,3733,3733,3738,3738,3733)NFL                             RICE8     996
3554    3733 NFL=FL(IPJ)                                                                   RICE8     997
3557         GO TO(3735,3738,3738,3735,3738,3738,3735,3735,3738,3738,3735,3735,            RICE8     998
            13738)NFL                                                                      RICE8     999
3577    3735 RIGHT=1.0                                                                     RICE8    1000
3601         GO TO 3740                                                                    RICE8    1001
3601    3738 RIGHT=0.                                                                      RICE8    1002
3602    3740 IF(FL(IJ).EQ.1)GO TO 3743                                                     RICE8    1003
3606         NFL=FL(IJ)-5                                                                  RICE8    1004
3610         GO TO(3753,3743,3743,3753,3753.3743,3743,3753)NFL                             RICE8    1005
3623    3743 NFL=FL(IMJ)                                                                   RICE8    1006
3626         GO TO(3745,3753,3753,3753,3748,3745,3753,3753,3745,3745,3753,3753,            RICE8    1007
            13745)NFL                                                                      RICE8    1008
3646    3745 SLEFT=1.0                                                                     RICE8    1009
3647         SLT=1.0                                                                       RICE8    1010
3651         GO TO 3755                                                                    RICE8    1011
3652    3748 IF(U(IMJ).EQ.0.)GO TO 3753                                                    RICE8    1012
3655         SLEFT=1.0                                                                     RICE8    1013
3656         SLT=0.0                                                                       RICE8    1014
3657         GO TO 3755                                                                    RICE8    1015
3657    3753 SLEFT=0.0                                                                     RICE8    1016
3660         SLT=0.0                                                                       RICE8    1017
3661    3755 IF(FL(IJ).EQ.1)GO TO 3758                                                     RICE8    1018
3665         NFL=FL(IJ)-5                                                                  RICE8    1019
3667         GO TO(3763,3763,3758,3758,3763,3763,3758,3758)NFL                             RICE8    1020
3702    3758 IF(FL(IJM).EQ.1)GO TO 3760                                                    RICE8    1021
3706         IF(FL(IJM).LT.6)GO TO 3763                                                    RICE8    1022
3711         NFL=FL(IJM)-5                                                                 RICE8    1023
3714         GO TO(3760,3760,3763,3763,3760,3760,3763,3763)NFL                             RICE8    1024
3727    3760 BOTTOM=1.0                                                                    RICE8    1025
3731         GO TO 3765                                                                    RICE8    1026
```

```
3731   3763 BOTTOM=0.0                                                        RICE8
3732   3765 D=RHO(IJ )-RHOT(IJ)+THETA*DT*((TOP*RHOV(IJ)*RJ2(J)-BOTTOM*RHOV(IJM RICE8
           1)*RJ2(J-1))/(RJ(J)*DY)+(RIGHT*RHOU(IJ)-SLEFT*RHOU(IMJ))*DDX)      RICE8
3764            IF(ABS(D).LE.ERRD)GO TO 3800                                  RICE8
3774            ITF=1                                                         RICE8
3774            DPB=-OMEGA*D/(DCSQ+PFAC)                                      RICE8
4000            PB(IJ)=PB(IJ)+DPB                                             RICE8
4004            RHO(IJ)=RHO(IJ)+DPB*DCSQ                                      RICE8
400E            RHOU(IJ)=RHOU(IJ)+RIGHT*UCON*DPB                              RICE8
4014            RHOU(IMJ)=RHOU(IMJ)-SLT*UCON*DPB                              RICE8
4020            RHOV(IJ)=RHOV(IJ)+TP*VCON*DPB                                 RICE8
4023            RHOV(IJM)=RHOV(IJM)-BOTTOM*VCON*DPB                           RICE8
4027   3800 CONTINUE                                                          RICE8
4034            IF(ITF.EQ.0) GO TO 3802                                       RICE8
4035            IF(IT.LT.ITMAX)GO TO 3705                                     RICE8
      C**** PRESSURE ITERATION DETERMINES FINAL MASS AND MOMENTUM VALUES * * * RICE8
      C**** PARTIALLY UPDATED PRESSURE---NEEDS TO BE UPDATED AT END OF CYCLE  RICE8
      C                                                                       RICE8
4040   3802 CONTINUE                                                          RICE8
4040            RETURN                                                        RICE8
4041            END                                                           RICE8
      C                                                                       RICE8
      C                                                                       RICE8
      C**** END OF THE SUBROUTINE TILDE ************************************** RICE8
      C                                                                       RICE8
      C                                                                       RICE8
```

```
          SUBROUTINE ENERGY                                               RICE8
  1       COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,    RCOM2
         1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC,ALA,BIE,THETA, RCOM2
         2PHI,OMEGA,GAMMA,DXOU,ASQ                                        RCOM2
  1       COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SECREQ,DDX,DDY      RCOM2
         1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT                           RCOM2
  1       COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1, RCOM2
         1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2, RCOM2
         2SIEL1,SIEL2                                                     RCOM2
  1       REAL MUO,MINT1,MINT2,MINL1,MINL2                                RCOM2
  1       COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                       RCOM2
         1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),     RCOM2
         2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)          RCOM2
  1       COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20) RCOM2
         1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),             RCOM2
         2SIEBO(20),FARO(7,20),FABO(7,20)                                 RCOM2
  1       INTEGER AM,BM                                                   RCOM2
  1       REAL MINRO,MINBO                                                RCOM2
    C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP) RCOM2
    C**** AND THE *READ FEATURE OF (UPDATE)                               RCOM2
    C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/      RCOM2
    C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC RCOM2
    C                                                                     RCOM2
  1          LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32                   RCOM2
         1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                           RCOM2
         2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62               RCOM2
         3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(                RCOM2
         4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                            RCOM2
         5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                         RCOM2
  1       DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                     RCOM2
         1 62, 32),DRHOB( 62, 7),STREB( 62)                               RCOM2
  1       EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),      RCOM2
         1(TFAB,DRHOB)                                                    RCOM2
  1       INTEGER FL,CYCLE                                                RCOM2
  1       REAL MU                                                         RCOM2
  1       DO 4503 J=2,JB2                                                 RICE8
  3       IBJ=IB2*(J-1)+IB                                                RICE8
  7       IB1J=IBJ+1                                                      RICE8
  7       IB2J=IBJ+2                                                      RICE8
 11       IB1JM=IB1J-IB2                                                  RICE8
 12       IF(FL(IB2J).NE.4)GO TO 4503                                     RICE8
    C                                                                     RICE8
    C**** RHO AND P    ARE SET CONTINUOUS(= COL.IB) AT OUTFLOW BOUNDARY   RICE8
    C                                                                     RICE8
 16       RHO(IB2J)=RHO(IB1J)=RHO(IBJ)                                    RICE8
 24       P(IB2J)=P(IB1J)=P(IBJ)                                          RICE8
 33  4503 CONTINUE                                                        RICE8
 36       JC=0                                                            RICE8
 37       DO 4850 J=2,JB1                                                 RICE8
 40       JC=JC+IB2                                                       RICE8
 41       DO 4850 I=2,IB1                                                 RICE8
 43       IJ=I+JC                                                         RICE8
    C                                                                     RICE8
    C**** START OF RE CALC..BYPASS ALL CALC.IF FL(IJ)=2,3,4 OR 5          RICE8
    C                                                                     RICE8
```

```
 44        IF(FL(IJ).GT.1.AND.FL(IJ).LT.6)GO TO 4850                          RICE6    1075
 56        IPJ=IJ+1                                                           RICE8    1076
 56        IMJ=IJ-1                                                           RICE8    1077
 57        IJP=IJ+IB2                                                         RICE8    1078
 61        IJM=IJ-IB2                                                         RICE8    1079
 61        IMJM=IJM-1                                                         RICE8    1080
 62        IPJM=IJM+1                                                         RICE8    1081
 63        IMJP=IJP-1                                                         RICE8    1082
 65        IPJP=IJP+1                                                         RICE8    1083
      C                                                                       RICE8    1084
      C**** START OF ENERGY-DENSITY FLUX CALC.                                RICE8    1085
      C                                                                       RICE8    1086
 65        IF(FL(IJ).EQ.1)GO TO 4510 $ NFL=FL(IJ)-5                           RICE8    1087
 73        GO TO(4510,4507,4507,4510,4510,4507,4507,4510)NFL                  RICE8    1088
106   4507 TRER=0. $ GO TO 4605                                               RICE8    1089
110   4510 NFL=FL(IPJ)                                                        RICE8    1090
113        GO TO(4515,4512,4512,4515,4512,4512,4515,4515,4512,4512,4515,4515, RICE8    1091
          14512)NFL                                                           RICE8    1092
133   4512 TRER=0. $ GO TO 4605                                               RICE8    1093
137   4515 UR=U(IJ)                                                           RICE8    1094
141        DIR=SIE(IPJ)-SIE(IJ)                                               RICE8    1095
144        PBR=(PB(IPJ)+PB(IJ))*.5                                            RICE8    1096
151        RMU=(MU(IPJ)+MU(IJ))*.5 $ RER=(RE(IPJ)+RE(IJ))*.5                  RICE8    1097
163        IF(FL(IPJ).EQ.1)GO TO 4521                                         RICE8    1098
167        IF(FL(IPJ).EQ.4)GO TO 4519                                         RICE8    1099
172        IF(FL(IPJ).EQ.7)GO TO 4519                                         RICE8    1100
175        IF(FL(IPJ).EQ.8)GO TO 4519                                         RICE8    [illegible]
200        URR=-UR $ GO TO 4522                                               RICE8    [illegible]
202   4519 URR=UR $ GO TO 4522                                                RICE8    1103
204   4521 URR=U(IPJ) $ GO TO 4532                                            RICE8    1104
210   4522 IF(FL(IPJ).EQ.7)GO TO 4532                                         RICE8    1105
214        IF(FL(IPJ).EQ.11)GO TO 4532                                        RICE8    1106
217        IF(FL(IPJP).EQ.1)GO TO 4532                                        RICE8    1107
222        IF(FL(IPJP).EQ.4)GO TO 4527                                        RICE8    1108
225        IF(FL(IPJ).EQ.12)GO TO 4529                                        RICE8    1109
231   4527 VZZTR=V(IPJM) $ VRZTR=V(IJ) $ GO TO 4535                           RICE8    1110
237   4529 VZZTR=-V(IPJM) $ VRZTR=-V(IJ) $ GO TO 4535                         RICE8    1111
245   4532 VZZTR=VRZTR=V(IPJ)                                                 RICE8    1112
251   4535 IF(FL(IPJ).EQ.1)GO TO 4541                                         RICE8    1113
255        IF(FL(IPJ).EQ.8)GO TO 4541                                         RICE8    1114
260        IF(FL(IPJ).EQ.12)GO TO 4541                                        RICE8    1115
263        IF(FL(IPJ).EQ.4)GO TO 4543                                         RICE8    1116
266   4537 IF(FL(IPJM).EQ.1)GO TO 4541                                        RICE8    1117
272        IF(FL(IPJ).EQ.11)GO TO 4545                                        RICE8    1118
275   4539 VRZBR=V(IJM) $ VZZBR=V(IPJ) $ GO TO 4550                           RICE8    1119
303   4543 IF(FL(IPJM).EQ.4)GO TO 4539                                        RICE8    1120
307   4541 VZZBR=V(IPJM) $ VRZBR=V(IPJM) $ GO TO 4550                         RICE8    1121
315   4545 VRZBR=-V(IJM) $ VZZBR=-V(IPJ) $ GO TO 4550                         RICE8    1122
323   4550 NFL=FL(IJM) $ GO TO(4554,4552,4552,4554,4554,4554,4552,4552,       RICE8    1123
          14554,4554,4552,4552,4554)NFL                                       RICE8    1124
346   4552 IF(FL(IPJM).EQ.1)GO TO 4554                                        RICE8    1125
352        IF(FL(IJM).EQ.2)GO TO 4556                                         RICE8    1126
355        IF(FL(IJM).EQ.7)GO TO 4556                                         RICE8    1127
360        IF(FL(IJM).EQ.8)GO TO 4556                                         RICE8    1128
363        GO TO 4558                                                         RICE8    1129
```

```
364    4554 UBR=U(IJM) $ GO TO 4560                                         RICE8
370    4556 UBR=UR $ GO TO 4560                                             RICE8
372    4558 UBR=-UR $ GO TO 4560                                            RICE8
374    4560 NFL=FL(IJP) $ GO TO(4564,4562,4562,4564,4564,4564,4562,4562,    RICE8
           14564,4564,4562,4562,4564)NFL                                    RICE8
417    4562 IF(FL(IPJP).EQ.1)GO TO 4564                                     RICE8
423         IF(FL(IJP).EQ.2)GO TO 4566                                      RICE8
426         IF(FL(IJP).EQ.7)GO TO 4566                                      RICE8
431         IF(FL(IJP).EQ.8)GO TO 4566                                      RICE8
434         GO TO 4568                                                      RICE8
435    4564 UTR=U(IJP) $ GO TO 4570                                         RICE8
441    4566 UTR=UR $ GO TO 4570                                             RICE8
443    4568 UTR=-UR $ GO TO 4570                                            RICE8
445    4570 IF(FL(IJ).EQ. 6)GO TO 4574                                      RICE8
451         IF(FL(IJ).EQ.10)GO TO 4574                                      RICE8
454    4572 VZZB=V(IJM) $ VRZB=V(IJM) $ GO TO 4580                          RICE8
462    4574 IF(FL(IJM).EQ.1)GO TO 4572                                      RICE8
466         IF(FL(IJ).EQ.6)GO TO 4578                                       RICE8
471         VRZB=-V(IPJM) $ VZZB=-V(IJ) $ GO TO 4580                        RICE8
477    4578 VRZB=V(IPJM) $ VZZB=V(IJ)                                       RICE8
505    4580 IF(FL(IJ).EQ.1)GO TO 4582                                       RICE8
511         IF(FL(IMJ).EQ.1)GO TO 4582                                      RICE8
514         IF(FL(IJ).EQ. 6)GO TO 4586                                      RICE8
517         IF(FL(IJ).EQ. 9)GO TO 4586                                      RICE8
522         GO TO 4584                                                      RICE8
523    4582 UL=U(IMJ) $ GO TO 4590                                          RICE8
527    4584 UL=-U(IJ) $ GO TO 4590                                          RICE8
533    4586 UL=U(IJ) $ GO TO 4590                                           RICE8
537    4590 IF   (FL(IJ).EQ. 1)GO TO 4592                                   RICE8
543         IF(FL(IJ).EQ. 6)GO TO 4592                                      RICE8
546         IF(FL(IJ).EQ.10)GO TO 4592                                      RICE8
551         IF(FL(IJP).EQ.1)GO TO 4592                                      RICE8
554         IF(FL(IJ).EQ.13)GO TO 4594                                      RICE8
557         VRZT=V(IPJ) $ VZZT=V(IJM) $ GO TO 4600                          RICE8
565    4592 VRZT=V(IJ) $ VZZT=V(IJ) $ GO TO 4600                            RICE8
573    4594 VRZT=-V(IPJ) $ VZZT=-V(IJM) $ GO TO 4600                        RICE8
601    4600 CONTINUE                                                        RICE8
607         SZZR=RMU*.5*((2.+ALA)*DDX*(URR-UL)+ALA*DDY/RJ(J)*(RJ2(J)*(VZZTR+ RICE8
           1VZZT)-RJ2(J-1)*(VZZBR+VZZB)))                                   RICE8
627         SRZR=RMU*.5*(DDX*((VRZTR+VRZBR)-(VRZT+VRZB))+DDY*(UTR-UBR))     RICE8
637         VR=.25*(VRZBR+VRZTR+VRZT+VRZB)                                  RICE8
       C**** ENERGY-DENSITY FLUX ACROSS RIGHT BOUNDARY=TRER        * * * * * * * * RICE8
       C                                                                    RICE8
643         TRER=UR*(RER+PBR-SZZR)-SRZR*VR-BIE*RMU*DDX*DIR                  RICE8
654    4605 IF(FL(IJ).EQ.1)GO TO 4607 $ NFL=FL(IJ)-5                        RICE8
662         GO TO(4607,4607,4609,4609,4607,4607,4609,4609)NFL               RICE8
675    4607 NFL=FL(IJP) $ GO TO(4615,4610,4610,4610,4612,4610,4610,4615,    RICE8
           14615,4610,4610,4615,4615)NFL                                    RICE8
720    4609 CONTINUE                                                        RICE8
720    4610 TRET=0. $ GO TO 4705                                            RICE8
722    4612 IF(V(IJ).EQ.0.)GO TO 4610                                       RICE8
726    4614 VT=V(IJ)                                                        RICE8
730         DIT=2.*(SIE(IJP)-SIE(IJ))                                       RICE8
735         PT=PB(IJP) $ TMU=MU(IJP) $ RET=RE(IJP) $ GO TO 4618             RICE8
746    4615 VT=V(IJ)                                                        RICE8
```

