380
3/20/81
T.J.

(2)

Dr 2503

DOE/ET/20280-2

A PRELIMINARY USER'S GUIDE FOR THE SIGMET MESOSCALE METEOROLOGY CODE

Special Report for Period June 15, 1977—June 15, 1978

289
~~323~~
DLA-325
NTIS-25
Sf. 34
382

By
P. C. Patnaik

June 1979

Work Performed Under Contract No. AC06-77ET20280

Science Applications, Inc.
La Jolla, California

U.S. Department of Energy

Solar Energy

Available from the National Technical Information Service, U. S. Department of Commerce, Springfield, Virginia 22161.

Price: Printed Copy A07
Microfiche A01

DOE/ET/20280-2
Distribution Category UC-60

A PRELIMINARY USER'S GUIDE FOR THE SIGMET MESOSCALE METEOROLOGY CODE

SPECIAL REPORT FOR THE
PERIOD
15 JUNE 1977 TO 15 JUNE 1978

P.C. Patnaik

SCIENCE APPLICATIONS, INC.
1200 Prospect St.
La Jolla, California 92038

June 1979

PREPARED FOR THE UNITED STATES
DEPARTMENT OF ENERGY
DIVISION OF SOLAR TECHNOLOGY
FEDERAL WIND ENERGY PROGRAM

DOE Contract No. AC06-77ET20280

CONTENTS

1. INTRODUCTION

A study has been performed in recent years, as part of the Federal Wind Energy Program, to develop a new Wind Energy Conversion System (WECS) siting methodology (Freeman, et al., 1976; Traci, et al., 1977, 1978). The methodology makes use of numerical windfield models to simulate windfield "events" based on historical or field test wind data, for the purpose of identifying high wind potential sites within a region of interest. One of the significant products of the methodology development study has been a detailed physics mesoscale meteorology model embodied in the SIGMET computer code. The code simulates the transient evolution of the atmosphere in a three-dimensional mesoscale ($\sim$100. Km) region including effects of complex terrain, non-linear atmospheric stability, planetary boundary layer turbulence, and thermal forcing due to surface heat transfer and solar/terrestrial radiation. In terms of the level of physical modeling mentioned as well as aspects of the numerical solution technique, SIGMET represents an advance in the art of mesoscale meteorology predictive techniques.

As with any complex numerical model, the SIGMET code is in a continual state of testing and development; however, in its current state, the code can be used by those with experience in the ways of numerical fluid dynamic models. Hence the present document is an attempt at providing guidance in the use of the code, although it is emphasized that, in this respect, it represents a preliminary User's Guide and not a comprehensive and detailed User's Manual. Descriptive details of the formulation, testing, verification and application of the model are presented in the previous references.

In Section 2 of this User's Guide, the formulation of the physical model is summarized including the modeling approaches and approximations for each of the principal physical effects that are included. A summary of the finite difference approximations and solution technique is presented in Section 3 and additional details of the physical and numerical model are provided in the subroutine descriptions of Section 4. A summary description of the SIGMET code organization is also provided there. Finally, descriptive comments on the code usage and required input are presented in Section 5 and a sample calculation is described in Section 6. A glossary of input variables required to run the code is presented in the Appendix.

2. FORMULATION

The SIGMET mesoscale meteorology simulation code represents an extension, in terms of physical modelling detail and numerical approach, of the work of Anthes (1972) and Anthes and Warner (1974). The code utilizes a finite difference technique to solve the so-called primitive equations which describe transient flow in the atmosphere. Conservation equations for the relevant meteorological variables (wind components, temperature and moisture) are formulated in an Eulerian framework and account for advection, stratification, Coriolis effects, turbulent heat, momentum and moisture transport and radiation. In short, the SIGMET modelling contains all of the physics required to simulate the time dependent meteorology of a region with description of both the planetary boundary layer and upper level flow as they are affected by synoptic forcing and complex terrain.

In the following subsections, the mathematical formulation of the SIGMET model and the various physical effects incorporated into it are summarized.

2.1 GOVERNING EQUATIONS

In many of the applications of mesoscale simulation the atmosphere boundary layer winds are of particular interest. Over regions of complex terrain these low-level winds suffer from numerical errors when finite differences are formed in Cartesian Coordinates. In order to avoid this problem, a coordinate transformation has been carried out to a new coordinate system (the sigma, σ, coordinate system) in which the lowest coordinate surface is conformal to the terrain surface. In this representation, after care is exercised in calculating the transformed pressure gradients, the needed resolution and accuracy near the surface are achieved by concentrating the finite difference grid in the near surface region. Expanding the grid into the upper levels with a non-constant grid distribution satisfies the less stringent accuracy requirements there while minimizing vertical grid points to maintain an economical calculation.

By virtue of the assumption of hydrostatic balance the vertical coordinate z can be replaced by the dimensionless pressure coordinate σ defined as follows:

$$\sigma = \frac{P - P_T}{P_s - P_T} = \frac{P - P_T}{\pi} \quad . \tag{2.1}$$

Here P is the pressure, P_T is the pressure at the top assumed to be constant and P_s is the surface pressure which varies along the terrain.

$$\frac{dz}{d\sigma} = - \frac{\pi}{g\rho} \tag{2.2}$$

gives the relationship between the altitude and the σ coordinate. The sigma coordinate is in the range $0 \le \sigma \le 1$, and takes the value $\sigma = 1$ at the surface. It is applicable to a surface with complex terrain, as illustrated in Figure 1.

This transformation is used to eliminate the altitude. After considerable manipulation, the primitive equations take the form:

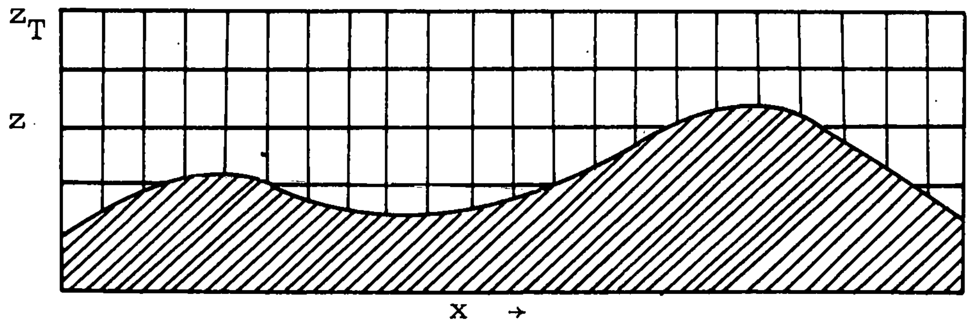
$$\frac{\partial \pi U}{\partial t} + \frac{\partial U \pi U}{\partial x} + \frac{\partial V \pi U}{\partial y} + \frac{\partial \dot{\sigma} \pi U}{\partial \sigma} - f\pi V + \pi\left(\frac{\partial \phi}{\partial x} + \frac{\sigma}{\rho}\frac{\partial \pi}{\partial x}\right) = F_V(U) + F_H(U)$$

$$\frac{\partial \pi V}{\partial t} + \frac{\partial U \pi V}{\partial x} + \frac{\partial V \pi V}{\partial y} + \frac{\partial \dot{\sigma} \pi V}{\partial \sigma} + f\pi U + \pi\left(\frac{\partial \phi}{\partial y} + \frac{\sigma}{\rho}\frac{\partial \pi}{\partial y}\right) = F_V(V) + F_H(V)$$