```
 750        DIT=SIE(IJP)-SIE(IJ)                                        RICE8
 753        PT=.5*(PB(IJP)+PB(IJ))                                      RICE8
 760        TMU=.5*(MU(IJP)+MU(IJ)) $ RET=.5*(RE(IJP)+RE(IJ))           RICE8
 772   4618 IF(FL(IJP).EQ.1)GO TO 4620                                  RICE8
 776        IF(FL(IJP).EQ.5)GO TO 4624                                  RICE8
1001        IF(FL(IJP).EQ.8)GO TO 4624                                  RICE8
1004        IF(FL(IJP).EQ.9)GO TO 4624                                  RICE8
1007        VTT=-VT $ GO TO 4626                                        RICE8
1011   4620 VTT=V(IJP) $ GO TO 4626                                     RICE8
1015   4624 VTT=VT                                                      RICE8
1017   4626 IF(FL(IJP).EQ. 8)GO TO 4628                                 RICE8
1023        IF(FL(IJP).EQ.12)GO TO 4628                                 RICE8
1026   4627 URZTR=U(IJP)                                                RICE8
1031        URRTR=U(IJP) $ GO TO 4635                                   RICE8
1034   4628 IF(FL(IPJP).EQ.1)GO TO 4627                                 RICE8
1040        IF(FL(IJP).EQ.8)GO TO 4632                                  RICE8
1043        URZTR=-U(IJ) $ URRTR=-U(IMJP) $ GO TO 4635                  RICE8
1051   4632 URZTR=U(IJ) $ URRTR=U(IMJP)                                 RICE8
1057   4635 IF(FL(IJP).EQ. 9)GO TO 4637                                 RICE8
1063        IF(FL(IJP).EQ..3)GO TO 4637                                 RICE8
1066   4636 URRTL=U(IMJP) $ URZTL=U(IMJP) $ GO TO 4645                  RICE8
1074   4637 IF(FL(IMJP).EQ.1)GO TO 4636                                 RICE8
1100        IF(FL(IJP).EQ.9)GO TO 4642                                  RICE8
1103        URZTL=-U(IMJ) $ URRTL=-U(IJP) $ GO TO 4645                  RICE8
1111   4642 URZTL=U(IMJ) $ URRTL=U(IJP)                                 RICE8
1117   4645 NFL=FL(IPJ) $ GO TO(4648,4649,4649,4649,4647,4648,4648,4649, RICE8
          14649,4648,4648,4649,4649)NFL                                 RICE8
1142   4647 IF(FL(IPJP).EQ.2)GO TO 4655                                 RICE8
1146        IF(FL(IPJP).EQ.3)GO TO 4653                                 RICE8
1151   4648 VTR=V(IPJ) $ GO TO 4656                                     RICE8
1155   4649 IF(FL(IPJP).EQ.1)GO TO 4648                                 RICE8
1161        IF(FL(IPJP).EQ.4)GO TO 4655                                 RICE8
1164        IF(FL(IPJ).EQ. 2)GO TO 4655                                 RICE8
1167        IF(FL(IPJ).EQ. 8)GO TO 4655                                 RICE8
1172        IF(FL(IPJ).EQ. 9)GO TO 4655                                 RICE8
1175        IF(FL(IPJ).EQ.4)GO TO 4655                                  RICE8
1200   4653 VTR=-VT $ GO TO 4656                                        RICE8
1202   4655 VTR=VT $ GO TO 4656                                         RICE8
1204   4656 NFL=FL(IMJ) $ GO TO(4659,4661,4661,4659,4657,4659,4659,4661, RICE8
          14661,4659,4659,4661,4661)NFL                                 RICE8
1227   4657 IF(U(IMJ).NE.0.)GO TO 4659                                  RICE8
1232   4658 IF(FL(IMJP).EQ.2)GO TO 4667                                 RICE8
1236        IF(FL(IMJP).EQ.3)GO TO 4669                                 RICE8
1241   4659 VTL=V(IMJ) $ GO TO 4670                                     RICE8
1245   4661 IF(FL(IMJP).EQ.1)GO TO 4659                                 RICE8
1251        IF(FL(IMJ).EQ. 3)GO TO 4669                                 RICE8
1254        IF(FL(IMJ).EQ.12)GO TO 4669                                 RICE8
1257        IF(FL(IMJ).EQ.13)GO TO 4669                                 RICE8
1262   4667 VTL=VT $ GO TO 4670                                         RICE8
1264   4669 VTL=-VT $ GO TO 4670                                        RICE8
1266   4670 IF(FL(IJ).EQ. 1)GO TO 4672                                  RICE8
1272        IF(FL(IJ).EQ. 6)GO TO 4672                                  RICE8
1275        IF(FL(IJ).EQ.10)GO TO 4672                                  RICE8
1300        IF(FL(IPJ).EQ.1)GO TO 4672                                  RICE8
1303        IF(FL(IJ).EQ.7)GO TO 4674                                   RICE8
```

```
1306        URZR=-U(IJP) $ URRR=-U(IMJ) $ GO TO 4675
1314   4672 URZR=U(IJ) $ URRR=U(IJ) $ GO TO 4675
1322   4674 URZR=U(IJP) $ URRR=U(IMJ)
1330   4675 IF(FL(IJ).EQ. 1)GO TO 4677
1334        IF(FL(IJ).EQ. 7)GO TO 4677
1337        IF(FL(IJ).EQ.11)GO TO 4677
1342        IF(FL(IMJ).EQ.1)GO TO 4677
1345        IF(FL(IJ).EQ.6)GO TO 4680
1350        URZL=-U(IMJP) $ URRL=-U(IJ) $ GO TO 4682
1356   4677 URZL=U(IMJ) $ URRL=U(IMJ) $ GO TO 4682
1364   4680 URZL=U(IMJP) $ URRL=U(IJ)
1372   4682 IF(FL(IJ).EQ.1) GO TO 4684
1376        IF(FL(IJM).EQ.1)GO TO 4684
1401        IF(FL(IJ).EQ.6)GO TO 4690
1404        IF(FL(IJ).EQ.7)GO TO 4690
1407        VB=-V(IJ) $ GO TO 4695
1412   4684 VB=V(IJM) $ GO TO 4695
1416   4690 VB=V(IJ)
1424   4695 SRRT=TMU*(DDY*(VTT-VB)+ALA*.5*(DDY/RJ2(J)*(RJ(J+1)*(VTT+VT)-
           1RJ(J)*(VT+VB))+DDX*(URRTR+URRR-URRTL-URRL)))
1450        SRZT=.5*TMU*(DDX*(VTR-VTL)+DDY*(URZTR+URZTL-URZR-URZL))
1461        UT=.25*(URZTR+URZTL+URZL+URZR)
      C**** ENERGY-DENSITY FLUX ACROSS TOP BOUNDARY=TRET        ********
      C
1465        TRET=VT*(RET+PT-SRRT)-SRZT*UT-BIE*TMU*DDY*DIT
1476   4705 IF(FL(IJ).EQ.1)GO TO 4707
1502        NFL=FL(IJ)-5
1504        GO TO(4709,4707,4707,4709,4709,4707,4707,4709)NFL
1517   4707 NFL=FL(IMJ) $ GO TO(4755,4709,4709,4709,4712,4755,4709,4709,
           14755,4755,4709,4709,4755)NFL
1542   4709 TREL=0. $ GO TO 4755
1544   4712 IF(U(IMJ).EQ.0.)GO TO 4709
1547        UL=U(IMJ)
1551        DIL=2.*(SIE(IJ)-SIE(IMJ))
1555        PL=PB(IMJ)
1560        XMUL=MU(IMJ) $ REL=RE(IMJ) $ ULL=U(IMJ)
1566        VTL=V(IMJ) $ VBL=V(IMJM)
1572        IF(FL(IJP).EQ. 2)GO TO 4716
1576        IF(FL(IJP).EQ. 6)GO TO 4716
1601        IF(FL(IJP).EQ. 9)GO TO 4716
1604        IF(FL(IJP).EQ. 3)GO TO 4718
1607        IF(FL(IJP).EQ.10)GO TO 4718
1612        IF(FL(IJP).EQ.13)GO TO 4718
1615        UTL=U(IMJP) $ GO TO 4720
1620   4716 UTL=UL $ GO TO 4720
1622   4718 UTL=-UL
1624   4720 IF(FL(IJM).EQ. 2)GO TO 4722
1630        IF(FL(IJM).EQ. 6)GO TO 4722
1633        IF(FL(IJM).EQ. 9)GO TO 4722
1636        IF(FL(IJM).EQ. 3)GO TO 4724
1641        IF(FL(IJM).EQ.10)GO TO 4724
1644        IF(FL(IJM).EQ.13)GO TO 4724
1647        UBL=U(IMJM) $ GO TO 4730
1652   4722 UBL=UL $ GO TO 4730
1654   4724 UBL=-UL
```

```
1656   4730 IF(FL(IJ).EQ.1)GO TO 4732
1662        IF(FL(IJ).EQ.7)GO TO 4734
1665        IF(FL(IJ).EQ.8)GO TO 4734
1670        UR=-U(IMJ) $ GO TO 4736
1673   4732 UR=U(IJ) $ GO TO 4736
1677   4734 UR=U(IMJ)
1702   4736 IF(FL(IJ).EQ. 7)GO TO 4738
1706        IF(FL(IJ).EQ.11)GO TO 4740
1711        VZZB=V(IJM) $ VRZB=V(IJM) $ GO TO 4742
1716   4738 VZZB=V(IJ) $ VRZB=V(IMJM) $ GO TO 4742
1724   4740 VZZB=-V(IJ) $ VRZB=-V(IMJM)
1732   4742 IF(FL(IJ).EQ. 8)GO TO 4744
1736        IF(FL(IJ).EQ.12)GO TO 4746
1741        VZZT=V(IJ) $ VRZT=V(IJ) $ GO TO 4750
1745   4744 VZZT=V(IJM) $ VRZT=V(IMJ) $ GO TO 4750
1753   4746 VZZT=-V(IJM) $ VRZT=-V(IMJ)
1764   4750 SZZL=XMUL*.5*((2.+ALA)*DDX*(UR-ULL)+ALA*DDY/RJ(J)*(RJ2(J)*
           1(VZZT+VTL)-RJ2(J-1)*(VZZB+VBL)))
2006        SRZL=.5*XMUL*(DDX*(VRZT+VRZB-VTL-VBL)+DDY*(UTL-UBL))
2017        VL=.25*(VRZT+VRZB+VTL+VBL)
     C**** ENERGY-DENSITY FLUX ACROSS LEFT BOUNDARY=TREL        * * * * * * * *
     C
2022        TREL=UL*(REL+PL-SZZL)-SRZL*VL-BIE*XMUL*DIL*DDX
2034   4755 IF(FL(IJM).EQ.2)GO TO 4757
2040        IF(FL(IJM).EQ.3)GO TO 4757
2043        IF(FL(IJM).EQ.5)GO TO 4757
2046        GO TO 4760
2046   4757 STREB(I)=0.
2051   4760 CONTINUE
     C****
     C**** RE(IJ) CALC.COMPLETE
     C
2054   4805 RE(IJ)= RE(IJ)+DT*(DDY/RJ(J)*(RJ2(J-1)*STREB(I)-RJ2(J)*TRET)+
           1DDX*(TREL-TRER))
2072        STREB(I)=TRET
     C
     C**** SETS TOP FLUX QUANTITY INTO BOTTOM AND RIGHT FLUX QUANTITY INTO
     C**** LEFT TO AVOID CALCULATING OVER AGAIN
     C
2075        TREL=TRER
2076   4850 CONTINUE
2103        JC=0
2104        DO 4950 J=2,JB1
2105        JC=JC+IB2
2106        DO 4950 I=2,IB1
2110        IJ=I+JC
2111        IF(FL(IJ).GT.1.AND.FL(IJ).LT.6)GO TO 4950
2124        IMJ=IJ-1
2125        IJM=IJ-IB2
     C
     C**** OBTAINS INTERNAL ENERGY FROM TOTAL ENERGY-DENSITY
     C
2126        EN=RE(IJ)/RHO(IJ)
2132        IF(FL(IJ).EQ.1)GO TO 4905
2135        NFL=FL(IJ)-5 $ GO TO(4908,4910,4910,4908,4908,4910,4910,4908)NFL
```

```
2153   4905 XKE=.25*(U(IJ)+U(IMJ))**2 $ GO TO 4915                        RICE8
2162   4908 XKE=U(IJ)**2 $ GO TO 4915                                     RICE8
2166   4910 XKE=U(IMJ)**2                                                 RICE8
2171   4915 IF(FL(IJ).EQ.1)GO TO 4917                                     RICE8
2175        NFL=FL(IJ)-5 $ GO TO(4919,4919,4921,4921,4919,4919,4921,4921)NFL  RICE8
2212   4917 YKE=.25*(V(IJ)+V(IJM))**2 $ GO TO 4925                        RICE8
2221   4919 YKE=V(IJ)**2 $ GO TO 4925                                     RICE8
2225   4921 YKE=V(IJM)**2                                                 RICE8
2230   4925 SIE(IJ)=EN-.5*(XKE+YKE)                                       RICE8
2236   4950 CONTINUE                                                      RICE8
2243        DO 4953 J=2,JB2                                               RICE8
2245        IBJ=IB2*(J-1)+IB                                              RICE8
2251        IB1J=IBJ+1                                                    RICE8
2251        IB2J=IBJ+2                                                    RICE8
2253        IB1JM=IB1J-IB2                                                RICE8
2254        IF(FL(IB2J).NE.4)GO TO 4953                                   RICE8
      C                                                                   RICE8
      C**** SETS SIE AND RE CONST.AT OUTFLOW BOUNDARY                     RICE8
      C**** ENERGY CHANGE TO RE(IB1,J) DOES NOT CONTAIN KINETIC ENERGY AS SPEC RICE8
      C**** CALC. ONLY CHANGES INTERNAL ENERGY                            RICE8
      C                                                                   RICE8
2260        SIE(IB2J)=SIE(IB1J)=SIE(IBJ)                                  RICE8
2267        RE(IB1J)=RHO(IB1J)*(SIE(IB1J)+.125*((U(IB1J)+U(IBJ))**2+      RICE8
           1 (V(IB1J)+V(IB1JM))**2))                                      RICE8
2304        RE(IB2J)=RE(IB1J)                                             RICE8
2307   4953 CONTINUE                                                      RICE8
2312        RETURN $ END                                                  RICE8
      C                                                                   RICE8
      C                                                                   RICE8
      C**** END OF THE SUBROUTINE ENERGY ................................ RICE8
      C                                                                   RICE8
      C                                                                   RICE8
```

```
          SUBROUTINE SPEC                                                RICE8   1382
    1     COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,   RCOM2      2
         1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC,ALA,BIE,THETA,RCOM2      3
         2PHI,OMEGA,GAMMA,DXOU,ASQ                                       RCOM2      4
    1     COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SECREQ,DDX,DDY     RCOM2      5
         1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT                          RCOM2      6
    1     COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1,RCOM2      7
         1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,RCOM2      8
         2SIEL1,SIEL2                                                    RCOM2      9
    1     REAL MUO,MINT1,MINT2,MINL1,MINL2                               RCOM2     10
    1     COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                      RCOM2     11
         1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),    RCOM2     12
         2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)         RCOM2     13
    1     COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20) RCOM2 14
         1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),            RCOM2     15
         2SIEBO(20),FARO(7,20),FABO(7,20)                                RCOM2     16
    1     INTEGER AM,BM                                                  RCOM2     17
    1     REAL MINRO,MINBO                                               RCOM2     18
C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP)  RCOM2     19
C**** AND THE *READ FEATURE OF (UPDATE)                                   RCOM2     20
C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/          RCOM2     21
C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC    RCOM2     22
C                                                                        RCOM2     23
    1        LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32                  RCOM2     24
         1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                          RCOM2     25
         2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62              RCOM2     26
         3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(               RCOM2     27
         4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                           RCOM2     28
         5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                        RCOM2     29
    1     DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                    RCOM2     30
         1 62, 32),DRHOB( 62, 7),STREB( 62)                              RCOM2     31
    1     EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),     RCOM2     32
         1(TFAB,DRHOB)                                                   RCOM2     33
    1     INTEGER FL,CYCLE                                               RCOM2     34
    1     REAL MU                                                        RCOM2     35
    1     DIMENSION DOMEGA(3),RM(7)                                      RICE8   1384
    1     DIMENSION DRHO(7),DRHOR(7),DRHOL(7),DRHOT(7),DRHOP(7)          RICE8   1385
         1,RHO1(7),RHO2(7),FAL(7),FAR(7),FAT(7)                          RICE8   1386
    1     REAL KF,KB                                                     RICE8   1387
    1     DIMENSION SUMB(100)                                            RICE8   1388
    1     DO 5002 K=1,NSPEC                                              RICE8   1389
    3     IF(SM(K).GT.0.)RM(K)=1./SM(K)                                  RICE8   1390
    7 5002 IF(SM(K).LE.0.)RM(K)=0.                                       RICE8   1391
   16 5000 NSP=NSPEC-1                                                   RICE8   1392
   20     JC=0                                                           RICE8   1393
C                                                                        RICE8   1394
C**** START SPECIES DENSITY-(FA) CALC.                                   RICE8   1395
C***** START OF SPECIES CONVECTION CALCULATION                           RICE8   1396
C                                                                        RICE8   1397
   21     DO 5200 J=2,JB1                                                RICE8   1398
   22     JC=JC+IB2                                                      RICE8   1399
   23     DO 5200 I=2,IB1                                                RICE8   1400
   25     IJ=I+JC                                                        RICE8   1401
   26     IPJ=IJ+1                                                       RICE8   1402
   27     IMJ=IJ-1                                                       RICE8   1403
```

```
 30          IJP=IJ+IB2                                                  RICE8   1404
 32          IJM=IJ-IB2                                                  RICE8   1405
 32          IMJM=IJM-1                                                  RICE8   1406
 33          IPJM=IJM+1                                                  RICE8   1407
 34          IMJP=IJP-1                                                  RICE8   1408
 36          IPJP=IJP+1                                                  RICE8   1409
 37          NFL=FL(IJ) $ GO TO(5005,5200,5200,5200,5200,5010,5020,5020,5010,  RICE8   1410
            15015,5020,5020,5015)NFL                                     RICE8   1411
 62     5005 UL=U(IMJ) $ GO TO 5025                                      RICE8   1412
 66     5010 UL=U(IJ)    $ GO TO 5025                                    RICE8   1413
 72     5015 UL=-U(IJ)   $ GO TO 5025                                    RICE8   1414
 76     5020 DO 5021 K=1,NSP                                             RICE8   1415
103     5021 FAR(K)=0.                                                   RICE8   1416
105          GO TO 5065                                                  RICE8   1417
105     5025 NFL=FL(IPJ) $ GO TO(5030,5020,5020,5045,5020,5020,5035,5035,5020  RICE8   1418
            1,5020,5040,5040,5020)NFL                                    RICE8   1419
130     5030 UR=U(IPJ) $ GO TO 5050                                      RICE8   1420
134     5035 UR=U(IJ)    $ GO TO 5050                                    RICE8   1421
140     5040 UR=-U(IJ)   $ GO TO 5050                                    RICE8   1422
144     5045 DO 5046 K=1,NSP                                             RICE8   1423
161          KZ=IB2*JB2*(K-1)                                            RICE8   1424
163          IJK=IJ+KZ                                                   RICE8   1425
164     5046 FAR(K)=2.*FA(IJK)*U(IJ)                                     RICE8   1426
173          GO TO 5065                                                  RICE8   1427
173     5050 TAUTE=0.125*DX*(UR-UL)                                      RICE8   1428
177          IF(TAUTE.LT.0.) GO TO 5055                                  RICE8   1429
200          TAUR=(1.+XI)*TAUTE $ GO TO 5060                             RICE8   1430
203     5055 TAUR=(1.-XI)*TAUTE                                          RICE8   1431
206     5060 DO 5061 K=1,NSP                                             RICE8   1432
230          KZ=IB2*JB2*(K-1)                                            RICE8   1433
232          IJK=IJ+KZ                                                   RICE8   1434
233          IPJK=IJK+1                                                  RICE8   1435
234     5061 FAR(K)=(FA(IPJK)+FA(IJK)-DT*DDX*(FA(IPJK)-FA(IJK))*U(IJ))*  RICE8   1436
            1U(IJ)-2.*DDX*TAUR*(FA(IPJK)-FA(IJK))                        RICE8   1437
251     5065 IF(FL(IJ).EQ.1)GO TO 5070                                   RICE8   1438
255          NFL=FL(IJ)-5 $ GO TO(5075,5075,5085,5085,5080,5080,5085,5085)NFL  RICE8   1439
272     5070 VB=V(IJM) $ GO TO 5090                                      RICE8   1440
276     5075 VB=V(IJ)    $ GO TO 5090                                    RICE8   1441
302     5080 VB=-V(IJ)   $ GO TO 5090                                    RICE8   1442
306     5085 DO 5086 K=1,NSP                                             RICE8   1443
313     5086 FAT(K)=0.                                                   RICE8   1444
315          GO TO 5140                                                  RICE8   1445
315     5090 NFL=FL(IJP) $ GO TO(5095,5085,5085,5085,5125,5085,5085,5100,5100  RICE8   1446
            1,5085,5085,5105,5105)NFL                                    RICE8   1447
340     5095 VT=V(IJP) $ GO TO 5110                                      RICE8   1448
344     5100 VT=V(IJ)    $ GO TO 5110                                    RICE8   1449
350     5105 VT=-V(IJ)                                                   RICE8   1450
353     5110 TAUTE=.125*DY*(VT*RJ2(J+1)-VB*RJ2(J-1))/RJ2(J)              RICE8   1451
364          IF(TAUTE.LT.0.)GO TO 5115                                   RICE8   1452
366          TAUT=(1.+XI)*TAUTE $ GO TO 5120                             RICE8   1453
371     5115 TAUT=(1.-XI)*TAUTE                                          RICE8   1454
374     5120 DO 5121 K=1,NSP                                             RICE8   1455
421          KZ=IB2*JB2*(K-1)                                            RICE8   1456
423          IJK=IJ+KZ                                                   RICE8   1457
424          IJPK=IJK+IB2                                                RICE8   1458
```

```
 425   5121 FAT(K)=((FA(IJPK)+FA(IJK)-DT*DDY*(FA(IJPK)-FA(IJK))*V(IJ))
          1*V(IJ)-2.*DDY*TAUT*(FA(IJPK)-FA(IJK)))*RJ2(J)
 443        GO TO 5140
 443   5125 IF(V(IJ).EQ.0.) GO TO 5085
 446        TAUTE=.125*DY*(V(IJ)*RJ2(J)-VB*RJ2(J-1))/RJ2(J)
 457        IF(TAUTE.LT.0.) GO TO 5130
 460        TAUT=(1.+XI)*TAUTE $ GO TO 5135
 463   5130 TAUT=(1.-XI)*TAUTE
 466   5135 DO 5136 K=1,NSP
 513        KZ=IB2*JB2*(K-1)
 515        IJK=IJ+KZ
 516        IJPK=IJK+IB2
 517   5136 FAT(K)=((FA(IJPK)-DT*DDY*(FA(IJPK)-FA(IJK))*V(IJ))*V(IJ)
          1-2.*DDY*TAUT*(FA(IJPK)-FA(IJK)))*2.*RJ2(J)
 536   5140 IF(FL(IJ).EQ.1) GO TO 5150
 542        NFL=FL(IJ)-5 $ GO TO(5145,5150,5150,5145,5145,5150,5150,5145)NFL
 557   5145 DO 5146 K=1,NSP
 564   5146 FAL(K)=0.
 566        GO TO 5170
 566   5150 NFL=FL(IMJ) $ GO TO(5170,5145,5145,5145,5155,5170,5145,5145,5170
          1,5170,5145,5145,5170)NFL
 611   5155 IF(U(IMJ).EQ.0.)GO TO 5145
 614        TAUTE=0.125*DX*(U(IJ)-U(IMJ))
 621        IF(TAUTE.LT.0.) GO TO 5160
 623        TAUL=(1.+XI)*TAUTE $ GO TO 5165
 626   5160 TAUL=(1.-XI)*TAUTE
 631   5165 DO 5166 K=1,NSP
 653        KZ=IB2*JB2*(K-1)
 655        IJK=IJ+KZ
 656        IMJK=IJK-1
 657   5166 FAL(K)=2.*((FA(IMJK)-DT*DDX*(FA(IJK)-FA(IMJK))*U(IMJ))*
          1U(IMJ)-2.*DDX*TAUL*(FA(IJK)-FA(IMJK)))
 674   5170 IF(FL(IJ).EQ.1) GO TO 5180
 700        NFL=FL(IJ)-5 $ GO TO(5175,5175,5180,5180,5175,5175,5180,5180)NFL
 715   5175 DO 5176 K=1,NSP
 724   5176 TFAB(I,K)=0.
 727        GO TO 5185
 727   5180 NFL=FL(IJM) $ GO TO(5185,5175,5175,5175,5175,5185,5185,5175,5175
          1,5185,5185,5175,5175)NFL
 752   5185 CONTINUE
 752        SUMR=0.
 753        DO 5186 K=1,NSP
 774        KZ=IB2*JB2*(K-1)
 777        IJK=IJ+KZ
1000        FA(IJK)=FA(IJK)+0.5*DT*(DDY/RJ(J)*(TFAB(I,K)-FAT(K))+DDX*
          1(FAL(K)-FAR(K)))
1012        FAL(K)=FAR(K)
1013        TFAB(I,K)=FAT(K)
1015   5186 SUMR=SUMR+FA(IJK)
1023        IJNS=IB2*JB2*NSP+IJ
1026        FA(IJNS)=RHO(IJ)-SUMR
1033   5200 CONTINUE
      C
      C**** ALL SPECIES DENSITY   CHANGES DUE TO CONVECTION,ADDED TO-FA-ARRAY
      C
```

```
      C***** START OF SPECIES DIFFUSION CALCULATION
      C
1040        JC=0
1041        DO 5198 J=2,JB1
1042        JC=JC+IB2
1043        DO 5198 I=2,IB1
1045        IJ=I+JC
1046        IF(FL(IJ).GT.1.AND.FL(IJ).LT.6)GO TO 5198
1060        SUM1=0.
1060        SUM2=0.
1061        DO 5196 K=1,NSPEC
1063        IF(FA(I,J,K).LE.0.)GO TO 5196
1074        SUM1=SUM1+FA(I,J,K)*CV(K)*GAMA(K)
1100        SUM2=SUM2+FA(I,J,K)*CV(K)
1102   5196 CONTINUE
1106        TE(IJ)=RHO(IJ)*SIE(IJ)/SUM2
1112        TEIJ=TE(IJ)
1114        ETA(IJ)=ETAO*SQRT(TEIJ   )/(SUM1-SUM2)
1124   5198 CONTINUE
1131        DO 5193 I=1,IB2
1136   5193 SUMB(I)=0.
1140        JC=0
1141        DO 5400 J=2,JB1
1142        SUML=0.
1142        JC=JC+IB2
1144        DO 5400 I=2,IB1
1146        IJ=I+JC
1147        IF(FL(IJ).GT.1.AND.FL(IJ).LT.6) GO TO 5400
1161        IPJ=IJ+1
1161        IMJ=IJ-1
1162        IJP=IJ+IB2
1164        IJM=IJ-IB2
1164        IMJM=IJM-1
1165        IPJM=IJM+1
1166        IMJP=IJP-1
1170        IPJP=IJP+1
      C
      C*****FLUX ACROSS CELL RIGHT BOUNDARY
1170        IF(FL(IJ).EQ.1)GO TO 5209
1173        NFL=FL(IJ)-5 $ GO TO(5209,5203,5203,5209,5209,5203,5203,5209)NFL
1211   5203 DO 5206 K=1,NSPEC
1213   5206 DRHOR(K)=0. $GO TO 5219
1217   5209 NFL=FL(IPJ) $ GO TO(5212,5203,5203,5203,5203,5203,5212,5212,
           15203,5203,5212,5212,5203)NFL
1242   5212 DO 5213 K=1,NSPEC
1264        KZ=IB2*JB2*(K-1)
1266        IJK=IJ+KZ
1267        IPJK=IJK+1
1270        RHO1(K)=FA(IJK)/RHO(IJ)
1274        RHO2(K)=FA(IPJK)/RHO(IPJ)
1277   5213 DRHOP(K)=(RHO2(K)-RHO1(K))*.5*(RHO(IJ)+RHO(IPJ))
1303        ASSIGN 5215 TO JRET $ GO TO 5233
1305   5215 DO 5218 K=1,NSPEC
1323   5218 DRHOR(K)=.5*DDX*DRHO(K)*(ETA(IPJ)+ETA(IJ))
      C
```

```
      C*****FLUX ACROSS CELL TOP BOUNDARY
1325   5219 IF(FL(IJ).EQ.1)GO TO 5221 $ NFL=FL(IJ)-5
1333        GO TO(5221,5221,5224,5224,5221,5221,5224,5224)NFL
1346   5221 NFL=FL(IJP) $ GO TO(5231,5224,5224,5224,5224,5224,5224,5231,
           15231,5224,5224,5231,5231)NFL
1371   5224 DO 5225 K=1,NSPEC
1373   5225 DRHOT(K)=0. $ GO TO 5263
1377   5231 DO 5232 K=1,NSPEC
1421        KZ=IB2*JB2*(K-1)
1423        IJK=IJ+KZ
1424        IJPK=IJK+IB2
1426        RHO1(K)=FA(IJK)/RHO(IJ)
1432        RHO2(K)=FA(IJPK)/RHO(IJP)
1435   5232 DRHOP(K)=(RHO2(K)-RHO1(K))*.5*(RHO(IJ)+RHO(IJP))
1441        ASSIGN 5257 TO JRET
1442   5233 SDR=0. $ SDP=0. $ SDM=0.
1444        DO 5242 K=1,NSPEC
1446        DRHO(K)=ALPHA(K)*DRHOP(K)
1450        IF(DRHO(K).LT.0.)GO TO 5236
1452        SDP=SDP+DRHO(K) $ GO TO 5242
1454   5236 CONTINUE
1454        SDM=SDM+DRHO(K) $ GO TO 5242
1457   5242 SDR=SDR+DRHO(K)
1464        IF(SDR)5245,5254,5250
1465   5245 DO 5248 K=1,NSPEC
1467        IF(SDM.EQ.0.)GO TO 5248
1470        IF(DRHO(K).GE.0.)GO TO 5248
1472        DRHO(K)=DRHO(K)-DRHO(K)*SDR/SDM
1474   5248 CONTINUE $ GO TO 5254
1477   5250 DO 5251 K=1,NSPEC
1501        IF(SDP.EQ.0.)GO TO 5251
1502        IF(DRHO(K).LE.0.)GO TO 5251
1504        DRHO(K)=DRHO(K)-DRHO(K)*SDR/SDP
1506   5251 CONTINUE $ GO TO 5254
1511   5254 GO TO JRET
1514   5257 DO 5260 K=1,NSPEC
1533   5260 DRHOT(K)=.5*DDY*(ETA(IJ)+ETA(IJP))*RJ2(J)*DRHO(K)
      C
      C*****FLUX ACROSS CELL LEFT BOUNDARY
1535   5263 IF(FL(IJ).EQ.1)GO TO 5269 $ NFL=FL(IJ)-5
1543        GO TO(5266,5269,5269,5266,5266,5269,5269,5266)NFL
1556   5266 DO 5267 K=1,NSPEC
1560   5267 DRHOL(K)=0. $ GO TO 5283
1564   5269 NFL=FL(IMJ) $ GO TO(5283,5266,5266,5266,5266,5283,5266,5266,
           15283,5283,5266,5266,5283)NFL
      C
      C*****FLUX ACROSS CELL BOTTOM BOUNDARY
1607   5283 IF(FL(IJM).GT.1.AND.FL(IJM).LE.5)GO TO 5288
1622        GO TO 5292
1622   5288 DO 5289 K=1,NSPEC
1624        IK=IB2*(K-1)+I
1627   5289 DRHOB(IK)=0. $ GO TO 5292
1635   5292 CONTINUE
1635        DO 5295 K=1,NSPEC
1653        IJK=IJ+IB2*JB2*(K-1)
```

```
1656          IK=IB2*(K-1)+I                                                  RICE8
1661          FA(IJK)=FA(IJK)+DT*(DDX*(DRHOR(K)-DRHOL(K))+DDY/RJ(J)          RICE8
             1*(DRHOT(K)-DRHOB(IK)))                                          RICE8
1675          DRHOL (K)=DRHOR(K)                                              RICE8
1676     5295 DRHOB(IK)=DRHOT(K)                                              RICE8
      C                                                                       RICE8
      C**** ALL SPECIES DENSITY CHANGES DUE TO DIFFUSION,ADDED TO-FA-ARRAY    RICE8
      C**** START OF ENTHALPY DIFFUSION SECTION                               RICE8
      C                                                                       RICE8
1703          SUMR=0.                                                         RICE8
1704          DO 5302 K=1,NSPEC                                               RICE8
1712          SUMR=SUMR+GAMA(K)*CV(K)*DRHOR(K)                                RICE8
1714     5302 CONTINUE                                                        RICE8
1715          IF(SUMR.EQ.0.)GO TO 5316                                        RICE8
1716          SUMR=.5*(TE(IJ)+TE(IPJ))*SUMR                                   RICE8
1724     5314 CONTINUE                                                        RICE8
1724     5316 SUMT=0.                                                         RICE8
1725          DO 5319 K=1,NSPEC                                               RICE8
1734          SUMT=SUMT+GAMA(K)*CV(K)*DRHOT(K)                                RICE8
1736     5319 CONTINUE                                                        RICE8
1737          IF(SUMT.EQ.0.)GO TO 5328                                        RICE8
1740          SUMT=.5*(TE(IJ)+TE(IJP))*SUMT                                   RICE8
1746     5328 CONTINUE                                                        RICE8
1751          DRE=((SUMR-SUML)*DDX+(SUMT-SUMB(I))*DDY)*DT                     RICE8
1757          RE(IJ)=RE(IJ)+DRE                                               RICE8
1761          SIE(IJ)=SIE(IJ)+DRE/RHO(IJ)                                     RICE8
      C                                                                       RICE8
      C**** RE(IJ) AND SIE(IJ) UPDATED TO INCLUDE CHANGES DUE TO ENTHALPY DIFF RICE8
      C                                                                       RICE8
1766          SUMB(I)=SUMT                                                    RICE8
1767          SUML=SUMR                                                       RICE8
      C*****   END OF ENTHALPY DIFFUSION SECTION                              RICE8
      C*****                                                                  RICE8
      C**** START OF CHEMISTRY SECTION                                        RICE8
      C                                                                       RICE8
1771          IF(SIE(I,J).LE.0.)GO TO 5400                                    RICE8
1775          IF(NR.EQ.0)GO TO 5400                                           RICE8
1776          LMAX=DT/DTC+.1*DTC                                              RICE8
2002          DO 5395 M=1,LMAX                                                RICE8
2003          LA=-7                                                           RICE8
2003          LB=-7                                                           RICE8
2005          DO 5375 L=1,NR                                                  RICE8
2007          LA=LA+7                                                         RICE8
2010          LB=LB+7                                                         RICE8
2011          DOMEGA(L)=0.                                                    RICE8
2012          SUM=0.                                                          RICE8
2012          SUM1=0.                                                         RICE8
2013          RP=1.                                                           RICE8
2014          PP=1.                                                           RICE8
2016          DO 5330 K=1,NSPEC                                               RICE8
2020          KZ=IB2*JB2*(K-1)                                                RICE8
2023          IJK=IJ+KZ                                                       RICE8
2025          KLA=LA+K                                                        RICE8
2027          KLB=LB+K                                                        RICE8
2030          IF(FA(IJK).GT.0.)SUM=SUM+FA(IJK)*CV(K)                          RICE8
```

```
2037        ROM=FA(IJK)*RM(K)                                              RICE8
2042        IF(AM(KLA).EQ.0)GO TO 5320                                     RICE8
2044        IF(ROM.LE.0.)RP=0.                                             RICE8
2047        IF(ROM.GT.0.)RP=RP*ROM**AM(KLA)                                RICE8
2055   5320 CONTINUE                                                       RICE8
2055        IF(BM(KLB).EQ.0)GO TO 5330                                     RICE8
2057        IF(ROM.LE.0.)PP=0.                                             RICE8
2061        IF(ROM.GT.0.)PP=PP*ROM**BM(KLB)                                RICE8
2067   5330 CONTINUE                                                       RICE8
2072        TE(IJ)=RHO(IJ)*SIE(IJ)/SUM                                     RICE8
2076        IF(TE(IJ).LE.0.)GO TO 5400                                     RICE8
2102        KB=KF=0. $ NBR=NFR=0                                           RICE8
2106        IF(CB(L).LE.0.)GO TO 5332                                      RICE8
2110        NBR=1                                                          RICE8
      C*****BACKWARD REACTION COEF.                                        RICE8
2110        IF(EB(L).EQ.0.)EKB=1.                                          RICE8
2113        IF(EB(L).NE.0.)EKB=EXP(-EB(L)/TE(IJ))                          RICE8
2124        TEB=TE(IJ)**ZETAB(L)                                           RICE8
2132        KB=CB(L)*TEB*EKB                                               RICE8
2134   5332 CONTINUE                                                       RICE8
2134        IF(CF(L).LE.0.)GO TO 5331                                      RICE8
2136        NFR=1                                                          RICE8
      C*****FORWARD REACTION COEF.                                         RICE8
2137        IF(EF(L).EQ.0.)EKF=1.                                          RICE8
2142        IF(EF(L).NE.0.)EKF=EXP(-EF(L)/TE(IJ))                          RICE8
2153        TEF=TE(IJ)**ZETAF(L)                                           RICE8
2161        KF=CF(L)*TEF*EKF                                               RICE8
2163   5331 CONTINUE                                                       RICE8
2163        IF(NFR*NBR.NE.1)GO TO 5345                                     RICE8
2167        OMEGA=KF*RP-KB*PP                                              RICE8
      C**** BOTH FORWARD AND BACKWARD REACTION COEF. SPECIFIED             RICE8
      C**** TEST TO SEE WHICH IS GREATER                                   RICE8
2172        IF(OMEGA.LT.0.)NFR=0                                           RICE8
2175        IF(OMEGA.GE.0.)NBR=0                                           RICE8
2200   5345 IF(NFR.EQ.1)GO TO 5350                                         RICE8
2202        IF(NBR.EQ.1)GO TO 5355                                         RICE8
2204        GO TO 5375                                                     RICE8
2204   5350 CONTINUE                                                       RICE8
      C**** FINDS THE MINIMUM CONCENTRATION WHICH WILL BE THE REFERENCE SPECIE RICE8
      C**** FOR THE FORWARD REACTION                                       RICE8
2204        KNI=0                                                          RICE8
2205        DO 5353 K=1,NSPEC                                              RICE8
2207        KLA=LA+K                                                       RICE8
2210        IF(AM(KLA).EQ.0)GO TO 5353                                     RICE8
2212        IJK=IJ+IB2*JB2*(K-1)                                           RICE8
2216        ROM=FA(IJK)*RM(K)/AM(KLA)                                      RICE8
2224        IF(ROM.LE.EP(K))GO TO 5375                                     RICE8
2227        IF(AM(KLA).EQ.BM(KLA))GO TO 5353                               RICE8
2232        KNI=KNI+1                                                      RICE8
2233        IF(KNI.EQ.1)RMIN=ROM                                           RICE8
2236        IF(KNI.EQ.1)KALPHA=K                                           RICE8
2241        IF(KNI.EQ.1)GO TO 5353                                         RICE8
2242        IF(ROM.LT.RMIN)KALPHA=K                                        RICE8
2246        IF(ROM.LT.RMIN)RMIN=ROM                                        RICE8
2251   5353 CONTINUE                                                       RICE8
```