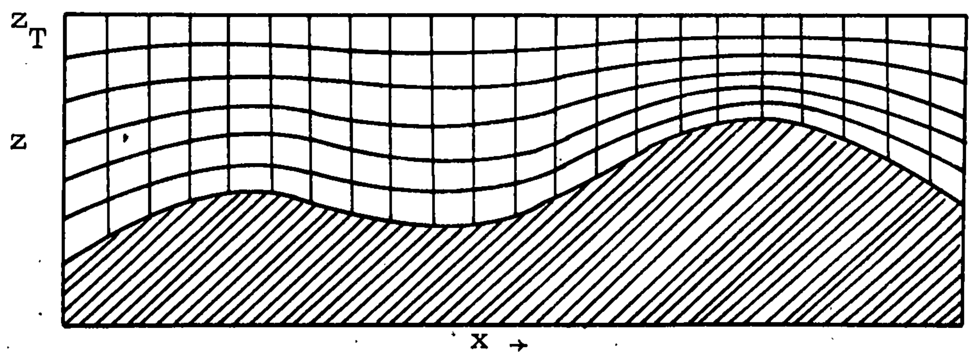
$$\frac{\partial \pi \theta}{\partial t} + \frac{\partial U \pi \theta}{\partial x} + \frac{\partial V \pi \theta}{\partial y} + \frac{\partial \dot{\sigma} \pi \theta}{\partial \sigma} = F_V(\theta) + F_H(\theta) + \pi Q$$

$$\frac{\partial \pi C}{\partial t} + \frac{\partial U \pi C}{\partial x} + \frac{\partial V \pi C}{\partial y} + \frac{\partial \dot{\sigma} \pi C}{\partial \sigma} = F_V(C) + F_H(C)$$

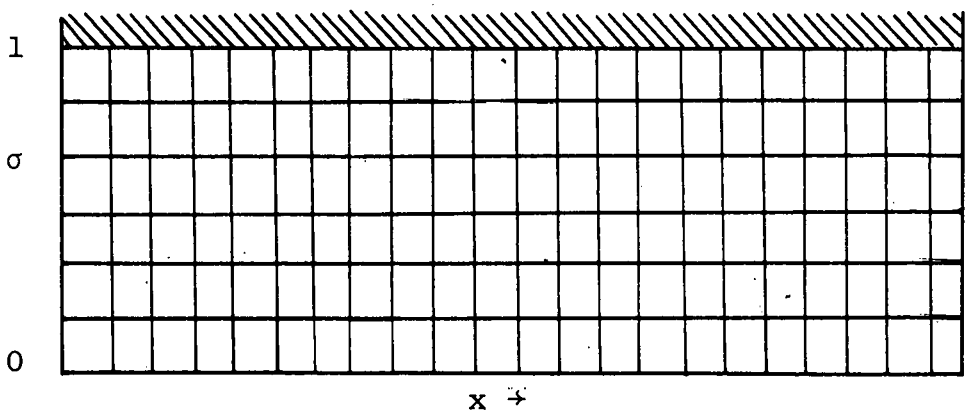
$$\frac{\partial \pi}{\partial t} + \frac{\partial U \pi}{\partial x} + \frac{\partial V \pi}{\partial y} + \frac{\partial \dot{\sigma} \pi}{\partial \sigma} = 0 \quad . \tag{2.3}$$



(a). Two-Dimensional Terrain in x,z Space.



(b). Two Dimensional Terrain in x,z Space with Contours of Constant σ Superimposed.



(c). Two-Dimensional Terrain in x,σ Space. The Location of the Terrain is indicated by the Cross-Hatched Region.

Figure 1. Coordinate Transformation

In addition, a model turbulent energy equation is solved as described below.

Here U , V and $\dot{\sigma}$ are the velocity components with reference to the x-y-σ coordinate system, $\phi = -gz$ is the geopotential height, θ is the potential temperature, C is the specific humidity and $f = 2\Omega\sin\theta_\ell$ is the Coriolis parameter (assumed constant in the SIGMET formulation using a representative latitude θ_ℓ). The effects of vertical and horizontal turbulent diffusion (to be elaborated in detail later) are contained in the terms F_V and F_H of (2.3). These are approximated by Fickian diffusion expressions:

$$F_V(\psi) = \left(\frac{g}{\pi}\right)^2 \frac{\partial}{\partial\sigma}\left(K_V\rho^2 \frac{\partial\pi\psi}{\partial\sigma}\right) \tag{2.4}$$

and

$$F_H(\psi) = \pi\left[\frac{\partial}{\partial x}\left(K_H \frac{\partial\psi}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_H \frac{\partial\psi}{\partial y}\right)\right] . \tag{2.5}$$

Note that in the above expression for F_H, lower order (in terms of surface slope) terms arising out of the transformation to the σ coordinate have been neglected. The equation for the rate of change of potential temperature θ contains a term Q through which heating due to radiative flux divergence is taken into account. Finally note that the equations are written in the so-called 'conservation' form which is preserved in the numerical scheme. By doing so mass and energy are conserved and certain nonlinear instabilities are avoided.

To complete the formulation, boundary and initial conditions must be appended to the above equation set. Considering boundary conditions first, the momentum equations are closed at the bottom boundary by employing a law-of-the-wall matching condition near the terrain surface. In addition, a surface heat balance equation is solved for given soil condition to calculate boundary values for potential temperature and moisture. At the top boundary, constant values are assumed for all quantities; this is justified by having the top boundary at sufficient height from the underlying terrain. Lateral boundary conditions are applied by distinguishing inflow (normal velocity pointing into the mesh) and outflow (normal velocity pointing away from the mesh) boundaries. On inflow boundaries all the variables are held at the ambient values (say, by interpolating from sounding measurements) thereby feeding information to the interior, whereas on the outflow boundaries the variables are extrapolated from the values inside the mesh, using a zero-gradient Neuman type boundary condition.