```
2254   5354 ROMREF=(FA(I,J,KALPHA)*RM(KALPHA)+DTC*AM(KALPHA,L)*KB*PP)/       RICE8
            1(1.+DTC*AM(KALPHA,L)*SM(KALPHA)*KF*RP/FA(I,J,KALPHA))           RICE8
2277        GO TO 5365                                                       RICE8
2300   5355 CONTINUE                                                         RICE8
       C**** FINDS THE MINIMUM CONCENTRATION WHICH WILL BE THE REFERENCE SPECIE RICE8
       C**** FOR THE BACKWARD REACTION                                       RICE8
2300        KNI=0                                                            RICE8
2301        DO 5358 K=1,NSPEC                                                RICE8
2303        KLB=LB+K                                                         RICE8
2304        IF(BM(KLB).EQ.0)GO TO 5358                                       RICE8
2306        IJK=IJ+IB2*JB2*(K-1)                                             RICE8
2312        ROM=FA(IJK)*RM(K)/BM(KLB)                                        RICE8
2320        IF(ROM.LE.EP(K))GO TO 5375                                       RICE8
2323        IF(BM(KLB).EQ.AM(KLB))GO TO 5358                                 RICE8
2326        KNI=KNI+1                                                        RICE8
2327        IF(KNI.EQ.1)RMIN=ROM                                             RICE8
2332        IF(KNI.EQ.1)KALPHA=K                                             RICE8
2335        IF(KNI.EQ.1)GO TO 5358                                           RICE8
2336        IF(ROM.LT.RMIN)KALPHA=K                                          RICE8
2342        IF(ROM.LT.RMIN)RMIN=ROM                                          RICE8
2345   5358 CONTINUE                                                         RICE8
2350   5359 ROMREF=(FA(I,J,KALPHA)*RM(KALPHA)+DTC*BM(KALPHA,L)*KF*RP)/       RICE8
            1(1.+DTC*BM(KALPHA,L)*SM(KALPHA)*KB*PP/FA(I,J,KALPHA))           RICE8
2373   5365 DOMEGA(L)=(ROMREF-FA(I,J,KALPHA)*RM(KALPHA))/(BM(KALPHA,L)-      RICE8
            1AM(KALPHA,L))                                                   RICE8
2412   5375 CONTINUE                                                         RICE8
2415        DO 5385 K=1,NSPEC                                                RICE8
2416        BMA=0.                                                           RICE8
2417        DO 5380 L=1,NR                                                   RICE8
2427   5380 BMA=BMA+(BM(K,L)-AM(K,L))*DOMEGA(L)                              RICE8
2434        FA(I,J,K)=FA(I,J,K)+SM(K)*BMA                                    RICE8
2444   5385 CONTINUE                                                         RICE8
2446        DRE=0.                                                           RICE8
2447        DO 5390 L=1,NR                                                   RICE8
2454   5390 DRE=DRE+Q(L)*DOMEGA(L)                                           RICE8
2460        RE(IJ)=RE(IJ)+DRE                                                RICE8
2462        SIE(IJ)=SIE(IJ)+DRE/RHO(IJ)                                      RICE8
       C                                                                     RICE8
       C**** RE(IJ) AND SIE(IJ) UPDATED TO INCLUDE CHANGES DUE TO CHEMICAL   RICE8
       C**** REACTION                                                        RICE8
       C                                                                     RICE8
2467   5395 CONTINUE                                                         RICE8
2472   5400 CONTINUE                                                         RICE8
2477        DO 5404 J=2,JB1                                                  RICE8
2501        IF(FL(IB2,J).NE.4)GO TO 5404                                     RICE8
2506        DO 5403 K=1,NSPEC                                                RICE8
2522        FA(IB1,J,K)=FA(IB,J,K)                                           RICE8
2524   5403 CONTINUE                                                         RICE8
       C                                                                     RICE8
       C**** SPECIES DENSITY(FA),RE AND SIE ARE SET CONSTANT AT OUTFLOW BND. RICE8
       C**** ENERGY CHANGE TO RE(IB1,J) DOES NOT CONTAIN KINETIC ENERGY      RICE8
       C                                                                     RICE8
2534        RE(IB1,J)=RE(IB1,J) + RHO(IB,J)*SIE(IB,J) - RHO(IB1,J)*SIE(IB1,J) RICE8
2545        RE(IB2,J)=RE(IB1,J)                                              RICE8
2547        SIE(IB2,J)=SIE(IB1,J)=SIE(IB,J)                                  RICE8
```

```
2561   5404 CONTINUE                                                        RICE8   1789
2564        RETURN $END                                                     RICE8   1790
     C                                                                      RICE8   1791
     C                                                                      RICE8   1792
     C**** END OF THE SUBROUTINE SPEC ************************************* RICE8   1793
     C                                                                      RICE8   1794
     C                                                                      RICE8   1795
```

```
          SUBROUTINE FLIC
 1        COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,
         1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLO,DTC,ALA,BIF,THETA,
         2PHI,OMEGA,GAMMA,DXOU,ASQ
 1        COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SECREQ,DDX,DDY
         1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT
 1        COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1,
         1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,
         2SIEL1,SIEL2
 1        REAL MUO,MINT1,MINT2,MINL1,MINL2
 1        COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),
         1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),
         2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)
 1        COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20)
         1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),
         2SIEBO(20),FARO(7,20),FABO(7,20)
 1        INTEGER AM,BM
 1        REAL MINRO,MINBO
    C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP)
    C**** AND THE *READ FEATURE OF (UPDATE)
    C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/
    C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC
    C
 1           LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32
         1),RHO( 62, 32),SIE( 62, 32),P( 62, 32
         2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62
         3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(
         4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(
         5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)
 1        DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(
         1 62, 32),DRHOB( 62, 7),STREB( 62)
 1        EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),
         1(TFAB,DRHOB)
 1        INTEGER FL,CYCLE
 1        REAL MU
 1        RAD=DY*JB
    C
    C**** THIS SUB.SETS CELL FLAGS BASED ON INPUT DATA,THERE ARE 13 TYPES
    C
 3        ZAX=DX*IB
 6        J2=1 $J3=5 $J4=9
11        YTS=-.5*DY
13        DO 245 J=1,JB2
14        XTS=-.5*DX
15        YTE=YTS+DY*(J-1)
23        DO 245 I=1,IB2
26        XTE=XTS+DX*(I-1)
    C
    C**** SETS EACH CELL FLAG,FL(I,J)=1,CELL FLAG WILL BE CHANGED FOR OTHER
    C     TYPES
32        FL(I,J)=1
35        IF(YTE.LT.0.)GO TO 210
36        IF(XTE.GT.ZAX)GO TO 220
42        IF(YTE.GT.RAD)GO TO 230
46        IF(XTE.LT.0.)GO TO 240
```

```
 46        GO TO 245
      C*****SETS FLAG FOR BOTTOM ROW
      C
 46     210 IF(NSL(1).EQ.1)GO TO 212
 50         FL(1,J)=2 $ GO TO 245
 55     212 FL(1,J)=3
 62         GO TO 245
      C*****SETS FLAGS FOR RIGHT (IB2) COLUMN
      C
 62     220 IF(YTE.LT.FLO(J2))  GO TO 222
 65         IF(YTE.LT.FLO(J2+1)) GO TO 228
 67         J2=J2+2 $ IF(J2.GT.3)J2=3
 73         IF(YTE.GT.FLO(J2).AND.YTE.LT.FLO(J2+1))GO TO 228
105     222 IF(NSL(2).EQ.1) GO TO 227
107         FL(1,J)=2 $ GO TO 245
114     227 FL(1,J)=3 $ GO TO 245
121     228 FL(1,J)=4 $ GO TO 245
      C*****SETS FLAGS FOR TOP (JB2) ROW
      C
126     230 IF(XTE.LT.FLO(J3)) GO TO 232
131         IF(XTE.LT.FLO(J3+1)) GO TO 238
133         J3=J3+2 $ IF(J3.GT.7)J3=7
137         IF(YTE.GT.FLO(J3).AND.YTE.LT.FLO(J3+1))GO TO 238
151     232 IF(NSL(3).EQ.1) GO TO 237
153         FL(I,J)=2  $ GO TO 245
160     237 FL(I,J)=3  $ GO TO 245
165     238 FL(I,J)=5  $ GO TO 245
      C*****SETS FLAGS FOR LEFT COLUMN
      C
172     240 IF(YTE.LT.FLO(J4)) GO TO 242
175         IF(YTE.LT.FLO(J4+1)) GO TO 248
177         J4=J4+2 $ IF(J4.GT.11)J4=11
203         IF(YTE.GT.FLO(J4).AND.YTE.LT.FLO(J4+1))GO TO 248
215     242 IF(NSL(4).EQ.1) GO TO 247
217         FL(I,J)=2  $ GO TO 245
224     247 FL(I,J)=3  $ GO TO 245
231     248 FL(I,J)=5
236     245 CONTINUE
243         IF(NO.LE.0)RETURN
      C
      C**** DEFINES OBSTACLE,(IF ANY),CELL FLAGS BASED ON DATA (OB,OIB,ITO,NSO
      C
246         DO 295 K=1,NO
250         X1=OB(1,K) $ X2=OB(2,K) $ Y1=OB(3,K) $ Y2=OB(4,K)
261         X3=OB(5,K) $ Y3=OB(6,K)
266         YTS=.5*DY
271         DO 285 J=2,JB1
272         XTS=.5*DX
273         YTE=YTS+DY*FLOAT(J-2)
300         DO 285 I=2,IB1
302         XTE=XTS+DX*FLOAT(I-2)
305         IF(XTE.LT.X1) GO TO 285
310         IF(XTE.GT.X2) GO TO 285
313         IF(YTE.LT.Y1) GO TO 285
315         IF(YTE.GT.Y2) GO TO 285
```

```
320          IF(ITO(K).EQ.1)GO TO 250                                  RICE8
322          IF(ITO(K)-3)283,275,270                                   RICE8
325      250 IYTE=1000.*YTE+.1*DY                                      RICE8
       C                                                               RICE8
       C**** CELL(I,J) IS A PARTIAL CELL                               RICE8
       C                                                               RICE8
331          IF(X3.EQ.X1.AND.Y3.EQ.Y2)GO TO 255                        RICE8
341          IF(X3.EQ.X1.AND.Y3.EQ.Y1)GO TO 260                        RICE8
350          IF(X3.EQ.X2.AND.Y3.EQ.Y1)GO TO 265                        RICE8
       C*****TRIANGLE LIES IN THIRD QUADRANT                           RICE8
       C                                                               RICE8
356          YTED=(Y1-Y2)/(X2-X1)*(XTE-X1)+Y2                          RICE8
364          IYTED=1000.*YTED+.1*DY                                    RICE8
367          IF(IYTE-IYTED)285,278,283                                 RICE8
       C*****TRIANGLE LIES IN FOURTH QUADRANT                          RICE8
       C                                                               RICE8
372      255 IYTED=1000.*((Y2-Y1)/(X2-X1)*(XTE-X1)+Y1)+.1*DY           RICE8
403          IF(IYTE-IYTED)285,279,283                                 RICE8
       C*****TRIANGLE LIES IN FIRST QUADRANT                           RICE8
       C                                                               RICE8
406      260 IYTED=1000.*((Y1-Y2)/(X2-X1)*(XTE-X1)+Y2)+.1*DY           RICE8
417          IF(IYTE-IYTED)283,276,285                                 RICE8
       C*****TRIANGLE LIES IN SECOND QUADRANT                          RICE8
       C                                                               RICE8
422      265 IYTED=1000.*((Y2-Y1)/(X2-X1)*(XTE-X1)+Y1)+.1*DY           RICE8
433          IF(IYTE-IYTED)283,277,285                                 RICE8
436      270 IF(YTE-DY.GT.Y1)GO TO 283                                 RICE8
443          IF(XTE.LT.OIB(1,K))GO TO 283                              RICE8
447          IF(XTE.GT.OIB(2,K))GO TO 283                              RICE8
       C                                                               RICE8
       C**** CELL(I,J) IS A SOLID WITH INFLOW ACROSS THE BOTTOM BOUNDARY  RICE8
       C                                                               RICE8
453          FL(I,J)=5 $ GO TO 295                                     RICE8
460      275 IF(XTE+DX.LT.X2)GO TO 283                                 RICE8
464          IF(YTE.LT.OIB(1,K))GO TO 283                              RICE8
470          IF(YTE.GT.OIB(2,K))GO TO 283                              RICE8
       C                                                               RICE8
       C**** CELL(I,J) IS A SOLID WITH INFLOW ACROSS THE RIGHT BOUNDARY   RICE8
       C                                                               RICE8
474          FL(I,J)=5 $ GO TO 285                                     RICE8
501      276 FL(I,J)=6 $ GO TO 281                                     RICE8
506      277 FL(I,J)=7 $ GO TO 281                                     RICE8
513      278 FL(I,J)=8 $ GO TO 281                                     RICE8
520      279 FL(I,J)=9 $ GO TO 281                                     RICE8
525      281 IF(NSO(K).EQ.1)FL(I,J)=FL(I,J)+4 $ GO TO 285              RICE8
537      283 IF(NSO(K).EQ.1)GO TO 284                                  RICE8
       C                                                               RICE8
       C**** CELL(I,J) IS A SOLID                                      RICE8
       C                                                               RICE8
542          FL(I,J)=2 $ GO TO 285                                     RICE8
547      284 FL(I,J)=3                                                 RICE8
554      285 CONTINUE                                                  RICE8
561      295 CONTINUE                                                  RICE8
564          RETURN                                                    RICE8
565          END                                                       RICE8
```

```
C                                                                        RICE8     1928
C                                                                        RICE8     1929
C**** END OF THE SUBROUTINE FLIC ****************************************RICE8     1930
C                                                                        RICE8     1931
C                                                                        RICE8     1932
```

```
          SUBROUTINE CVIC                                                RICE8    1933
      C**** TAPE-5 DATA FOR STORAGE AND RESTART                          RCOM1       2
  1       COMMON/TAPED/ITD,NTD,NSDMP,NFILE,NWDMP,NCYDMP                  RCOM1       3
      C**** THE RCOM1 DATA BLOCK IS USED IN RICE(MAIN-PROGRAM),PROG AND RTAPE5 RCOM1 4
      C                                                                  RCOM1       5
      C**** THE RCOM2 DATA BLOCK IS COMMON TO RICE AND ALL SUBROUTINES   RCOM1       6
      C**** BOTH RCOM1 AND RCOM2 ARE LOADED USING *COMDECK FEATURE OF (UPDATE) RCOM1 7
  1       COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,   RCOM2       2
         1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC,ALA,BIE,THETA, RCOM2      3
         2PHI,OMEGA,GAMMA,DXOU,ASQ                                       RCOM2       4
  1       COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SECREQ,DDX,DDY     RCOM2       5
         1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,X1,IT                          RCOM2       6
  1       COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1, RCOM2      7
         1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2, RCOM2      8
         2SIEL1,SIEL2                                                    RCOM2       9
  1       REAL MUO,MINT1,MINT2,MINL1,MINL2                               RCOM2      10
  1       COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                      RCOM2      11
         1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),    RCOM2      12
         2BM(7,3),C(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)         RCOM2      13
  1       COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20) RCOM2  14
         1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),            RCOM2      15
         2SIEBO(20),FARO(7,20),FABO(7,20)                                RCOM2      16
  1       INTEGER AM,BM                                                  RCOM2      17
  1       REAL MINRO,MINBO                                               RCOM2      18
      C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP) RCOM2 19
      C**** AND THE *READ FEATURE OF (UPDATE)                            RCOM2      20
      C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/   RCOM2      21
      C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC RCOM2  22
      C                                                                  RCOM2      23
  1          LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32                  RCOM2      24
         1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                          RCOM2      25
         2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62              RCOM2      26
         3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(               RCOM2      27
         4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                           RCOM2      28
         5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                        RCOM2      29
  1       DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                    RCOM2      30
         1 62, 32),DRHOB( 62, 7),STREB( 62)                              RCOM2      31
  1       EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),     RCOM2      32
         1(TFAB,DRHOB)                                                   RCOM2      33
  1       INTEGER FL,CYCLE                                               RCOM2      34
  1       REAL MU                                                        RCOM2      35
        3 FORMAT(1H06312)                                                RICE8    1936
        4 FORMAT(1H1)                                                    RICE8    1937
      C                                                                  RICE8    1938
      C**** THIS SUBROUTINE SETS SOME PROBLEM CONSTANTS AND DEFINES THE  RICE8    1939
      C**** COMPUTING MESH FLUID VARIABLE INITIAL CONDITIONS FROM INPUT DATA RICE8 1940
      C                                                                  RICE8    1941
  1       PRINT 4                                                        RICE8    1942
      C                                                                  RICE8    1943
      C**** DEFINES PROGRAM CONSTANTS BASED ON INPUT DATA                RICE8    1944
      C                                                                  RICE8    1945
  5       DDX=1./DX $ DDY=1./DY                                          RICE8    1946
 10       VCON=PHI*DT*DDY $DDX2=1./DX**2 $ DDY2=1./DY**2                 RICE8    1947
 16       UCON=PHI*DT*DDX                                                RICE8    1948
 20       ERRD=RHOO*.00001                                               RICE8    1949
```

```
 22          ITMAX=1000.
 23          PFAC=2.*THETA*PHI*DT**2*(DDY2+DDX2)
 31          XI=1.
 33          RRHO=1./RHOO
 35          DO 297 J=1,JB2
       C
       C**** DETERMINES COORDINATES TO BE USED---RECTANGULAR(ITC=0),CYL.(ITC=1)
       C
 37          IF(ITC.EQ.1)GO TO 294
 42          RJ(J)=1.
 43          RJ2(J)=1.
 46          GO TO 296
 54      294 RJ(J)=(J-1)*DY-DY/2.
 56          RJ2(J)=(J-1)*DY
 61      296 CONTINUE
 61          KPR=JB2-J+1
 64          WRITE( 9,3),(FL(I,KPR),I=1,IB2)
 76      297 WRITE(12,3),(FL(I,KPR),I=1,IB2)
114          PRINT 4
       C                        -----PRINTS CELL FLAGS-----
       C**** CLEARS ARRAYS----BYPASSES IF RESTART DATA READ FROM TAPE5
       C
117          IF(ITD.GT.1)GO TO 406
123          DO 405 J=1,JB2
124          DO 405 I=1,IB2
125          P(I,J)=RHOU(I,J)=RHOV(I,J)=PB(I,J)=0.
143          RHOVT(I,J)=RHOUT(I,J)=RHOT(I,J)=CSQ(I,J)=0.
160          RE(I,J)=U(I,J)=V(I,J)=SIE(I,J)=0.
175          RHO(I,J)=MU(I,J)=CQ(I,J)=ETA(I,J)=0.
212          TE(I,J)=0.
215          DO 403 K=1,NSPEC
225          FA(I,J,K)=0.0
226      403 CONTINUE
230      405 CONTINUE
235      406 CONTINUE
235          YTS=-.5*DY
237          DO 495 J=1,JB2
241          YTE=YTS+DY*(J-1)
245          XTS=-.5*DX
250          DO 490 I=1,IB2
251          IF(FL(I,J).NE.5.AND.ITD.GT.1)GO TO 490
264          XTE=XTS+DX*(I-1)
       C
       C**** SETS ALL INFLOW DATA INTO FICTIOUS CELL IF FL(I,J)=5
       C**** U-VEL.=0. AT TOP BOUNDARY,V-VEL.=0. AT LEFT BOUNDARY
       C
270          IF(J.EQ.1)GO TO 490
272          IF(J.EQ.JB2) GOTO 430
273          IF(I.EQ.1)GO TO 410
274          IF(I.EQ.IB2)GO TO 415
276          IF(I.EQ.IB1)GO TO 420
277          IF(J.EQ.JB1)GO TO 435
301          IF(FL(I,J).GT.1.AND.FL(I,J).LT.5)GO TO 490
314          IF(FL(I,J).EQ.1)GO TO 455
321          IF(NO.LE.0)GO TO 407
```

```
322          GO TO 475                                                    RICE8
323      407 U(I,J)=UO $ V(I,J)=VO $ GO TO 490                            RICE8
    C*****FIRST COLUMN (FICTIOUS CELLS)                                   RICE8
    C**** SETS ALL FLUID VARIABLES TO VALUES DEFINED FOR LEFT INFLOW OPENING RICE8
    C                                                                     RICE8
333      410 IF(FL(I,J).NE.5)GO TO 490                                    RICE8
340          IF(YTE.LT.FLO(10))GO TO 413                                  RICE8
343          U(I,J)=UINL2 $ RHO(I,J)=RINL2 $ SIE(I,J)=SIEL2 $ MU(I,J)=MINL2 RICE8
360          DO 411 K=1,NSPEC                                             RICE8
362      411 FA(I,J,K)=FAL2(K) $ GO TO 490                                RICE8
372      413 U(I,J)=UINL1 $ RHO(I,J)=RINL1 $ SIE(I,J)=SIEL1 $ MU(I,J)=MINL1 RICE8
410          DO 414 K=1,NSPEC                                             RICE8
411      414 FA(I,J,K)=FAL1(K) $ GO TO 490                                RICE8
    C*****LAST COLUMN (FICTIOUS CELLS)                                    RICE8
    C**** ONLY THE U-VEL.ARE DEFINED AT THE OUTFLOW BOUNDARY              RICE8
    C                                                                     RICE8
421      415 IF(FL(I,J).NE.4)GO TO 490                                    RICE8
426          U(I,J)=UO $ GO TO 490                                        RICE8
    C*****LAST COLUMN OF REAL CELLS(I=IB1) ,FL(IB1,J)CANNOT = 5 (IN-FLOW CEL RICE8
    C                                                                     RICE8
432      420 IF(J.EQ.JB1)GO TO 435                                        RICE8
434          IF(FL(I,J).GT.1.AND.FL(I,J).LT.6)GO TO 490                   RICE8
447          IF(FL(I,J).EQ.1)GO TO 427                                    RICE8
454          IF(NO.LE.0)GO TO 407                                         RICE8
455          LAG=FL(I,J)-5                                                RICE8
462          GO TO (423,460,490,423,423,460,490,423) LAG                  RICE8
475      423 CONTINUE                                                     RICE8
475      427 IF(FL(I+1,J).NE.4) GO TO 460                                 RICE8
502          U(I,J)=UO $ GO TO 460                                        RICE8
    C                                                                     RICE8
    C**** TOP ROW(FICTIOUS CELLS) (J=JB2)                                 RICE8
    C**** SETS ALL FLUIO VARIABLES TO VALUES DEFINED FOR TOP INFLOW OPENINGS RICE8
    C                                                                     RICE8
506      430 IF(FL(I,J).EQ.5)GO TO 432                                    RICE8
513          GO TO 490                                                    RICE8
514      432 IF(XTE.LT.FLO(6))GOTO 434                                    RICE8
517          RHO(I,J)=RINT2 $ SIE(I,J)=SIET2 $ MU(I,J)=MINT2              RICE8
531          DO 433 K=1,NSPEC                                             RICE8
533      433 FA(I,J,K)=FAT2(K) $ GO TO 490                                RICE8
543      434 RHO(I,J)=RINT1 $ SIE(I,J)=SIET1 $ MU(I,J)=MINT1              RICE8
556          DO 436 K=1,NSPEC                                             RICE8
557      436 FA(I,J,K)=FAT1(K) $ GO TO 490                                RICE8
    C*****LAST ROW OF REAL CELLS (J=JB1)                                  RICE8
    C                                                                     RICE8
567      435 IF(FL(I,J).GT.1.AND.FL(I,J).LT.5)GO TO 490                   RICE8
603          IF(FL(I,J).EQ.1)GO TO 455                                    RICE8
610          IF(FL(I,J).EQ.5) GO TO 485                                   RICE8
614          IF(NO.LE.0)GO TO 407                                         RICE8
615      437 LAG=FL(I,J)-5                                                RICE8
622          GO TO (443,438,490,445,443,438,490,445) LAG                  RICE8
635      438 IF(FL(I,J+1).NE.5) GO TO 490                                 RICE8
642          IF(XTE.LT.FLO(6)) GO TO 440                                  RICE8
645          V(I,J)=VINT2 $ GO TO 490                                     RICE8
    C*****SETS V(I,J) BASED ON CELL(I,J+1).FLUID BELOW ,U(I,J) HAS NOT BEEN RICE8
    C                                                                     RICE8
```

```
 652     440 V(I,J)=VINT1 $ GO TO 490                                              RICE8
 657     443 IF(FL(I,J+1).NE.5)GO TO 445                                           RICE8
 664         U(I,J)=U0                                                             RICE8
 667         IF(XTE.LT.FLO(6))GO TO 444                                            RICE8
 672         V(I,J)=VINT2 $ GO TO 490                                              RICE8
 676     444 V(I,J)=VINT1 $ GO TO 490                                              RICE8
      C*****SETS U(I,J) BASED ON CELL(I+1,J),FLUID AT LEFT,V(I,J) HAS BEEN SET RICE8
      C                                                                            RICE8
 703     445 IF(FL(I+1,J).EQ.2)GO TO 490                                           RICE8
 710         IF(FL(I+1,J).EQ.3)GO TO 490                                           RICE8
 714         IF(FL(I+1,J).EQ.4)GO TO 449                                           RICE8
 720         IF(FL(I+1,J).EQ.1) GO TO 449                                          RICE8
 724         IF(FL(I+1,J).EQ.5) GO TO 490                                          RICE8
 730          LAG=FL(I+1,J)-5                                                      RICE8
 733         GO TO (490,449,449,490,490,449,449,490)  LAG                          RICE8
 746     449 U(I,J)=U0 $ GO TO 490                                                 RICE8
      C*****FLUID REAL CELLS  ************************************************ RICE8
      C*****SETS U(I,J) BASED ON CELL(I+1,J),FLUID AT LEFT,V(I,J) HAS NOT BEEN RICE8
      C                                                                            RICE8
 753     455 IF(FL(I+1,J).EQ.2)GO TO 460                                           RICE8
 760         IF(FL(I+1,J).EQ.3)GO TO 460                                           RICE8
 764         IF(FL(I+1,J).EQ.4)GO TO 459                                           RICE8
 770         IF(FL(I+1,J).EQ.1) GO TO 459                                          RICE8
 774         IF(FL(I+1,J).EQ.5) GO TO 460                                          RICE8
1000          LAG=FL(I+1,J)-5                                                      RICE8
1003         GO TO (460,459,459,460,460,459,459,460)  LAG                          RICE8
1016     459 U(I,J)=U0 $ GO TO 460                                                 RICE8
      C*****SETS V(I,J) BASED ON CELL(I,J+1),FLUID BELOW,U(I,J) HAS BEEN SET       RICE8
      C                                                                            RICE8
1023     460 IF(FL(I,J+1).GT.1.AND.FL(I,J+1).LT.5)GO TO 490                        RICE8
1037         IF(FL(I,J+1).EQ.1)GO TO 468                                           RICE8
1044         IF(FL(I,J+1).EQ.5) GO TO 470                                          RICE8
1050          LAG=FL(I,J+1)-5                                                      RICE8
1053         GO TO (490,490,468,468,490,490,468,468)  LAG                          RICE8
1066     468 V(I,J)=V0 $ GO TO 490                                                 RICE8
1073     470 ASSIGN 473 TO NOR $ GO TO 507                                         RICE8
1075     473 IF(ITO(KO).EQ.3) GO TO 490                                            RICE8
1100         V(I,J)=VINBO(KO) $ GO TO 490                                          RICE8
1105     475 IF(FL(I,J).EQ.5) GO TO 485                                            RICE8
1112         LAG=FL(I,J)-5                                                         RICE8
1115         GO TO (455,460,490,455,455,460,490,455) LAG                           RICE8
1130     485 ASSIGN 488 TO NOR $ GO TO 505                                         RICE8
1132     488 IF(ITO(KO).EQ.4) GO TO 490                                            RICE8
1135         U(I,J)=UINRO(KO) $ GO TO 490                                          RICE8
      C                                                                            RICE8
      C**** SETS U OR V VEL..BASED ON TYPE OF OBSTACLE                             RICE8
      C                                                                            RICE8
      C*****DEFINES OBSTACLE NUMBER AND RETURNS TO NOR                             RICE8
1142     505 XIO=XTE $ YIO=YTE $ GO TO 515                                         RICE8
1146     506 XIO=XTE+DX $ YIO=YTE $ GO TO 515                                      RICE8
1152     507 XIO=XTE $ YIO=YTE+DY $ GO TO 515                                      RICE8
1156     515 DO 520 K=1,NO                                                         RICE8
1160         IF(OB(1,K).LT.XIO.AND.OB(2,K).GT.XIO) GO TO 517                       RICE8
1173         GO TO 520                                                             RICE8
1173     517 IF(OB(3,K).LT.YIO.AND.OB(4,K).GT.YIO) GO TO 521                       RICE8
```

```
1206      520 CONTINUE                                                  RICE6    2115
1211      521 KO=K $ GO TO NOR                                          RICE8    2116
1216      490 CONTINUE                                                  RICE8    2117
1221      495 CONTINUE                                                  RICE8    2118
1223          YTS=-.5*DY                                                RICE8    2119
1225          DO 530 J=2,JB1                                            RICE8    2120
1227          YTE=YTS+DY*(J-1) $ XTS=-.5*DX                             RICE8    2121
1235          DO 529 I=2,IB2                                            RICE8    2122
1237          IF(FL(I,J).NE.5.AND.ITD.GT.1)GO TO 529                    RICE8    2123
1252          XTE=XTS+DX*(I-1)                                          RICE8    2124
1256          IF(FL(I,J).EQ.1)GO TO 525                                 RICE8    2125
1263          IF(FL(I,J).EQ.4)GO TO 525                                 RICE8    2126
1267          IF(FL(I,J).GT.5)GO TO 525                                 RICE8    2127
1274          IF(FL(I,J).NE.5)GO TO 529                                 RICE8    2128
1277          ASSIGN 523 TO NOR $ GO TO 505                             RICE8    2129
1301      523 IF(ITO(KO).EQ.3)GOTO 524                                  RICE8    2130
      C                                                                 RICE8    2131
      C**** SETS OBSTACLE INFLOW CELLS TO INPUT DATA VALUES             RICE8    2132
      C                                                                 RICE8    2133
1304          RHO(I,J)=RINBO(KO) $ SIE(I,J)=SIEBO(KO)                   RICE8    2134
1313          MU(I,J)=MINBO(KO)                                         RICE8    2135
1316          DO 526 K=1,NSPEC                                          RICE8    2136
1320      526 FA(I,J,K)=FABO(K,KO) $ GO TO 529                          RICE8    2137
1332      524 RHO(I,J)=RINRO(KO) $ SIE(I,J)=SIERO(KO)                   RICE8    2138
1342          MU(I,J)=MINRO(KO)                                         RICE8    2139
1345          DO 527 K=1,NSPEC                                          RICE8    2140
1347      527 FA(I,J,K)=FARO(K,KO) $ GO TO 529                          RICE8    2141
      C                                                                 RICE8    2142
      C**** SETS SPECIFIED VALUES INTO ALL FLUID REAL CELLS             RICE8    2143
      C                                                                 RICE8    2144
1361      525 RHO(I,J)=RHO0 $ SIE(I,J)=SIE0 $ MU(I,J)=MU0               RICE8    2145
1374          DO 528 K=1,NSPEC                                          RICE8    2146
1375      528 FA(I,J,K)=FA0(K)      $ GO TO 529                         RICE8    2147
1405      529 CONTINUE                                                  RICE8    2148
1410      530 CONTINUE                                                  RICE8    2149
1412      600 DO 655 J=2,JB2                                            RICE8    2150
1414          DO 655 I=1,IB1                                            RICE8    2151
1415          IF(FL(I,J).GT.1.AND.FL(I,J).LT.5)GO TO 655                RICE8    2152
      C                                                                 RICE8    2153
      C**** DEFINES COMPUTED FLUID VARIABLES                            RICE8    2154
1430          IF(NSPEC.LE.0)GO TO 603                                   RICE8    2155
1431          SUM=0.                                                    RICE8    2156
1432          SUM1=0.                                                   RICE8    2157
1433          DO 602 K=1,NSPEC                                          RICE8    2158
1434          IF(FA(I,J,K).LE.0.)GO TO 602                              RICE8    2159
1445          SUM=SUM+FA(I,J,K)*CV(K)                                   RICE8    2160
1450          SUM1=SUM1+FA(I,J,K)*CV(K)*GAMA(K)                         RICE8    2161
1453      602 CONTINUE                                                  RICE8    2162
1456          IF(SUM.NE.0.)GAMMA=SUM1/SUM                               RICE8    2163
1460          IF(SUM.LE.0.)TE(I,J)=0.                                   RICE8    2164
1466          IF(SUM.GT.0.)TE(I,J)=RHO(I,J)*SIE(I,J)/SUM                RICE8    2165
1502      603 CONTINUE                                                  RICE8    2166
1502          P(I,J)=(GAMMA-1.)*RHO(I,J)*SIE(I,J)                       RICE8    2167
1515          IF(FL(I,J).EQ.5)GO TO 645                                 RICE8    2168
1523          RHOU(I,J)=(RHO(I,J)+RHO(I+1,J))*U(I,J)*.5                 RICE8    2169
```

```
1531          RHOV(I,J)=(RHO(I,J)+RHO(I,J+1))*V(I,J)*.5          RICE8
1540          IF(FL(I,J).EQ.1)GO TO 605 $ NFL=FL(I,J)-5          RICE8
1547          GO TO (610,615,615,610,610,615,615,610)NFL         RICE8
1562      605 XKE=.25*(U(I,J)+U(I-1,J))**2 $ GO TO 620           RICE8
1573      610 XKE=U(I,J)**2 $ GO TO 620                          RICE8
1600      615 XKE=U(I-1,J)**2                                    RICE8
1605      620 IF(FL(I,J).EQ.1)GO TO 625 $ NFL=FL(I,J)-5          RICE8
1616          GO TO (630,630,635,635,630,630,635,635)NFL         RICE8
1631      625 YKE=.25*(V(I,J)+V(I,J-1))**2 $ GO TO 640           RICE8
1642      630 YKE=V(I,J)**2 $ GO TO 640                          RICE8
1647      635 YKE=V(I,J-1)**2                                    RICE8
1654      640 RE(I,J)=RHO(I,J)*(SIE(I,J)+.5*(XKE+YKE))           RICE8
1671          GO TO 655                                          RICE8
1671      645 IF(U(I,J).EQ.0.) GO TO 650                         RICE8
1677          RE(I,J)=RHO(I,J)*(SIE(I,J)+.5*U(I,J)**2)           RICE8
1704          RHOU(I,J)=RHO(I,J)*U(I,J)                          RICE8
1710          GO TO 655                                          RICE8
1714      650 RE(I,J)=RHO(I,J)*(SIE(I,J)+.5*V(I,J-1)**2)         RICE8
1722          RHOV(I,J)=RHO(I,J)*V(I,J-1)                        RICE8
1726      655 CONTINUE                                           RICE8
1733          RETURN                                             RICE8
1734          END                                                RICE8
      C                                                          RICE8
      C                                                          RICE8
      C**** END OF THE SUBROUTINE CVIC ************************** RICE8
      C                                                          RICE8
      C                                                          RICE8
```