Initialization of all meteorological variables is performed by interpolating the available sounding data on constant σ levels. In the event of single-station data, this means the variables are assumed constant on a given σ surface. But for multi-station data weightings are assigned to individual stations. In SIGMET $1/r^2$ weighting is utilized. If there are N stations located at $(x_1, y_1), (x_2, y_2) \ldots (x_N, y_N)$, the variable $V(x, y, \sigma)$, for instance, can be initialized from the measurements $V_1, V_2 \ldots V_N$ by:

$$V(x,y,\sigma) = R^2 \sum_{n=1}^{N} \frac{V_n(\sigma)}{(x-x_n)^2 + (y-y_n)^2}$$

where

$$\frac{1}{R^2} = \sum_{n=1}^{N} \frac{1}{(x-x_n)^2 + (y-y_n)^2} .$$

This initialization scheme results in a reasonable initial field but does not guarantee a divergence-free velocity field. As a result, a transient disturbance develops which fortunately has a small lifetime.

2.2 TURBULENCE MODEL

A unique feature of SIGMET among mesoscale meteorology models is its use of a state-of-the-art turbulence model which includes the non-equilibrium effects of production (due to mean strain), dissipation (stability) and diffusion. These effects are included in the Mellor and Yamada (1975) model. This formulation, called Level 2½ makes use of a model transport equation for the turbulent kinetic energy (q^2 is twice the turbulent kinetic energy) given by:

$$\frac{\partial q^2}{\partial t} = 2 \left[S^2 (K_M - Ri K_\Theta) - q^3/\Lambda_1 \right] + \frac{\partial}{\partial z} \left(\frac{5}{3} \lambda_1 \, q \, \frac{\partial q^2}{\partial z} \right) \qquad (2.6)$$

where the gradient Richardson number is $Ri = -\frac{g}{\rho} \frac{\partial \rho}{\partial z} S^2$. The potential density ρ takes into account the effects of compressibility of the atmosphere or convective stability. The quantity S is the mean shear of the wind which enters as a production term of the turbulent kinetic energy equation above. The effects of the stability of the atmosphere are incorporated into the Richardson number: negative values, corresponding to instability, increase the turbulence, while positive values, corresponding to a stably stratified atmosphere decrease the turbulence. Diffusion coefficients K_M and K_θ, appearing in Eq. (2.6), are evaluated in terms of q^2, the Richardson number, and a length scale measured from the surface. In order to fully describe the turbulence formulation it is necessary to write equations for several intermediate quantities. These equations are obtained from the equations of motion through the process of identifying mean and fluctuating components of all independent variables and averaging over various quadratic combinations of them. Closure approximations are introduced which replace certain cubic and pressure-containing terms with quantities which are contained within the equation set for the mean variables and the variances of the fluctuations. The equations for the variances are

$$\overline{\theta^2} = - \frac{\Lambda_2}{q} \frac{\partial \theta}{\partial z} \overline{w\theta}$$

$$\overline{w\theta} = - \frac{3\ell_2}{q} \left[\overline{w^2} \frac{\partial \theta}{\partial z} - (1-C_3) \frac{g}{\theta} \overline{\theta^2} \right]$$

$$\overline{w^2} = \frac{q^2}{3} + \frac{2\ell_1}{q} \left[\overline{uw} \frac{\partial U}{\partial z} + \overline{vw} \frac{\partial V}{\partial z} + 2(1-C_2) \frac{g}{\theta} \overline{w\theta} \right]$$

$$+ \frac{4\ell_1}{3q} \frac{\partial}{\partial z} \left(q\lambda_1 \frac{\partial q^2}{\partial z} \right)$$

$$\overline{uw} = - \frac{3\ell_1}{q} \left[\left(\overline{w^2} - C_1 q^2 \right) \frac{\partial U}{\partial z} - (1-C_2) \frac{g}{\theta} \overline{u\theta} \right]$$

$$\overline{u\theta} = - \frac{3\ell_2}{q} \left[\overline{uw} \frac{\partial \theta}{\partial z} + \overline{w\theta} \frac{\partial U}{\partial z} \right]$$

$$\overline{vw} = - \frac{3\ell_1}{q} \left[\left(\overline{w^2} - C_1 q^2 \right) \frac{\partial V}{\partial z} - (1-C_2) \frac{g}{\theta} \overline{v\theta} \right]$$

$$\overline{v\theta} = - \frac{3\ell_2}{q} \left[\overline{vw} \frac{\partial \theta}{\partial z} + \overline{w\theta} \frac{\partial V}{\partial z} \right] \qquad (2.7)$$

where the bar above a quantity indicates that an average over fluctuations is taken. The fluctuations of the windspeed are (u,v,w) and the potential temperature fluctuation is θ. Length scales ℓ_1, ℓ_2, Λ_1, Λ_2, λ_1, are introduced in the closure approximations as per Yamada and Mellor (1975).

These equations now form a closed set which can be solved for the turbulence quantities which enter the equations for the mean quantities, U, V, θ, and C. These are the turbulent viscosity K_M (which enters the equations for U and V) and the turbulent heat conductivity K_θ (which enters the equations for temperature and water vapor mixing ratio) for turbulent transport in the vertical direction. These coefficients are defined as follows:

$$
\begin{aligned}
K_M &= - \overline{uw}/\frac{\partial U}{\partial z} = - \overline{vw}/\frac{\partial V}{\partial z} \\
K_\theta &= - \overline{w\theta}/\frac{\partial \theta}{\partial z} = - \overline{wc}\ \frac{\partial C}{\partial z} \quad .
\end{aligned}
\tag{2.8}
$$

Algebraic reduction of Eqs. (2.7) and (2.8) results in:

$$
\begin{aligned}
K_\theta &= \frac{3\ell_2\overline{w^2}/q}{1 + \dfrac{3\ell_2\Lambda_2(1-C_3)g}{q^2\theta}\dfrac{\partial\theta}{\partial z}} \\
K_M &= \frac{3\ell_1/q\left[\overline{w^2} - C_1q^2 - (1-C_2)\dfrac{3\ell_2 g}{\theta}\dfrac{\partial\theta}{\partial z}K_H\right]}{1 + \dfrac{q\ell_1\ell_2(1-C_2)g}{q^2\theta}\dfrac{\partial\theta}{\partial z}}
\end{aligned}
\tag{2.9}
$$

in which

$$
\overline{w^2} = \frac{q^2}{3} + \frac{4}{3}\frac{\ell_1}{q}\frac{\partial}{\partial z}\left(q\lambda_1\frac{\partial q^2}{\partial z}\right)
$$

$$
- \frac{2\ell_1}{q}\left[S^2K_M + 2(1-C_2)\frac{g}{\theta}\frac{\partial\theta}{\partial z}K_\theta\right] \quad .
$$

These three equations can be solved simultaneously, to obtain K_M and K_θ in terms of q^2 and gradients of the mean variables.