```
            SUBROUTINE VPLOT(K,XX1,YY1,XX2,YY2)                         RICE8   2198
14          COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,   RCOM2      2
           1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,IPL,TPLD,DTC,ALA,BIE,THETA,  RCOM2      3
           2PHI,OMEGA,GAMMA,DXOU,ASO                                     RCOM2      4
14          COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SECREO,DDX,DDY     RCOM2      5
           1,UCON,VCON,FRRO,ITMAX,PFAC,RRHO,X1,IT                        RCOM2      6
14          COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1,  RCOM2      7
           1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,  RCOM2      8
           2SIEL1,SIEL2                                                  RCOM2      9
14          REAL MUO,MINT1,MINT2,MINL1,MINL2                             RCOM2     10
14          COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                     RCOM2     11
           1FAL1(7),FAL2(7),SH(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),    RCOM2     12
           2BM(7,3),D(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)        RCOM2     13
14          COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20) RCOM2  14
           1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),           RCOM2     15
           2SIEBO(20),FARO(7,20),FABO(7,20)                              RCOM2     16
14          INTEGER AM,BM                                                RCOM2     17
14          REAL MINRO,MINBO                                             RCOM2     18
     C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP) RCOM2   19
     C**** AND THE *READ FEATURE OF (UPDATE)                             RCOM2     20
     C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD     RCOM2     21
     C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC  RCOM2  22
     C                                                                   RCOM2     23
14             LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32               RCOM2     24
           1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                        RCOM2     25
           2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62            RCOM2     26
           3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(             RCOM2     27
           4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                         RCOM2     28
           5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                      RCOM2     29
14          DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                  RCOM2     30
           1 62, 32),DRHOB( 62, 7),STREB( 62)                            RCOM2     31
14          EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),    RCOM2     32
           1(TFAB,DRHOB)                                                 RCOM2     33
14          INTEGER FL,CYCLE                                             RCOM2     34
14          REAL MU                                                      RCOM2     35
14          DIMENSION KFLO(12),KOB(6,20),KOIB(2,20)                      RICE8   2199
     C       K=0 CONVERT GRID,K=1 PLOTS GRID,K=2 CONVERTS AND PLOTS VECTORS RICE8  2200
      2015 FORMAT(20X,A10)                                               RICE8   2201
      2016 FORMAT(3X,10A8* T=*1PE12.5* CYCLE=*I5)                        RICE8   2202
14          IF(K-1)1,15,25                                               RICE8   2203
16        1 IYB=916                                                      RICE8   2204
17          XL=0.                                                        RICE8   2205
20          XR=IB*DX                                                     RICE8   2206
22          YT=JB*DY                                                     RICE8   2207
25          YB=0.                                                        RICE8   2208
25          IF(XR.LE.1.13556*YT)GO TO 2                                  RICE8   2209
31          IXL=0 $ IXR=1022 $ IYT=916-YT*1022/XR $ GO TO 5              RICE8   2210
40        2 X=XR*450/YT $ IXL=511-X $ IXR=511+X $ IYT=16                 RICE8   2211
     C*****CONVERTS GRID TO 4020 COORDINATES---K=0                       RICE8   2212
52        5 CONTINUE                                                     RICE8   2213
52          DO 8 J=1,12                                                  RICE8   2214
54          IF(J.GE.5.AND.J.LE.8) GO TO 6                                RICE8   2215
65          CALL CONVRT(FLO(J),KFLO(J),YB,YT,IYB,IYT)                    RICE8   2216
73          GO TO 8                                                      RICE8   2217
77        6 CALL CONVRT(FLO(J),KFLO(J),XL,XR,IXL,IXR)                    RICE8   2218
```

```
111        8 CONTINUE                                                RICE8
113          IF(NO.LE.0)RETURN                                        RICE8
116          DO 14 N=1,NO                                             RICE8
120          DO 9 J=1,2                                               RICE8
121        9 CALL CONVRT(OB(J,N),KOB(J,N),XL,XR,IXL,IXR)              RICE8
137          CALL CONVRT(OB(5,N),KOB(5,N),XL,XR,IXL,IXR)              RICE8
150          DO 10 J=3,4                                              RICE8
155       10 CALL CONVRT(OB(J,N),KOB(J,N),YB,YT,IYB,IYT)              RICE8
173          CALL CONVRT(OB(6,N),KOB(6,N),YB,YT,IYB,IYT)              RICE8
204          DO 11 J=1,2                                              RICE8
211          IF(ITO(N).EQ.3) CALL CONVRT(OIB(J,N),KOIB(J,N),YB,YT,IYB,IYT)   RICE8
227       11 IF(ITO(N).EQ.4) CALL CONVRT(OIB(J,N),KOIB(J,N),XL,XR,IXL,IXR)   RICE8
247       14 CONTINUE                                                 RICE8
252          RETURN                                                   RICE8
     C*****PLOTS GRID USING DRV(...)                                  RICE8
253       15 CONTINUE                                                 RICE8
253          CALL ADV(1)                                              RICE8
255          NGR=0                                                    RICE8
256          IF(LPR.LE.0)GO TO 17                                     RICE8
263          CALL LINCNT(60)                                          RICE8
264          WRITE (12,2015) JAM                                      RICE8
275          WRITE (12,2016) NAME,TIME,CYCLE                          RICE8
315       17 CONTINUE                                                 RICE8
315          CALL DRV(IXL,IYB,IXR,IYB)                                RICE8
321          IF(FLO( 2).LE.FLO( 1)) GO TO 101  $CALL DRV(IXR,IYB,IXR,KFLO(1))   RICE8
334          IF(FLO( 3).LE.FLO( 2))GO TO 100                          RICE8
342          CALL DRV(IXR,KFLO(2),IXR,KFLO(3)) $ CALL DRV(IXR,KFLO(4),IXR,IYT)  RICE8
355          GO TO 102                                                RICE8
361      100 CALL DRV(IXR,KFLO(2),IXR,IYT)                            RICE8
366          GO TO 102                                                RICE8
372      101 CALL DRV(IXR,IYB,IXR,IYT)                                RICE8
401      102 IF(FLO(6).LE.FLO(5))GO TO 111 $ CALL DRV(IXL,IYT,KFLO(5),IYT)  RICE8
411          IF(FLO( 7).LE.FLO( 6))GO TO 110                          RICE8
417          CALL DRV(KFLO(6),IYT,KFLO(7),IYT) $  CALL DRV(KFLO(8),IYT,IXR,IYT) RICE8
430          GO TO 112                                                RICE8
434      110 CALL DRV(KFLO(6),IYT,IXR,IYT)                            RICE8
440          GO TO 112                                                RICE8
444      111 CALL DRV(IXL,IYT,IXR,IYT)                                RICE8
453      112 IF(FLO(10).LE.FLO(9))GO TO 121 $ CALL DRV(IXL,IYB,IXL,KFLO(9))  RICE8
463          IF(FLO(11).LE.FLO(10))GO TO 120                          RICE8
471          CALL DRV(IXL,KFLO(10),IXL,KFLO(11))$CALL DRV(IXL,KFLO(12),IXL,IYT) RICE8
504          GO TO 122                                                RICE8
510      120 CALL DRV(IXL,KFLO(10),IXL,IYT)                           RICE8
515          GO TO 122                                                RICE8
521      121 CALL DRV(IXL,IYB,IXL,IYT)                                RICE8
530      122 CONTINUE                                                 RICE8
530          IF(NO.LE.0)RETURN                                        RICE8
533          DO 295 N=1,NO $ KX1=KOB(1,N) $ KX2=KOB(2,N) $ KY1=KOB(3,N)   RICE8
544          KY2=KOB(4,N) $ KX3=KOB(5,N) $ KY3=KOB(6,N)               RICE8
553          IF(ITO(N).EQ.1) GO TO 250                                RICE8
556          CALL DRV(KX1,KY1,KX1,KY2) $ CALL DRV(KX1,KY2,KX2,KY2)    RICE8
564          IF(ITO(N)-3)205,215,225                                  RICE8
572      205 CALL DRV(KX1,KY1,KX2,KY1) $ CALL DRV(KX2,KY1,KX2,KY2) $ GO TO 295  RICE8
605      215 CALL DRV(KX1,KY1,KX2,KY1) $ CALL DRV(KX2,KY1,KX2,KOIB(1,N))   RICE8
620          CALL DRV(KX2,KOIB(2,N),KX2,KY2) $ GO TO 295              RICE8
```

```
 633       225 CALL DRV(KX2,KY1,KX2,K(2) $ CALL DRV(KX1,KY1,KOIB(1,N),KY1)     RICE8
 646           CALL DRV(KOIB(2,N),KY1,KX2,KY1) $ GO TO 295                    RICE8
 657       250 IF(OB(5,N).EQ.OB(1,N).AND.OB(6,N).EQ.OB(3,N)) GO TO 255         RICE8
 673           IF(OB(5,N).EQ.OB(2,N).AND.OB(6,N).EQ.OB(4,N)) GO TO 255         RICE8
 706           IF(OB(5,N).EQ.OB(2,N).AND.OB(6,N).EQ.OB(3,N)) GO TO 260         RICE8
 721           IF(OB(5,N).EQ.OB(1,N).AND.OB(6,N).EQ.OB(4,N)) GO TO 260         RICE8
 734           GO TO 295                                                       RICE8
 734       255 CALL DRV(KX2,KY1,KX3,KY3) $ CALL DRV(KX1,KY2,KX3,KY3)           RICE8
 743           CALL DRV(KX1,KY2,KX2,KY1) $ GO TO 295                           RICE8
 752       260 CALL DRV(KX1,KY1,KX3,KY3) $ CALL DRV(KX2,KY2,KX3,KY3)           RICE8
 761           CALL DRV(KX1,KY1,KX2,KY2)                                       RICE8
 767       295 CONTINUE                                                        RICE8
 772           IF(NGR.EQ.1) GO TO 22                                           RICE8
 774           NGR=1                                                           RICE8
 775           GO TO 17                                                        RICE8
 775        22 CONTINUE                                                        RICE8
 775           RETURN                                                          RICE8
     C*****CONVERTS AND PLOTS VEL.VECTORS FOR EACH REAL CELL---K=2             RICE8
 776        25 CONTINUE                                                        RICE8
 776           CALL CONVRT(XX1,IX1,XL,XR,IXL,IXR)                              RICE8
1002           CALL CONVRT(XX2,IX2,XL,XR,IXL,IXR)                              RICE8
1011           CALL CONVRT(YY1,IY1,YB,YT,,YB,IYT)                              RICE8
1020           CALL CONVRT(YY2,IY2,YB,YT,IYB,IYT)                              RICE8
1027           CALL DRV(IX1,IY1,IX2,IY2)                                       RICE8
1032           RETURN                                                          RICE8
1033           END                                                             RICE8
     C                                                                         RICE8
     C                                                                         RICE8
     C**** END OF THE SUBROUTINE VPLOT *************************************** RICE8
     C                                                                         RICE8
     C                                                                         RICE8
```

```
          SUBROUTINE CNPLOT(K,JSP)                                         RICE8    2305
 6        COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,    RCOM2       2
         1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLO,DTC,ALA,BIE,THETA,  RCOM2       3
         2PHI,OMEGA,GAMMA,DXOU,ASQ                                         RCOM2       4
 6        COMMON/ICCD/IB,JB,IBI,JBI,NAME(10),JNM,TBEG,SECREQ,DDX,DDY      RCOM2       5
         1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT                            RCOM2       6
 6        COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1, RCOM2       7
         1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,  RCOM2       8
         2SIEL1,SIEL2                                                      RCOM2       9
 6        REAL MUO,MINT1,MINT2,MINL1,MINL2                                 RCOM2      10
 6        COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                        RCOM2      11
         1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),      RCOM2      12
         2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)           RCOM2      13
 6        COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),PINRO(20) RCOM2    14
         1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),              RCOM2      15
         2SIEBO(20),FARO(7,20),FABO(7,20)                                  RCOM2      16
 6        INTEGER AM,BM                                                    RCOM2      17
 6        REAL MINRO,MINBO                                                 RCOM2      18
C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP)   RCOM2      19
C**** AND THE *READ FEATURE OF (UPDATE)                                    RCOM2      20
C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/           RCOM2      21
C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC     RCOM2      22
C                                                                          RCOM2      23
 6          LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32                     RCOM2      24
         1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                            RCOM2      25
         2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62                RCOM2      26
         3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(                 RCOM2      27
         4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                             RCOM2      28
         5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                          RCOM2      29
 6        DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                      RCOM2      30
         1 62, 32),DRHOB( 62, 7),STREB( 62)                                RCOM2      31
 6        EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),      RCOM2      32
         1(TFAB,DRHOB)                                                     RCOM2      33
 6        INTEGER FL,CYCLE                                                 RCOM2      34
 6        REAL MU                                                          RCOM2      35
 2001 FORMAT(4X,*RHO*)                                                     RICE8    2307
 2002 FORMAT(4X,*PB*)                                                      RICE8    2308
 2003 FORMAT(4X,*SIE*)                                                     RICE8    2309
 2004 FORMAT(4X,*TEMPERATURE*)                                             RICE8    2310
 2005 FORMAT(4X,*MACH NO.*)                                                RICE8    2311
 2006 FORMAT(4X,*SPEC.FA*)                                                 RICE8    2312
 2007 FORMAT(4X,*SPEC.FB*)                                                 RICE8    2313
 2008 FORMAT(4X,*SPEC.FC*)                                                 RICE8    2314
 2009 FORMAT(4X,*SPEC.FD*)                                                 RICE8    2315
 2010 FORMAT(4X,*SPEC.FE*)                                                 RICE8    2316
 2011 FORMAT(4X,*SPEC.FF*)                                                 RICE8    2317
 2012 FORMAT(4X,*SPEC.FG*)                                                 RICE8    2318
 2015 FORMAT(20X,A10)                                                      RICE8    2319
 2016 FORMAT(3X,10A8* T=*1PE12.5* CYCLE=*I5)                               RICE8    2320
 2017 FORMAT(3X,*QMX=*F8.4,*   QMN=*F8.4,*   MAX.CON.LINE=*F8.4,*   MIN.   RICE8    2321
     1CON.LINE=*F8.4,*   INTERVAL=*F8.4)                                   RICE8    2322
C      K=0 CONVERT GRID,K=1 PLOTS GRID,K=2 CONVERTS AND PLOTS CONTOUR      RICE8    2323
 6        DIMENSION ZC(20),CON(11)                                         RICE8    2324
 6        DIMENSION XY(2),YX(2)                                            RICE8    2325
 6        DIMENSION KFLO(12),KOB(6,20),KOIB(2,20)                          RICE8    2326
```

```
   6          IF(K-1)1,15,25                                              RICE6
  10        1 IYB=916                                                     RICE8
  11          XL=0.                                                       RICE8
  12          XR=IB*DX                                                    RICE8
  14          YT=JB*DY                                                    RICE8
  17          YB=0.                                                       RICE8
  17          IF(XR.LE.1.13556*YT)GO TO  2                                RICE8
  23          IXL=0 $ IXR=1022 $ IYT=916-YT*1022/XR $ GO TO  5            RICE8
  32        2 X=XR*450/YT $ IXL=511-X $ IXR=511+X $ IYT=16                RICE8
        C*****CONVERTS GRID TO 4020 COORDINATES---K=0                     RICE8
  44        5 CONTINUE                                                    RICE8
  44          DO 8 J=1,12                                                 RICE8
  46          IF(J.GE.5.AND.J.LE.8) GO TO 6                               RICE8
  57          CALL CONVRT(FLO(J),KFLO(J),YB,YT,IYB,IYT)                   RICE8
  63          GO TO 8                                                     RICE8
  65        6 CALL CONVRT(FLO(J),KFLO(J),XL,XR,IXL,IXR)                   RICE8
  75        8 CONTINUE                                                    RICE8
  77          IF(NO.LE.0)RETURN                                           RICE8
 102          DO 14 N=1,NO                                                RICE8
 104          DO 9 J=1,2                                                  RICE8
 105        9 CALL CONVRT(OB(J,N),KOB(J,N),XL,XR,IXL,IXR)                 RICE8
 121          CALL CONVRT(OB(5,N),KOB(5,N),XL,XR,IXL,IXR)                 RICE8
 130          DO 10 J=3,4                                                 RICE8
 133       10 CALL CONVRT(OB(J,N),KOB(J,N),YB,YT,IYB,IYT)                 RICE8
 147          CALL CONVRT(OB(6,N),KOB(6,N),YB,YT,IYB,IYT)                 RICE8
 156          DO 11 J=1,2                                                 RICE8
 161          IF(ITO(N).EQ.3) CALL CONVRT(OIB(J,N),KOIB(J,N),YB,YT,IYB,IYT)  RICE8
 175       11 IF(ITO(N).EQ.4) CALL CONVRT(OIB(J,N),KOIB(J,N),XL,XR,IXL,IXR)  RICE8
 213       14 CONTINUE                                                    RICE8
 216          RETURN                                                      RICE8
        C*****PLOTS GRID USING DRV---K 1                                  RICE8
 217       15 CONTINUE                                                    RICE8
 217          CALL ADV(1)                                                 RICE8
 221          NGR=0                                                       RICE8
 222          IF(LPR.LE.0)GO TO 17                                        RICE8
 225          CALL LINCNT(60)                                             RICE8
 226          WRITE (12,2015) JNM                                         RICE8
 235          WRITE (12,2016) NAME,TIME,CYCLE                             RICE8
 252       17 CONTINUE                                                    RICE8
 252          CALL DRV(IXL,IYB,IXR,IYB)                                   RICE8
 256          IF(FLO( 2).LE.FLO( 1)) GO TO 101  $CALL DRV(IXR,IYB,IXR,KFLO(1))  RICE8
 266          IF(FLO( 3).LE.FLO( 2))GO TO 100                             RICE8
 272          CALL DRV(IXR,KFLO(2),IXR,KFLO(3)) $ CALL DRV(IXR,KFLO(4),IXR,IYT)  RICE8
 302          GO TO 102                                                   RICE8
 304      100 CALL DRV(IXR,KFLO(2),IXR,IYT)                               RICE8
 310          GO TO 102                                                   RICE8
 312      101 CALL DRV(IXR,IYB,IXR,IYT)                                   RICE8
 320      102 IF(FLO(6).LE.FLO(5))GO TO 111 $ CALL DRV(IXL,IYT,KFLO(5),IYT)  RICE8
 326          IF(FLO( 7).LE.FLO( 6))GO TO 110                             RICE8
 332          CALL DRV(KFLO(6),IYT,KFLO(7),IYT) $  CALL DRV(KFLO(8),IYT,IXR,IYT) RICE8
 342          GO TO 112                                                   RICE8
 344      110 CALL DRV(KFLO(6),IYT,IXR,IYT)                               RICE8
 350          GO TO 112                                                   RICE8
 352      111 CALL DRV(IXL,IYT,IXR,IYT)                                   RICE8
 360      112 IF(FLO(10).LE.FLO(9))GO TO 121 $ CALL DRV(IXL,IYB,IXL,KFLO(9))  RICE8
```