The equation for the turbulent kinetic energy contains a diffusion term as well as terms describing production and dissipation. This equation is integrated numerically in an implicit fashion (upon linearization of the dissipation term) utilizing the zero gradient condition at the top of the mesh and the law-of-the-wall surface boundary condition:

$$q^2 = 6.1\ u_\tau^2$$

where u_τ is the friction velocity determined by matching the calculated velocity just off the surface to the empirical law-of-the-wall profile (see Yamada and Mellor, 1975). This method of setting q^2 at the surface was found to be quite satisfactory. A more accurate alternative to this procedure would be to resolve the laminar sub-layer which would be prohibitively expensive in computer storage and time requirements for anything but a ID calculation.

The horizontal diffusivity K_H differs from the vertical diffusivity, which is calculated using the above prescription, by a factor which depends on the atmospheric stability. In a stable atmosphere the horizontal diffusivity is relatively enhanced over the vertical, while in an unstable atmosphere the opposite is the case, although to a lesser degree. The ratio of horizontal-to-vertical diffusivity as a function of Pasquill category stability classes is given in Table 1. These categories, in turn, are

	Stability Class					
	A	B	C	D	E	F
Ratio, K_H/K_V	0.1	0.5	1.5	6	10	25

Table 1. Ratio of Horizontal and Vertical Diffusivities Versus Stability Class

related to the temperature gradient as given by Table 2. In SIGMET often rather large values for K_H are used for the sole purpose of damping out the unwanted computational modes which grow in the presence of sharp terrain features.

Stability Classification	Pasquill Categories	Temperature Change With Height (^oC/100m)
Extremely Unstable	A	< - 1.9
Moderately Unstable	B	- 1.9 to - 1.7
Slightly Unstable	C	- 1.7 to - 1.5
Neutral	D	- 1.5 to - 0.5
Slightly Stable	E	- 0.5 to 1.5
Moderately Stable	F	1.5 to 4.0
Extremely Stable	G	> 4.0

Table 2. Classification of Atmospheric Stability (According to NRC Regulatory Guide 1.23)

2.3 THERMAL ENERGY EXCHANGE MODEL

SIGMET includes a relatively sophisticated treatment of thermal energy exchange processes in the atmosphere, namely; solar and terrestrial radiation and energy exchange between the atmosphere and the surface. The model is briefly summarized here with additional details provided in the appropriate subroutine description of Section 4.2.

In the atmosphere radiative transfer plays an important role in the balance of energy budget. It generally can be separated into two categories: solar radiation and the terrestrial reradiation. Solar radiation is the primary heat source for the atmosphere. In the spectral range, solar radiation spans the so-called shortwave range from 0.15 to 4 μm, whereas terrestrial radiation consists of the longwave length from 4 μm to 50 μm.

In estimation of solar radiation two factors are generally considered: absorption by water vapor and Rayleigh Scattering by air molecules. Because of difficulty in specifying the optical spatial and temporal distribution of clouds, the optical effects of atmospheric particles have been neglected. Given the solar radiation flux at the top of the atmosphere (same as top of the mesh) which depends on the solar constant, the mean and actual distance between earth and sun and the zenith angle of the sun defined as a function of the latitude, the same is calculated in a vertical column by accounting for the amount of water vapor in that column through empirical formulae. This accounts for the absorbed part of the solar radiation S_a. The scattered part S, is estimated by taking into account the Rayleigh scattering, ground reflection and absorption in stratosphere by ozone. In SIGMET this is done again by using well established empirical relationships (Katayama, 1974).

The atmospheric long wave radiation is absorbed by water vapor, carbon dioxide and ozone. In a cloudless sky the downward and upward radiation fluxes, taking into account these absorptions, are given through integrated relationships in the spectral frequency domain. Using the boundary conditions that the downward flux is zero at the top of the atmosphere and the upward flux at the earth's surface is that given by black body radiation at the surface temperature, the net terrestrial radiation flux R_z is estimated. Further correction terms to account for temperature discontinuity between the earth's surface and the air next to it are also included.

The divergence of the sum of solar and terrestrial flux F gives the radiative contribution to the potential temperature equation through:

$$\pi Q = - \frac{\pi}{C_p \rho} \frac{\partial F}{\partial z} = \frac{g}{C_p} \frac{\partial F}{\partial \sigma} \tag{2.10}$$

or in terms of the quantities defined above:

$$\pi Q = \frac{g}{C_p} \frac{\partial}{\partial \sigma} \left(R_Z - S_s - S_a \right) \quad . \tag{2.11}$$

Finally, a few notes and comments about the treatment of thermal forcing terms in SIGMET are mentioned. Because of the small ratio of vertical to horizontal zone size in the mesoscale calculations both two- and three-dimensional calculations are approximated by one-dimensional (vertical column) radiation calculation with negligible error. The equation derived for the one-dimension model can still be used in multi-dimensions with special care in treating water vapor content. In multi-dimensions, the effective water vapor content is integrated along radial directions towards the sun.

Since the radiation calculation is very time consuming and expensive, proper care should be exercised to determine if radiation is essential to the problem under study. It contributes the driving force for problems such as diurnal cycle, land-sea breeze, slope wind, etc.; but for problems in which terrain-induced dynamics play an important role and which attain a steady state relatively quickly the radiation calculation should be avoided.

The surface heat balance equation can be compactly written as

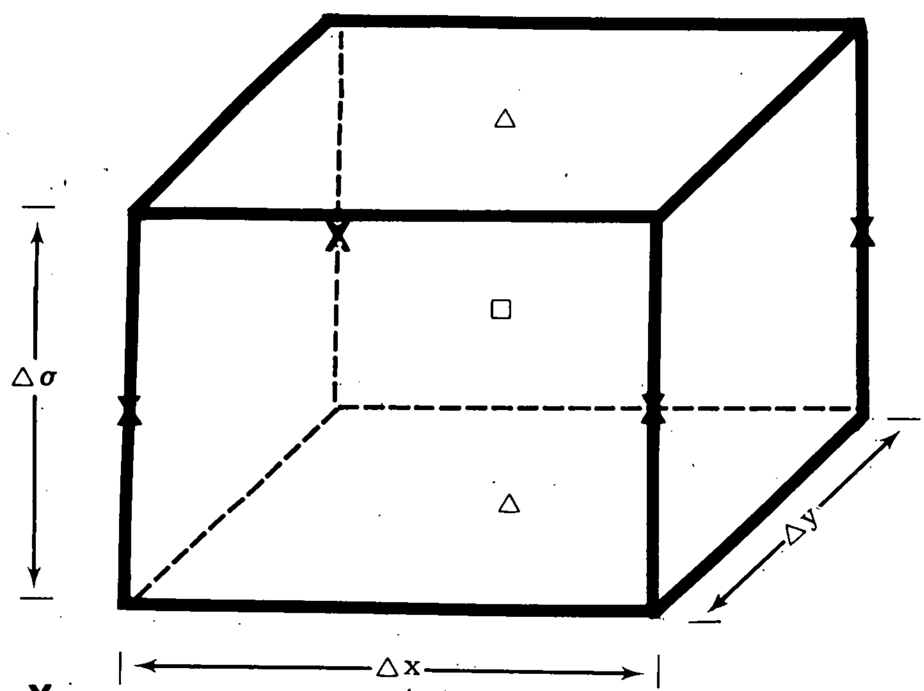
$$R - S + H + LE + G = 0 \quad , \tag{2.12}$$