```
366        IF(FLO(11).LE.FLO(10))GO TO 120                                    RICE8
372        CALL DRV(IXL,KFLO(10),IXL,KFLO(11))$CALL DRV(IXL,KFLO(12),IXL,IYT) RICE8
402        GO TO 122                                                          RICE8
404    120 CALL DRV(IXL,KFLO(10),IXL,IYT)                                     RICE8
410        GO TO 122                                                          RICE8
412    121 CALL DRV(IXL,IYB,IXL,IYT)                                          RICE8
420    122 CONTINUE                                                           RICE8
420        IF(NO.LE.0)RETURN                                                  RICE8
423        DO 295 N=1,NO $ KX1=KOB(1,N) $ KX2=KOB(2,N) $ KY1=KOB(3,N)         RICE8
434        KY2=KOB(4,N) $ KX3=KOB(5,N) $ KY3=KOB(6,N)                         RICE8
443        IF(ITO(N).EQ.1) GO TO 250                                          RICE8
446        CALL DRV(KX1,KY1,KX1,KY2) $ CALL DRV(KX1,KY2,KX2,KY2)              RICE8
454        IF(ITO(N)-3)205,215,225                                            RICE8
461    205 CALL DRV(KX1,KY1,KX2,KY1) $ CALL DRV(KX2,KY1,KX2,KY2) $ GO TO 295  RICE8
472    215 CALL DRV(KX1,KY1,KX2,KY1) $ CALL DRV(KX2,KY1,KX2,KO1B(1,N))        RICE8
503        CALL DRV(KX2,KO1B(2,N),KX2,KY2) $ GO TO 295                        RICE8
512    225 CALL DRV(KX2,KY1,KX2,KY2) $ CALL DRV(KX1,KY1,KO1B(1,N),KY1)        RICE8
523        CALL DRV(KO1B(2,N),KY1,KX2,KY1) $ GO TO 295                        RICE8
532    250 IF(OB(5,N).EQ.OB(1,N).AND.OB(6,N).EQ.OB(3,N)) GO TO 255           RICE8
545        IF(OB(5,N).EQ.OB(2,N).AND.OB(6,N).EQ.OB(4,N)) GO TO 255           RICE8
560        IF(OB(5,N).EQ.OB(2,N).AND.OB(6,N).EQ.OB(3,N)) GO TO 260           RICE8
573        IF(OB(5,N).EQ.OB(1,N).AND.OB(6,N).EQ.OB(4,N)) GO TO 260           RICE8
606        GO TO 295                                                          RICE8
606    255 CALL DRV(KX2,KY1,KX3,KY3) $ CALL DRV(KX1,KY2,KX3,KY3)              RICE8
615        CALL DRV(KX1,KY2,KX2,KY1) $ GO TO 295                              RICE8
622    260 CALL DRV(KX1,KY1,KX3,KY3) $ CALL DRV(KX2,KY2,KX3,KY3)              RICE8
631        CALL DRV(KX1,KY1,KX2,KY2)                                          RICE8
636    295 CONTINUE                                                           RICE8
641        IF(NGR.EQ.1) GO TO 22                                              RICE8
643        NGR=1                                                              RICE8
644        GO TO 17                                                           RICE8
644     22 CONTINUE                                                           RICE8
644        RETURN                                                             RICE8
645     25 CONTINUE                                                           RICE8
645        XS=XL+DX/2. $ YS=YB+DY/2.                                          RICE8
653        DO 28 J=2,JB1                                                      RICE8
654        DO 28 I=2,IB1                                                      RICE8
6          IF(FL(I,J).NE.1)GO TO 28                                           RICE8
662        QMN=QMX=CQ(I,J)                                                    RICE8
666        GO TO 29                                                           RICE8
666     28 CONTINUE                                                           RICE8
673     29 CONTINUE                                                           RICE8
673        DO 41 J=2,JB1                                                      RICE8
675        DO 41 I=2,IB1                                                      RICE8
676        IF(FL(I   ,J   ).GT.1.AND.FL(I   ,J   ).LT.6) GO TO 41             RICE8
     C*****CALCULATES -SPACING BETWEEN CONTOUR LINES (DQ)-MAX.CQ (QMX)-MIN.CQ RICE8
711        IF(CQ(I,J).LT.QMN) QMN=CQ(I,J)                                     RICE8
721        IF(CQ(I,J).GT.QMX)QMX=CQ(I,J)                                      RICE8
732     41 CONTINUE                                                           RICE8
737        DQ=(QMX-QMN)/10. $ SUM=QMN $ CC=0.                                 RICE8
743        MINQ=2                                                             RICE8
745        DO 43 I=1,11                                                       RICE8
746        CON(I)=SUM+(I-1)*DQ                                                RICE8
753        IF(CC.GT.0.) GO TO 43                                              RICE8
755        IF(CON(I).LE.QMN) GO TO 43                                         RICE8
```

```
 757        CC=1. $ MINQ=1                                                RICE6  2437
 761     43 CONTINUE                                                      RICE8  2438
 763        IF(LPR.LE.0)GO TO 48                                          RICE8  2439
 765        CALL LINCNT(62)                                               RICE8  2440
 766        IF(K.EQ.3) GO TO 52                                           RICE8  2441
 771        GO TO (44,45,46,47,49) JSP                                    RICE8  2442
1002     44 WRITE(12,2001)                                                RICE8  244
1006        GO TO 51                                                      RICE8  2444
1010     45 WRITE(12,2002)                                                RICE8  2445
1014        GO TO 51                                                      RICE8  2446
1016     46 WRITE(12,2003)                                                RICE8  2447
1022        GO TO 51                                                      RICE8  2448
1024     47 WRITE(12,2004)                                                RICE8  2449
1030        GO TO 51                                                      RICE8  2450
1032     49 WRITE(12,2005)                                                RICE8  2451
1036        GO TO 51                                                      RICE8  2452
1040     52 GO TO (53,54,56,57,58,59,62)JSP                               RICE8  2453
1053     53 WRITE(12,2006)                                                RICE8  2454
1057        GO TO 51                                                      RICE8  2455
1061     54 WRITE(12,2007)                                                RICE8  2456
1065        GO TO 51                                                      RICE8  2457
1067     56 WRITE(12,2008)                                                RICE8  2458
1073        GO TO 51                                                      RICE8  2459
1075     57 WRITE(12,2009)                                                RICE8  2460
1101        GO TO 51                                                      RICE8  2461
1103     58 WRITE(12,20 0)                                                RICE8  2462
1107        GO TO 51                                                      RICE8  2463
1111     59 WRITE(12,2011)                                                RICE8  2464
1115        GO TO 51                                                      RICE8  2465
1117     62 WRITE(12,2012)                                                RICE8  2466
1125     51 CONTINUE                                                      RICE8  2467
1125        CALL LINCNT(59)                                               RICE8  2468
1127        WRITE(12,2017) QMX,QMN,CON(10),CON(MINQ),DQ                   RICE8  2469
1151     48 CONTINUE                                                      RICE8  2470
1151        DO 97 J=2,JB                                                  RICE8  2471
1153        YD=YS+(J-2)*DY                                                RICE8  2472
1157        DO 95 I=2,IB                                                  RICE8  2473
1161        XD=XS+(I-2)*DX                                                RICE8  2474
     C*****BYPASSES OBSTACLE                                              RICE8  2475
1165        L12=L13=L24=L34=0.                                            RICE8  2476
1171        IF(FL(I  ,J  ).GT.1.AND.FL(I  ,J  ).LT.6) L12=L13=1           RICE8  2477
1207        IF(FL(I+1,J  ).GT.1.AND.FL(I+1,J  ).LT.6)L12=L24=1            RICE8  2478
1223        IF(FL(I  ,J+1).GT.1.AND.FL(I  ,J+1).LT.6) L13=L34=1           RICE8  2479
1240        IF(FL(I+1,J+1).GT.1.AND.FL(I+1,J+1).LT.6) L24=L34=1           RICE8  2480
1254        IF(FL(I,J).LT.6)GO TO 55 $ NFL=FL(I,J)-5                      RICE8  2481
1265        GO TO (74,70,68,71,74,70,68,71)NFL                            RICE8  2482
1300     55 IF(FL(I+1,J).LT.6)GO TO 60 $ NFL=FL(I+1,J)-5                  RICE8  2483
1311        GO TO (70,74,72,69,70,74,72,69)NFL                            RICE8  2484
1324     60 IF(FL(I,J+1).LT.6)GO TO 65 $ NFL=FL(I,J+1)-5                  RICE8  2485
1335        GO TO (71,66,73,74,71,66,73,74)NFL                            RICE8  2486
1350     65 IF(FL(I+1,J+1).LT.6)GO TO 74 $ NFL=FL(I+1,J+1)-5              RICE8  2487
1361        GO TO (67,72,74,73,67,72,74,73)NFL                            RICE8  2488
1374     66 L13=L34=1 $ GO TO 74                                          RICE8  248
1377     67 L24=L34=1 $ GO TO 74                                          RICE8  24
1402     68 L12=L13=1 $ GO TO 74                                          RICE8  249
```

```
1405      69 L12=L24=1 $ GO TO 74                                    RICE8
1410      70 L12=1 $ GO TO 74                                        RICE8
1412      71 L13=1 $ GO TO 74                                        RICE8
1414      72 L24=1 $ GO TO 74                                        RICE8
1416      73 L34=1 $ GO TO 74                                        RICE8
1420      74 CONTINUE                                                RICE8
1420         DO 93 N=1,11                                            RICE8
      C*****IF CON.GT.CQ SETS KN=1----IF CON.LE.CQ SETS KN=0         RICE8
1422         M=12-N $ K1=K2=K3=K4=1                                  RICE8
1427         IF(CON(M).LE.CQ(I  ,J  ))K1=0                           RICE8
1436         IF(CON(M).LE.CQ(I+1,J  ))K2=0                           RICE8
1445         IF(CON(M).LE.CQ(I  ,J+1))K3=0                           RICE8
1454         IF(CON(M).LE.CQ(I+1,J+1))K4=0                           RICE8
1463         IF(K1*K2*K3*K4.NE.0)GO TO 93                            RICE8
1470         IF((K1+K2+K3+K4).NE.0)GO TO 75                          RICE8
1471         GO TO 93                                                RICE8
      C*****FINDS TWO INTERSECTION POINTS OF FOUR CELL REGION        RICE8
1472      75 IC=0                                                    RICE8
1473         IF((K1+K3).NE.1)GO TO 77                                RICE8
1476         IF(L13.EQ.1)GO TO 77                                    RICE8
1477         DR=0. $ III=I $ IC=IC+1 $ ASSIGN 77 TO KR1 $ GO TO 86   RICE8
1504      77 IF((K1+K2).NE.1)GO TO 79                                RICE8
1507         IF(L12.EQ.1)GO TO 79                                    RICE8
1510         DZ=0. $ JJ1=J $ IC=IC+1 $ ASSIGN 79 TO KR1 $ GO TO 87   RICE8
1515      79 IF((K2+K4).NE.1)GO TO 81                                RICE8
1520         IF(L24.EQ.1)GO TO 81                                    RICE8
1521         DR=DX $ III=I+1 $ IC=IC+1 $ ASSIGN 81 TO KR1 $ GO TO 86 RICE8
1527      81 IF((K3+K4).NE.1)GO TO 83                                RICE8
1532         IF(L34.EQ.1)GO TO 83                                    RICE8
1533         DZ=DY $ JJ1=J+1 $ IC=IC+1 $ ASSIGN 83 TO KR1 $ GO TO 87 RICE8
1541      83 GO TO 93                                                RICE8
1546      86 XY(IC)=XD+DR                                            RICE8
1547         YX(IC)=YD+DY*((CON(M)-CQ(III,J))/(CQ(III,J+1)-CQ(III,J)))  RICE8
1557         IF(IC.EQ.2)GO TO 89 $ GO TO KR1                         RICE8
1570      87 YX(IC)=YD+DZ                                            RICE8
1571         XY(IC)=XD+DX*((CON(M)-CQ(I,JJ1))/(CQ(I+1,JJ1)-CQ(I,JJ1)))  RICE8
1601         IF(IC.EQ.2)GO TO 89 $ GO TO KR1                         RICE8
      C*****CONVERTS AND PLOTS CONTOURS FOR FOUR REAL CELLS----K=2   RICE8
1606      89 CALL CONVRT(XY(1) ,IX1,XL,XR,IXL,IXR)                   RICE8
1613         CALL CONVRT(XY(2),IX2,XL,XR,IXL,IXR)                    RICE8
1620         CALL CONVRT(YX(1),IY1,YB,YT,IYB,IYT)                    RICE8
1625         CALL CONVRT(YX(2),IY2,YB,YT,IYB,IYT)                    RICE8
1632         IF(M.EQ.10) CALL PLT(IX1,IY1,24)                        RICE8
1640         IF(M.EQ.MINO) CALL PLT(IX1,IY1,35)                      RICE8
1647         CALL DRV(IX1,IY1,IX2,IY2)                               RICE8
1657      93 CONTINUE                                                RICE8
1661      95 CONTINUE                                                RICE8
1664      97 CONTINUE                                                RICE8
1666         RETURN                                                  RICE8
1667         END                                                     RICE8
      C                                                              RICE8
      C                                                              RICE8
      C**** END OF THE SUBROUTINE CNPLOT ******************************** RICE8
      C                                                              RICE8
      C                                                              RICE8
```

```
          SUBROUTINE COPLOT(K,JJSP)                                              RICE8    2547
 6        COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,           RCOM2       2
         1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC,ALA,BIE,THETA,        RCOM2       3
         2PHI,OMEGA,GAMMA,DXOU,ASQ                                              RCOM2       4
 6        COMMON/ICCD/IB,JB,IB1,JB1,NAME(10),JNM,TBEG,SECREQ,DDX,DDY            RCOM2       5
         1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT                                 RCOM2       6
 6        COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1,       RCOM2       7
         1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,        RCOM2       8
         2SIEL1,SIEL2                                                           RCOM2       9
 6        REAL MUO,MINT1,MINT2,MINL1,MINL2                                      RCOM2      10
 6        COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),                             RCOM2      11
         1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),           RCOM2      12
         2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)                RCOM2      13
 6        COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20)    RCOM2      14
         1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),                   RCOM2      15
         2SIEBO(20),FARO(7,20),FABO(7,20)                                       RCOM2      16
 6        INTEGER AM,BM                                                         RCOM2      17
 6        REAL MINRO,MINBO                                                      RCOM2      18
      C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP)  RCOM2      19
      C**** AND THE *READ FEATURE OF (UPDATE)                                   RCOM2      20
      C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/          RCOM2      21
      C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC    RCOM2      22
      C                                                                         RCOM2      23
 6           LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32                         RCOM2      24
         1),RHO( 62, 32),SIE( 62, 32),P( 62, 32                                 RCOM2      25
         2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62                     RCOM2      26
         3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(                      RCOM2      27
         4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(                                  RCOM2      28
         5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)                               RCOM2      29
 6        DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(                           RCOM2      30
         1 62, 32),DRHOB( 62, 7),STREB( 62)                                     RCOM2      31
 6        EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),            RCOM2      32
         1(TFAB,DRHOB)                                                          RCOM2      33
 6        INTEGER FL,CYCLE                                                      RCOM2      34
 6        REAL MU                                                               RCOM2      35
      C   K=0 CONVERT GRID,K=1 PLOTS GRID,K=2 CONVERTS AND PLOTS SPECIES        RICE8    2549
 6        DIMENSION KFLO(12),KOB(6,20),KOIB(2,20)                               RICE8    2550
       2015 FORMAT(20X,A7)                                                      RICE8    2551
       2016 FORMAT(3X,10A8* T=*1PE12.5* CYCLE=*I5)                              RICE8    2552
 6        IF(K-1)1,15,25                                                        RICE8    2553
10      1 IYB=916                                                               RICE8    2554
11        XL=0.                                                                 RICE8    2555
12        XR=IB*DX                                                              RICE8    2556
14        YT=JB*DY                                                              RICE8    2557
17        YB=0.                                                                 RICE8    2558
17        IF(XR.LE.1.13556*YT)GO TO  2                                          RICE8    2559
23        IXL=0 $ IXR=1022 $ IYT=916-YT*1022/XR $ GO TO  5                      RICE8    2560
32      2 X=XR*450/YT $ IXL=511-X $ IXR=511+X $ IYT=16                          RICE8    2561
      C*****CONVERTS GRID TO 4020 COORDINATES---K=0                             RICE8    2562
44      5 CONTINUE                                                              RICE8    2563
44        DO 8 J=1,12                                                           RICE8    2564
46        IF(J.GE.5.AND.J.LE.8) GO TO 6                                         RICE8    2565
57        CALL CONVRT(FLO(J),KFLO(J),YB,YT,IYB,IYT)                             RICE8    2566
63        GO TO 8                                                               RICE8    2567
65      6 CALL CONVRT(FLO(J),KFLO(J),XL,XR,IXL,IXR)                             RICE8    2568
```