where R is the terrestrial radiation flux, S is the solar radiation flux, H is the sensible heat flux, LE is the latent heat flux and G is the surface heat flux into the soil. In SIGMET both H and LE are formulated to incorporate logarithmic dependence of windspeed on height and constant flux near the ground. The component G is expressed by performing a transformation or stretching of the distance coordinate into the soil. The above heat balance equation then becomes a nonlinear equation involving the surface temperature T_g and surface vapor content C_g . In SIGMET this nonlinear equation is very efficiently solved by combining this equation with the temperature and water vapor equation in the atmosphere and solving the equation set implicitly. Through H, LE and G account is taken of the detailed characteristics of the soil, such as its composition, roughness, deep soil temperature, surface vegetation, etc.

3. NUMERICAL SOLUTION PROCEDURE

The governing differential equations, presented above, are evaluated numerically by approximating them according to a finite difference procedure. The difference equations which result consist of a set of algebraic equations for the values of horizontal velocity components, potential temperature, moisture and turbulent kinetic energy at an array of spatial points covering the three-dimensional volume that represents the mesoscale region. Based upon values at one instant in time, the solution of this equation set results in updated quantities at the array of points for a small time interval later. Thus, starting with prescribed initial conditions, the transient evolution of the meteorological variables in the solution domain is developed. The finite difference solution procedure used is now described.

In the procedure used in SIGMET, the horizontal spatial interval, Δx or Δy , is taken to be uniform in either the x or y direction but with $\Delta x \neq \Delta y$ in general. The velocity components are attributed to locations different from those for the water vapor concentration, the temperature and the geopotential. Thus, a staggered grid is used in the horizontal plane which associates the latter to the centers of grid rectangles and the velocity components to their corners. In the vertical direction, the spatial interval is specified in terms of the σ coordinate which measures the fraction of the pressure interval between the top of the mesh and the surface. Values of σ are given a nonuniform distribution, which typically results in higher resolution near the surface. In this manner, the region of larger shear near the surface is more accurately resolved, while the region near the top of the mesh has lower resolution. In the vertical plane the variables are also staggered. The staggering is achieved by defining: horizontal velocity components at the center of the vertical sides of the grid rectangle: the temperature, water vapor concentration and geopotential at the center of the rectangle; $\dot{\sigma}$, q^2 at the center of the top and bottom sides (vertical interfaces) of the rectangle. The complete staggering is shown schematically in Figure 2 for a three-dimensional grid box. The vertical extent of the calculation is typically a few kilometers, which is sufficient to contain the disturbed airflow in the mesh.



X = u, v, πu, πv points

△ = $\dot{\sigma}$, q^2 points

□ = π, T, C, φ points

Figure 2. A Three-Dimensional Grid Element

The time integration is performed by the leapfrog differencing method, in which the variables are advanced by a double time interval. This method requires the storage of variables at two times simultaneously. The leapfrog scheme is advantageous, because the rate terms corresponding to advection and pressure gradient are evaluated efficiently at the center of the double time interval, thereby insuring second-order accuracy of the time integration. The diffusion terms, by virtue of requirements for numerical stability, are evaluated in a different way. The vertical component which accounts for rapid diffusion through the lower part of the boundary layer, would require a small time interval for a stable explicit integration. To avoid such a costly restriction, an implicit integration has been implemented. This method uses variables at the advanced time, resulting in simultaneous tri-diagonal equations that are solved by two vertical sweeps for each column of zones. The horizontal explicit diffusion stability condition is less restrictive and is evaluated in terms of quantities at the earliest time. The resulting set of equations is conditionally stable, with the most severe stability criterion determined by the external gravity wave propagation. This stability condition is

$$\Delta t \leq \frac{\min(\Delta x\ ,\ \Delta y)}{2c}$$

where $c = \sqrt{gH}$ is the gravity wave speed, determined by the vertical extent (H) of the mesh and the gravitational constant (g). In a typical SIGMET calculation of 25 zones in both x and y direction with $\Delta x = \Delta y \approx 1500.$Km , and a vertical extent of 3000.m, the gravity wave stability condition limits the time step to $\sim$ 7. seconds.

In developing the finite difference expressions, the notation of Shuman and Stackpole (1968) is used, whereby horizontal differences and averages become:

$$\alpha_x = (\alpha_{i+\frac{1}{2},j} - \alpha_{i-\frac{1}{2},j})/\Delta x$$

$$\alpha_y = (\alpha_{i,j+\frac{1}{2}} - \alpha_{i,j-\frac{1}{2}})/\Delta y$$

$$\overline{\alpha}^x = (\alpha_{i+\frac{1}{2},j} + \alpha_{i-\frac{1}{2},j})/2$$

$$\overline{\alpha}^y = (\alpha_{i,j+\frac{1}{2}} + \alpha_{i,j-\frac{1}{2}})/2 \qquad (3.1)$$

$$\tilde{\alpha}^x = \overline{\overline{\alpha}^x}^x = (\alpha_{i+1,j} + 2\alpha_{i,j} + \alpha_{i-1,j})/4$$

$$\tilde{\alpha}^y = \overline{\overline{\alpha}^y}^y = (\alpha_{i,j+1} + 2\alpha_{i,j} + \alpha_{i,j-1})/4 \ . \qquad \text{(3.1 cont'd)}$$

For vertical differences and averages, the following are used:

$$\delta\alpha = (\alpha_{k+\frac{1}{2}} - \alpha_{k-\frac{1}{2}})$$

$$\overline{\alpha}^\sigma = (\alpha_{k+\frac{1}{2}} + \alpha_{k-\frac{1}{2}})/2 \ . \qquad (3.2)$$

For the temporal differences and averages,

$$\alpha_t = (\alpha^{n+\frac{1}{2}} - \alpha^{n-\frac{1}{2}})/\Delta t$$

$$\overline{\alpha}^t = (\alpha^{n+\frac{1}{2}} + \alpha^{n-\frac{1}{2}})/2 \qquad (3.3)$$

$$\overline{\alpha}^{2t} = (\alpha^{n+1} + \alpha^{n-1})/2$$

are used. In the above, i , j , and k are the horizontal and vertical grid indices, n is the time index and Δx , Δy , $\Delta\sigma$ and Δt are finite difference increments in two horizontal directions, in vertical direction and in time, respectively.

Using the above compact notation the finite difference equations approximating the differential equations are written as:

$$\overline{\pi u}^t_t = -\left(\overline{u}^x \ \overline{\tilde{\pi}\tilde{u}^y}^x\right)_x - \left(\overline{u}^y \ \overline{\tilde{\pi}\tilde{v}^x}^y\right)_y - \delta\left(\overline{\dot{\sigma}}^{xy} \ \overline{\pi u}^\sigma\right)/\Delta\sigma$$

$$+ f\pi v - \overline{\pi}^{xy}\left[\left(\frac{\sigma}{\overline{\rho}^{xy}}\right)\overline{\pi}^y_x + \overline{\phi}^y_x\right] \qquad (3.4)$$

$$\overline{\pi v}^{t}_{t} = -\left(\overline{v}^{x}\ \overline{\overline{\pi u}^{y}}^{x}\right)_{x} - \left(\overline{v}^{y}\ \overline{\overline{\pi v}^{x}}^{y}\right)_{y} - \delta\left(\overline{\dot{\sigma}}^{xy}\ \overline{\pi v}^{\sigma}\right)/\Delta\sigma$$

$$- f\pi u - \overline{\pi}^{xy}\left[\left(\frac{\sigma}{\overline{\rho}^{xy}}\right)\overline{\pi}^{x}_{y} + \overline{\phi}^{x}_{y}\right] \qquad (3.5)$$

$$\overline{\pi\theta}^{t}_{t} = -\left(\overline{u}^{y}\ \overline{\pi\theta}^{x}\right)_{x} - \left(\overline{v}^{x}\ \overline{\pi\theta}^{y}\right)_{y} - \left(\delta\ \dot{\sigma}\ \overline{\pi\theta}^{\sigma}\right)/\Delta\sigma + \pi Q \qquad (3.6)$$

$$\overline{\pi C}^{t}_{t} = -\left(\overline{u}^{y}\ \overline{\pi C}^{x}\right)_{x} - \left(\overline{v}^{x}\ \overline{\pi C}^{y}\right)_{y} - \delta\left(\dot{\sigma}\ \overline{\pi C}^{\sigma}\right)/\Delta\sigma \qquad (3.7)$$

$$\overline{\pi}^{t}_{t} = -\overline{\pi u}^{y}_{x} - \overline{\pi v}^{x}_{y} - \delta\left(\pi\dot{\sigma}\right)/\Delta\sigma \qquad (3.8)$$

The diffusion terms, which for brevity are omitted in the first four finite difference equations, have the following common form.

$$\pi\left[\left(K_{H}\psi_{x}\right)^{n-1}_{x} + \left(K_{H}\psi_{y}\right)^{n-1}_{y}\right] + \left(\frac{g}{\pi}\right)^{2}\delta\left[K_{V}\rho^{2}\ \delta\left(\pi\psi\right)^{n+1}/\Delta\sigma\right]/\Delta\sigma \qquad (3.9)$$

where $\psi = (u\ ,\ v\ ,\ \theta \text{ or } C)$ and π , ρ are centered properly depending on the diffusing quantity.

As discussed previously, lagging and leading time level values are used for stability and accuracy. It is noted that, allowing for variable $\Delta\sigma$, the vertical differences and averages presented above are modified to account for altitude-dependent weighting functions and variable $\Delta\sigma$. Also, the four-point averaging operator for the nonlinear convective term in the momentum equations is used to conserve kinetic energy (Anthes, 1972) in the numerical solution.

A set of necessary and sufficient boundary conditions and appropriate initial conditions are required to complete the problem definition. The finite difference procedure mimics the mathematical statement of these conditions mentioned in Section 2, i.e.: ambient flow properties are prescribed on the upwind lateral surfaces, a zero gradient or outflow condition is utilized on the downwind boundary, $\dot{\sigma} \equiv 0$ is prescribed on the top boundary of the computing mesh and finally a law-of-the-wall matching condition is used for the horizontal velocity components at the bottom surface.

4. PROGRAM ORGANIZATION AND DESCRIPTION

As noted above, the SIGMET computer program implements the finite difference solution procedure to solve the combined set of governing equations and boundary conditions as described in the previous sections. The program is written in standard Fortran coding for specific application on a Control Data Corporation 7600 mainframe computer. The logical organization of the program is summarized below followed by a narrative description of the function of each subroutine.

4.1 SIGMET CODE ORGANIZATION

SIGMET is written in modular form, through the use of subroutines, with each module representing an aspect of the physical model. There is thus a close correspondence between the subroutine structure and the individual physical processes being simulated. The logical flow of the program is controlled by subprogram MAIN and proceeds through the subroutines as indicated in the summary flowchart of Figure 3.

The numerical time and 3D space dependent solution of the system of equations proceeds sequentially through the subroutines indicated in the flow diagram. The solution state is advanced forward in time in a step-by-step fashion by the execution of the subroutines and thereby mimicks the actual evolution of the atmospheric properties. One computational cycle advances all of the state variables by a small increment corresponding to the intensity of the physical processes calculated at the time in question. The increment in time represented by one cycle is quite small so that the rates of change of the atmospheric variables may be considered constant over the interval. Thus, beginning with an initial condition state, the desired solution evolves iteratively in response to physical processes and applied boundary conditions.