```
 75        8 CONTINUE                                                          RICE8    2569
 77          IF(NO.LE.0)RETURN                                                 RICE8    2570
102          DO 14 N=1,NO                                                      RICE8    2571
104          DO 9 J=1,2                                                        RICE8    2572
105        9 CALL CONVRT(OB(J,N),KOB(J,N),XL,XR,IXL,IXR)                       RICE8    2573
121          CALL CONVRT(OB(5,N),KOB(5,N),XL,XR,IXL,IXR)                       RICE8    2574
130          DO 10 J=3,4                                                       RICE8    2575
133       10 CALL CONVRT(OB(J,N),KOB(J,N),YB,YT,IYB,IYT)                       RICE8    2576
147          CALL CONVRT(OB(6,N),KOB(6,N),YB,YT,IYB,IYT)                       RICE8    2577
156          DO 11 J=1,2                                                       RICE8    2578
161          IF(ITO(N).EQ.3) CALL CONVRT(OIB(J,N),KOIB(J,N),YB,YT,IYB,IYT)     RICE8    2579
175       11 IF(ITO(N).EQ.4) CALL CONVRT(OIB(J,N),KOIB(J,N),XL,XR,IXL,IXR)     RICE8    2580
213       14 CONTINUE                                                          RICE8    2581
216          RETURN                                                            RICE8    2582
      C*****PLOTS GRID USING DRV---K=1                                         RICE8    2583
217       15 CONTINUE                                                          RICE8    2584
217          NGR=0                                                             RICE8    2585
220          IF(LPR.LE.0)GO TO 17                                              RICE8    2586
222          CALL LINCNT(60)                                                   RICE8    2587
223          WRITE (12,2015) JNM                                               RICE8    2588
232          WRITE (12,2016) NAME,TIME,CYCLE                                   RICE8    2589
247       17 CALL DRV(IXL,IYB,IXR,IYB)                                         RICE8    2590
253          IF(FLO( 2).LE.FLO( 1)) GO TO 101  $CALL DRV(IXR,IYB,IXR,KFLO(1))  RICE8    2591
263          IF(FLO( 3).LE.FLO( 2))GO TO 103                                   RICE8    2592
267          CALL DRV(IXR,KFLO(2),IXR,KFLO(3)) $ CALL DRV(IXR,KFLO(4),IXR,IYT) RICE8    2593
277          GO TO 102                                                         RICE8    2594
301      103 CALL DRV(IXR,KFLO(2),IXR,IYT)                                     RICE8    2595
305          GO TO 102                                                         RICE8    2596
307      101 CALL DRV(IXR,IYB,IXR,IYT)                                         RICE8    2597
315      102 IF(FLO(6).LE.FLO(5))GO TO 111 $ CALL DRV(IXL,IYT,KFLO(5),IYT)     RICE8    2598
323          IF(FLO( 7).LE.FLO( 6))GO TO 110                                   RICE8    2599
327          CALL DRV(KFLO(6),IYT,KFLO(7),IYT) $  CALL DRV(KFLO(8),IYT,IXR,IYT) RICE8   2600
337          GO TO 112                                                         RICE8    2601
341      110 CALL DRV(KFLO(6),IYT,IXR,IYT)                                     RICE8    2602
345          GO TO 112                                                         RICE8    2603
347      111 CALL DRV(IXL,IYT,IXR,IYT)                                         RICE8    2604
355      112 IF(FLO(10).LE.FLO(9))GO TO 121 $ CALL DRV(IXL,IYB,IXL,KFLO(9))    RICE8    2605
363          IF(FLO(11).LE.FLO(10))GO TO 120                                   RICE8    2606
367          CALL DRV(IXL,KFLO(10),IXL,KFLO(11))$CALL DRV(IXL,KFLO(12),IXL,IYT) RICE8   2607
377          GO TO 122                                                         RICE8    2608
401      120 CALL DRV(IXL,KFLO(10),IXL,IYT)                                    RICE8    2609
405          GO TO 122                                                         RICE8    2610
407      121 CALL DRV(IXL,IYB,IXL,IYT)                                         RICE8    2611
415      122 CONTINUE                                                          RICE8    2612
415          IF(NO.LE.0)RETURN                                                 RICE8    2613
420          DO 295 N=1,NO $ KX1=KOB(1,N) $ KX2=KOB(2,N) $ KY1=KOB(3,N)        RICE8    2614
431          KY2=KOB(4,N) $ KX3=KOB(5,N) $ KY3=KOB(6,N)                        RICE8    2615
440          IF(ITO(N).EQ.1) GO TO 250                                         RICE8    2616
443          CALL DRV(KX1,KY1,KX1,KY2) $ CALL DRV(KX1,KY2,KX2,KY2)             RICE8    2617
451          IF(ITO(N)-3)205,215,225                                           RICE8    2618
456      205 CALL DRV(KX1,KY1,KX2,KY1) $ CALL DRV(KX2,KY1,KX2,KY2) $ GO TO 295 RICE8    2619
467      215 CALL DRV(KX1,KY1,KX2,KY1) $ CALL DRV(KX2,KY1,KX2,KOIB(1,N))       RICE8    2620
500          CALL DRV(KX2,KOIB(2,N),KX2,KY2) $ GO TO 295                       RICE8    2621
507      225 CALL DRV(KX2,KY1,KX2,KY2) $ CALL DRV(KX1,KY1,KOIB(1,N),KY1)       RICE8    2622
520          CALL DRV(KOIB(2,N),KY1,KX2,KY1) $ GO TO 295                       RICE8    2623
```

```
 527     250 IF(OB(5,N).EQ.OB(1,N).AND.OB(6,N).EQ.OB(3,N)) GO TO 255
 542         IF(OB(5,N).EQ.OB(2,N).AND.OB(6,N).EQ.OB(4,N)) GO TO 255
 555         IF(OB(5,N).EQ.OB(2,N).AND.OB(6,N).EQ.OB(3,N)) GO TO 260
 570         IF(OB(5,N).EQ.OB(1,N).AND.OB(6,N).EQ.OB(4,N)) GO TO 260
 603         GO TO 295
 603     255 CALL DRV(KX2,KY1,KX3,KY3) $ CALL DRV(KX1,KY2,KX3,KY3)
 612         CALL DRV(KX1,KY2,KX2,KY1) $ GO TO 295
 517     260 CALL DRV(KX1,KY1,KX3,KY3) $ CALL DRV(KX2,KY2,KX3,KY3)
 626         CALL DRV(KX1,KY1,KX2,KY2)
 633     295 CONTINUE
 636         IF(NGR.EQ.1) GO TO 22
 640         NGR=1
 641         GO TO 17
 641      22 CONTINUE
 641         RETURN
      C*****CONVERTS AND PLOTS SPECIES FOR EACH REAL CELL---K=2
      C**** TO PLOT IN COLOR USE ICHAR=42 AND CHANGE COLOR FOR EACH SPECIES
      C
 642      25 CONTINUE
 642         GO TO (35,37,39,41,43,44,46)JJSP
 655      35 ICHAR=42 $ GO TO 45
 657      37 ICHAR=16 $ GO TO 45
 661      39 ICHAR=44 $ GO TO 45
 663      41 ICHAR=32 $ GO TO 45
 665      43 ICHAR=63 $ GO TO 45
 667      44 ICHAR=58 $ GO TO 45
 671      46 ICHAR=13 $ GO TO 45
 673      45 CONTINUE
 673         DO 100 J=2,JB2
 675         YB1=YB+(J-2)*DY
 701         DO 100 I=2,IB2
 703         IF(FL(I,J).GT.1.AND.FL(I,J).LT.6) GO TO 100
 720         XL1=XL+(I-2)*DX
 725         KK=CQ(I,J)
 727         IF(KK.LT.1) GO TO 100
 731         DO 50 L=1,KK
      C
      C**** IF COLOR MOVIE IS BEING MADE,RANDSV AND RANDST MUST BE USED TO
      C**** KEEP DOTS STEADY IN EACH CELL(START RANDOM WITH SAME NUMBER)
      C
 733         Y1Y=RANDOM(DUMMY)*DY+YB1
 737         X1X=RANDOM(DUMMY)*DX+XL1
 744         IF(FL(I,J).EQ.1) GO TO 65
 752         XPC=X1X-XL1
 753         YPC=Y1Y-YB1
 756         NFL=FL(I,J)-5 $ GO TO (52,54,56,58,52,54,56,58)NFL
 776      52 YLINE=    DY-DY*DDX*XPC
1001         IF(YPC.LE.YLINE)GO TO 50 $ GO TO 65
1004      54 YLINE=    DY*DDX*XPC
1006         IF(YPC.LE.YLINE)GO TO 50 $ GO TO 65
1012      56 YLINE=    DY-DY*DDX*XPC
1015         IF(YPC.GE.YLINE)GO TO 50 $ GO TO 65
1020      58 YLINE=    DY*DDX*XPC
1022         IF(YPC.GE.YLINE)GO TO 50 $ GO TO 65
1026      65 CONTINUE
```

```
1026        CALL CONVRT(YIY,IY,YB,YT,IYB,IYT)                          RICE6    2679
1033        CALL CONVRT(XIX,IX,XL,XR,IXL,IXR)                          RICE8    2680
1037        CALL PLT(IX,IY,ICHAR)                                      RICE8    2681
1044     50 CONTINUE                                                   RICE8    2682
1047    100 CONTINUE                                                   RICE8    2683
1054        RETURN                                                     RICE8    2684
1055        END                                                        RICE8    2685
      C                                                                RICE8    2686
      C                                                                RICE8    2687
      C**** END OF THE SUBROUTINE COPLOT ***************************** RICE8    2688
      C                                                                RICE8    2689
      C                                                                RICE8    2690
```

```
      SUBROUTINE TANVEL
      COMMON/GEOMD/IB2,JB2,NO,NSL(4),LPR,CYCLE,ITC,JPLOT(5),NSPEC,
     1ISPEC(7),NR,DX,DY,TIME,DT,TSTOP,TPR,TPL,TPLD,DTC,ALA,BIE,THETA,
     2PHI,OMEGA,GAMMA,DXOU,ASQ
      COMMON/ICCD/IB,JB,IBI,JBI,NAME(10),JNM,TBEG,SECREQ,DDX,DDY
     1,UCON,VCON,ERRD,ITMAX,PFAC,RRHO,XI,IT
      COMMON/INPFLD/FLO(12),UO,VO,VINT1,VINT2,UINL1,UINL2,RHOO,RINT1,
     1RINT2,RINL1,RINL2,MUO,MINT1,MINT2,MINL1,MINL2,SIEO,SIET1,SIET2,
     2SIEL1,SIEL2
      REAL MUO,MINT1,MINT2,MINL1,MINL2
      COMMON/SPECD/ETAO,FAO(7),FAT1(7),FAT2(7),
     1FAL1(7),FAL2(7),SM(7),EP(7),GAMA(7),CV(7),ALPHA(7),AM(7,3),
     2BM(7,3),Q(3),CF(3),ZETAF(3),EF(3),CB(3),ZETAB(3),EB(3)
      COMMON/OBSD/ITO(20),NSO(20),OB(6,20),OIB(2,20),UINRO(20),RINRO(20)
     1,MINRO(20),SIERO(20),VINBO(20),RINBO(20),MINBO(20),
     2SIEBO(20),FARO(7,20),FABO(7,20)
      INTEGER AM,BM
      REAL MINRO,MINBO
C**** /CELLD/ COMMON BLOCK IS LOADED USING A PRE-PROCESSOR PROGRAM(PREP)
C**** AND THE *READ FEATURE OF (UPDATE)
C**** THIS ALLOWS A VARIABLE SUBSCRIPT FOR ALL ARRAYS IN /CELLD/
C**** DATA ARE READ FROM A CARD TO SET DIMENSIONS TO IB2,JB2 AND NSPEC
C
         LCM/CELLD/FL( 62, 32),U( 62, 32),V( 62, 32
     1),RHO( 62, 32),SIE( 62, 32),P( 62, 32
     2),MU( 62, 32),CSQ( 62, 32),RHOT( 62, 32),RHOU( 62
     3, 32),RHOV( 62, 32),RE( 62, 32),ETA( 62, 32),CQ(
     4 62, 32),FA( 62, 32, 7),RJ( 32),RJ2(
     5 32),RFLB( 62),TFAB( 62, 7),TE( 62, 32)
      DIMENSION RHOUT( 62, 32),RHOVT( 62, 32),PB(
     1 62, 32),DRHOB( 62, 7),STREB( 62)
      EQUIVALENCE (RHOU,RHOUT),(RHOV,RHOVT),(P,PB),(STREB,RFLB),
     1(TFAB,DRHOB)
      INTEGER FL,CYCLE
      REAL MU
      JC=-IB2
 1500 DO 1595 J=1,JB2
      JC=JC+IB2
      DO 1590 I=1,IB2
      IJ=I+JC
      IF(FL(IJ).EQ.1)GO TO 1590
      IF(FL(IJ).GE.5)GO TO 1590
      IPJ=IJ+1
      IMJ=IJ-1
      IJP=IJ+IB2
      IJM=IJ-IB2
      IF(J.EQ.1)GO TO 1510
      IF(I.EQ.1)GO TO 1520
      IF(I.EQ.IB2)GO TO 1530
      IF(J.EQ.JB2)GO TO 1540
      GO TO 1550
C*****BOTTOM ROW
 1510 IF(FL(IJ).EQ.2)GO TO 1514
      IF(FL(IJP).EQ.5)GO TO 1514
      U(IJ)=-U(IJP) $ GO TO 1590
```

```
 45    1514 U(IJ)=U(IJP) $ GO TO 1590                                  RICE8   2713
      C*****FIRST COLUMN                                               RICE8   2714
 52    1520 IF(J.EQ.JB2)GO TO 1540                                     RICE8   2715
 54         IF(FL(IPJ).EQ.2)GO TO 1590                                 RICE8   2716
 60         IF(FL(IPJ).EQ.3)GO TO 1590                                 RICE8   2717
 63         IF(FL(IJP).EQ.5)GO TO 1590                                 RICE8   2718
 66         IF(FL(IJ).EQ.2)GO TO 1524                                  RICE8   2719
 72         V(IJ)=-V(IPJ) $ GO TO 1590                                 RICE8   2720
 75    1524 V(IJ)=V(IPJ) $ GO TO 1590                                  RICE8   2721
      C*****LAST COLUMN                                                RICE8   2722
102    1530 IF(FL(IJ).NE.4)GO TO 1536                                  RICE8   2723
106         IF(FL(IJP).NE.4)GO TO 1590                                 RICE8   2724
111    1534 V(IJ)=V(IMJ) $ GO TO 1590                                  RICE8   2725
116    1536 IF(FL(IJP).EQ.4)GO TO 1590                                 RICE8   2726
122         IF(FL(IJ).EQ.2)GO TO 1534                                  RICE8   2727
125         V(IJ)=-V(IMJ) $ GO TO 1590                                 RICE8   2728
      C*****LAST ROW                                                   RICE8   2729
131    1540 IF(FL(IPJ).EQ.5)GO TO 1590                                 RICE8   2730
135         IF(FL(IJ).EQ.2)GO TO 1544                                  RICE8   2731
140         U(IJ)=-U(IJM) $ GO TO 1590                                 RICE8   2732
144    1544 U(IJ)=U(IJM) $ GO TO 1590                                  RICE8   2733
      C*****REGULAR CELLS OF COMPUTING MESH (2,IB1) (2,JB2)****************** RICE8   2734
151    1550 IF(FL(IJ).EQ.4)GO TO 1590                                  RICE8   2735
155         IF(FL(IPJ).EQ.1)GO TO 1560                                 RICE8   2736
160         IF(FL(IMJ).EQ.1)GO TO 1570                                 RICE8   2737
163         IF(FL(IJP).EQ.1)GO TO 1580                                 RICE8   2738
166         IF(FL(IJM).NE.1)GO TO 1590                                 RICE8   2739
      C*****FL(IJ)=SOLID=2,3 AND FL(IJM)=FLUID=1                       RICE8   2740
172         IF(FL(IPJ).EQ.1)GO TO 1590                                 RICE8   2741
175         IF(FL(IPJ).EQ.5)GO TO 1556                                 RICE8   2742
200         IF(FL(IPJ).GE.4)GO TO 1590                                 RICE8   2743
203    1553 IF(FL(IJ).EQ.2)GO TO 1554                                  RICE8   2744
207         U(IJ)=-U(IJM) $ GO TO 1590                                 RICE8   2745
213    1554 U(IJ)=U(IJM) $ GO TO 1590                                  RICE8   2746
220    1556 IF(U(IPJ).EQ.0.)GO TO 1590                                 RICE8   2747
223         GO TO 1553                                                 RICE8   2748
      C*****FL(IJ)=SOLID=2,3 AND FL(IPJ)=FLUID=1                       RICE8   2749
223    1560 IF(FL(IJP).EQ.1)GO TO 1590                                 RICE8   2750
227         IF(FL(IJP).GE.4)GO TO 1590                                 RICE8   2751
232         IF(FL(IJ).EQ.2)GO TO 1564                                  RICE8   2752
236         V(IJ)=-V(IPJ) $ GO TO 1590                                 RICE8   2753
242    1564 V(IJ)=V(IPJ) $ GO TO 1590                                  RICE8   2754
      C*****FL(IJ)=SOLID=2,3 AND FL(IMJ)=FLUID=1                       RICE8   2755
247    1570 IF(FL(IJP).EQ.1)GO TO 1590                                 RICE8   2756
253         IF(FL(IJP).GE.4)GO TO 1590                                 RICE8   2757
256         IF(FL(IJ).EQ.2)GO TO 1574                                  RICE8   2758
262         V(IJ)=-V(IMJ) $ GO TO 1590                                 RICE8   2759
266    1574 V(IJ)=V(IMJ) $ GO TO 1590                                  RICE8   2760
      C*****FL(IJ)=SOLID=2,3 AND FL(IJP)=FLUID=1                       RICE8   2761
273    1580 IF(FL(IPJ).EQ.1)GO TO 1590                                 RICE8   2762
277         IF(FL(IPJ).EQ.5)GO TO 1566                                 RICE8   2763
302         IF(FL(IPJ).GE.4)GO TO 1590                                 RICE8   2764
305    1583 IF(FL(IJ).EQ.2)GO TO 1584                                  RICE8   2765
311         U(IJ)=-U(IJP) $ GO TO 1590                                 RICE8   2766
315    1584 U(IJ)=U(IJP) $ GO TO 1590                                  RICE8   2767
```

```
322    1586 IF(U(IPJ).EQ.0.)GO TO 1590                       RICE8    2768
325         GO TO 1583                                       RICE8    2769
325    1590 CONTINUE                                         RICE8    2770
330    1595 CONTINUE                                         RICE8    2771
332         RETURN $ END                                     RICE8    2772
```

☆US GOVERNMENT PRINTING OFFICE 1975—677-178/38