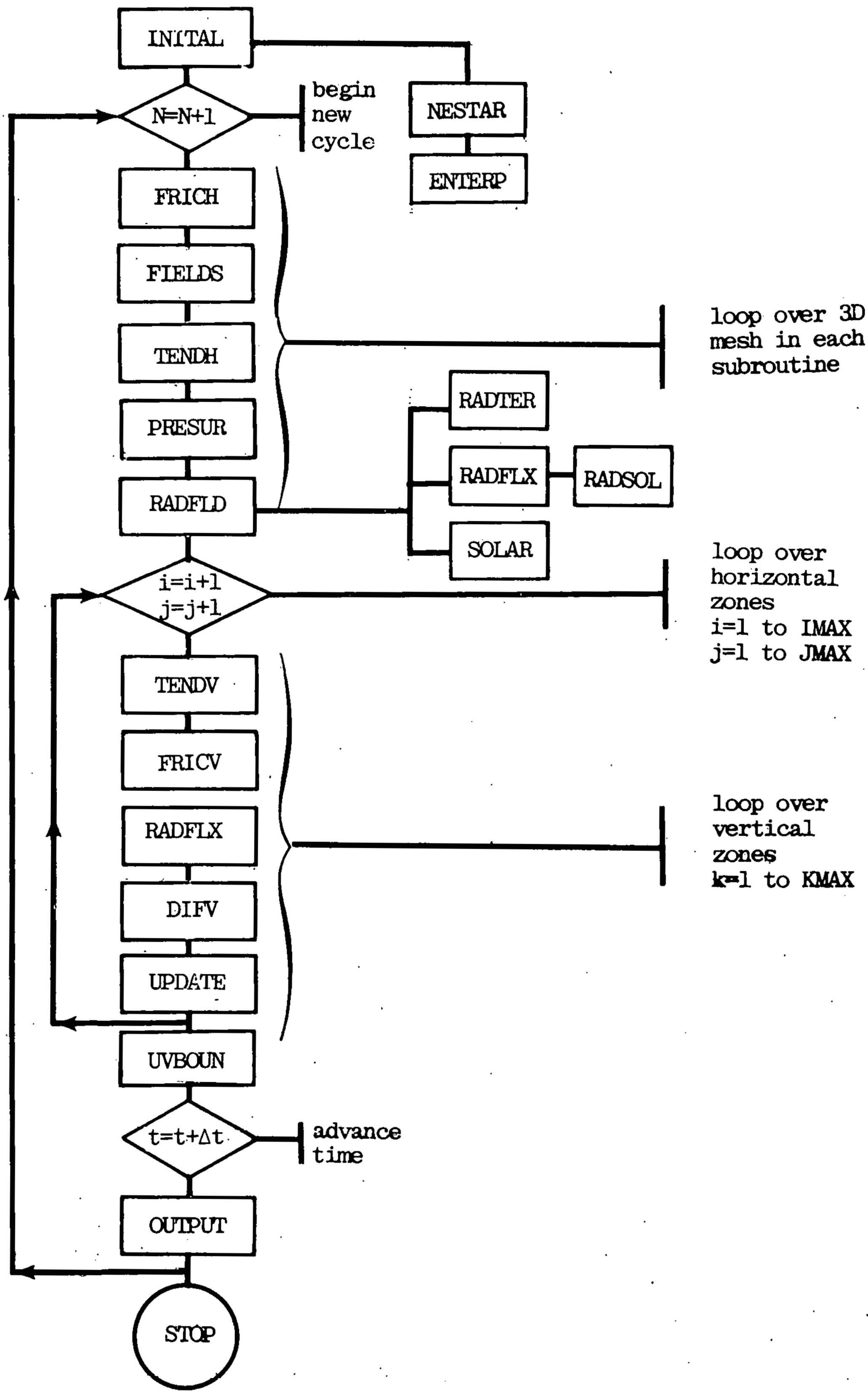


Figure 3. SIGMET Logical Flow Diagram (All Subroutines Controlled and Called by MAIN)

Within each cycle, a sequence of computational steps is performed to advance the solution through one time interval. These steps are performed in the subroutines indicated in Figure 3 and in the order shown. Because of the solution's explicit nature, the order of execution of some of these routines is not important; the order has been arranged primarily for programming simplicity and for reducing the amount of computer storage required to contain the current values of the state variables. Because of the sequential nature of the solution it is not necessary to store these variables as a function of time. Only instantaneous values are required to continue the solution and information regarding the solution state at any time is provided by periodic edits of these values.

4.2 SUBROUTINE DESCRIPTION

The subroutines that affect the numerical solution are now described in some detail. This includes a description of the function and operation of each subroutine as well as some additional explanatory comments concerning the physical model and finite difference technique.

4.2.1 INITAL

The INITAL Subroutine processes data, as provided by data cards, so that all information required for subsequent normal execution of a computation cycle is available. The data cards provide values for the following quantities:

- spatial increments of the horizontal and vertical distance
- time intervals
- number of cycles of calculation
- terrain-height array
- latitude and longitude of the region
- number of parameters characterizing the terrain such as volumetric fractions of clay, sand and organic material in the sand and organic material in the terrain, roughness height, vegetation fraction, etc.

- a number of parameters characterizing the solar and terrestrial radiation through the atmosphere
- tables in pressure coordinates describing the vertical profiles of the ambient values of horizontal velocity components, temperature and amount of water vapor in the atmosphere
- initial surface temperature
- initial surface wetness and
- indicators controlling the output of information.

Options are provided so that some of the read-in quantities can be overwritten and suitable data generated internally in INITAL. For example, the vertical coordinate σ can be generated using a log-linear formula or a geometric progression series resulting in smooth nonuniform distribution of σ levels. As another example, the ambient temperature table can be constructed using a temperature near the surface, two lapse rates and an inversion pressure height. The surface temperature can be defined using the temperature table and the surface pressure and the water vapor concentration can be calculated from relative humidity values instead of being read-in directly.

The initial data are processed so that they are acceptable to the other subroutines of SIGMET. In some cases, this involves conversion of units from those in which the input data are commonly available to those consistent with the formulation. In addition, a number of auxiliary quantities are formed that are used during the calculation but do not change with time. Precalculation of these quantities saves computing time. In other cases, the data are interpolated to provide values at locations specified by the computational grid. This second function is performed in a subroutine called NESTAR in which such quantities as the pressure thickness of the part of the atmosphere for which calculation is performed, the horizontal velocity components, the temperature and the initial water vapor concentration (if any) are formed. These quantities are assigned to the appropriate storage arrays.

4.2.2 NESTAR

In NESTAR a pressure surface is calculated corresponding to a prespecified constant boundary-layer height above the local terrain. Above this surface, the values of the horizontal velocity components are interpolated from the tables; below the surface, the values are computed from a power-law profile. Temperature and moisture are interpolated from this table at all levels. All the interpolation is done by calling the subroutine function ENTERP. Subroutine NESTAR also defines initial values of the turbulent kinetic-energy arrays using the properties of the ambient atmosphere and the "level 2½" formulation of Yamada-Mellor turbulence prescription.

4.2.3 FRICH

The FRICH Subroutine performs two functions: horizontal diffusion coefficients are calculated as functions of the local state variables, and using these calculations the turbulent horizontal diffusion taking place in one time interval is calculated. The result of these steps is to evaluate the contribution of horizontal diffusion to the rates of change of all of the primary variables.

The horizontal diffusion coefficients are obtained from Smith and Howard (1972) and are incorporated into FRICH in the form of tables. Entries of the main table are values of the diffusion coefficient as a function of altitude for each of the standard stability categories according to the AEC classification. A factor accounting for the dependence on stability of the ratio of horizontal-to-vertical-diffusion coefficient is also applied.

Because the horizontal diffusion is a slow process, the calculation can be performed explicitly without regard for the time interval. Consequently, the second task of the FRICH Subroutine is to perform a calculation of the increment to the primary variables, u , v , T and C , resulting from horizontal diffusion. Numerical stability of this diffusion calculation applied to the leapfrog method requires that a special treatment of the time centering be used. If this term were the only one being calculated, a typical equation would be evaluated as follows:

$$\frac{\phi_{ij}^{n+1} - \phi_{ij}^{n-1}}{2\Delta t} = \tag{4.1}$$

$$K_H \left[\frac{\phi_{i+1,j}^{n-1} - 2\phi_{ij}^{n-1} + \phi_{i-1,j}^{n-1}}{\Delta x^2} + \frac{\phi_{i,j+1}^{n-1} - 2\phi_{ij}^{n-1} + \phi_{i,j-1}^{n-1}}{\Delta y^2} \right]$$

which has the stability requirement

$$\Delta t \leq \frac{1}{4K_H} \left(\frac{1}{\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}} \right) \tag{4.2}$$

The time centering of the right-hand term is not permitted, because such a scheme is numerically unstable. Because of the large horizontal-space increments, the stability condition given by Eq. (4.2) is considerably less stringent that that for gravity waves to be discussed below.

4.2.4 FIELDS

The FIELDS Subroutine is executed after FRICH at the beginning of each cycle to calculate auxiliary quantities, including the two components of horizontal velocity, temperature, and moisture. These quantities are obtained from the primary variables, πu , πv , $\pi\Theta$, and πC , where $\pi = P_s - P_t$ is the pressure thickness of the computational region, and Θ is the potential temperature. By placing this subroutine after FRICH, storage of u , v , Θ and C at two time levels is avoided. FRICH uses the old values of these variables for stability and then FIELDS updates them to current time.

4.2.5 TENDH

In TENDH, the contribution of horizontal advection (the process of moving a particular property of the material from one zone to another by the horizontal-velocity components) to the rate of change is calculated. This process depends only on the horizontal components of velocity as indicated by the following advection equation:

$$\frac{\partial \pi\phi}{\partial t} - \frac{\partial \pi u\phi}{\partial x} - \frac{\partial \pi v\phi}{\partial y} \qquad (4.3)$$

The terms in Eq. (4.3) are evaluated explicitly in conservative form. The difference equation is consistent with the leapfrog time differencing and staggered spatial grid of Figure (2). In finite difference notation, the advection terms are written:

$$\begin{aligned} \overline{\pi U}^{t}_{t} &= -\left(\overline{U}^{x}\ \overline{\widetilde{\pi u}^{x}}^{y}\right)_{x} - \left(\overline{U}^{y}\ \overline{\widetilde{\pi v}^{x}}^{y}\right)_{y} \\ \overline{\pi\phi}^{t}_{t} &= -\left(\overline{u}^{y}\ \overline{\pi\phi}^{x}\right)_{x} - \left(\overline{v}^{x}\ \overline{\pi\phi}^{y}\right)_{y} \end{aligned} \qquad (4.4)$$

where $\Pi = (u\ ,\ v)$ and $\phi = (0\ ,\ C)$. The particular form of the differencing is dictated by the desire for a relatively simple conservative form that is accurate to second order and is computationally stable. The advection terms are evaluated at the midlevel of the three time levels of the leapfrog scheme, thereby assuring second-order time accuracy.

Stability analysis indicates that Eq. (4.4) is stable with no damping when the following inequality is observed:

$$\Delta t \le \frac{1}{\frac{u}{\Delta x} + \frac{v}{\Delta y}} \qquad (4.5)$$

Typically this stability condition is much less restrictive than the gravity-wave condition.

Boundary conditions at the lateral mesh surfaces are also incorporated into the TENDH routine. These conditions account for the ambient atmospheric properties that enter the calculational region from upwind and the removal of the disturbed quantities from the mesh at the downwind boundary. These boundary conditions are implemented by prescribing ambient values on the surfaces that have a wind component directed into the mesh and by extrapolating values from the mesh interior to the boundary on those surfaces that have wind components directed out of the computational region.

4.2.6 PRESUR

In the PRESUR Subroutine, pressure gradient effects are taken into account, and rates of change of the horizontal-velocity components corresponding to them are formed. In the leapfrog equations, centered as indicated in Figure 2, the pressure gradients can be calculated to second-order accuracy in both space and time without greatly complicating the equations. The calculation is performed explicitly (i.e., the gradient is evaluated from known quantities evaluated at the current cycle). Corresponding to this explicit formualtion, an explicit stability condition exists, which severely limits the time interval. This gravity-wave-propagation condition is, in fact, the most restrictive of all the time-interval constraints and typically limits the time interval to a few seconds. The stability condition is given by

$$\Delta t \leq \frac{\min\ (\Delta x\ ,\ \Delta y)}{2\sqrt{gH}}\ , \tag{4.6}$$

where H is the effective thickness of the computational region, and g is the acceleration of gravity.

The terms of interest in the PRESUR Subroutine give rise to a rate of change of the horizontal-velocity components. The differential equations are

$$\frac{\partial \pi u}{\partial t} = -\pi\left(\frac{\partial \phi}{\partial x} + \frac{\sigma}{\rho}\frac{\partial \pi}{\partial x}\right)$$

$$\frac{\partial \pi v}{\partial t} = -\pi\left(\frac{\partial \phi}{\partial y} + \frac{\sigma}{\rho}\frac{\partial \pi}{\partial y}\right) \qquad (4.7)$$

where $\phi = gz = \frac{\pi}{g}\int_{\sigma}^{1}\frac{d\sigma}{\rho} + gz_s$ is the geopotential, and z_s is the surface altitude.

The above terms are approximated by the difference equations

$$\overline{\pi u}_t^t = -\overline{\pi}^{xy}\left[\frac{\sigma}{\overline{\rho}^{xy}}\,\overline{\pi}_x^y + \overline{\phi}_x^y\right]$$

$$\overline{\pi v}_t^t = -\overline{\pi}^{xy}\left[\frac{\sigma}{\overline{\rho}^{xy}}\,\overline{\pi}_y^x + \overline{\phi}_y^x\right] \qquad (4.8)$$

The terms on the right-hand side of Eq. (4.8) are evaluated from quantities available at the current time and involve centered spatial differences which results in second-order accuracy in both space and time.

The calculation in PRESUR is carried out in two steps: first the geopotential height is formed by integration of the hydrostatic equation; second, the contribution of pressure terms to the momentum's rate of change is formed. A correction term, formed on the first time cycle, is applied to each term throughout the calculation. As seen in Eq. (4.7) the pressure gradient is the sum of several terms. When terrain is present, each term is large and should cancel each other when the atmosphere is at rest. However, because of the finite-difference approximation, these terms do not cancel exactly, giving rise to a spurious acceleration. The correction term eliminates this error, thereby increasing the accuracy of the dynamical calculation.

4.2.7 RADFLD

This subroutine is accessed to account for the solar and terrestrial radiation through the atmosphere. Because the amount of time spent in terrestrial radiation is about one-third of the total computational time, this subroutine is called only at a preset interval to update the radiation fluxes. At other times an extrapolation procedure is used to approximate the radiation fluxes.

RADFLD calls subroutine SOLAR, which determines various sun related quantities such as: solar constant for current earth-sun distance, current declination of the sun, current hour angle, zenith angle, normal insolation and other miscellaneous quantities. RADFLD then computes the amount of water vapor in a given vertical column taking into account the amount above the column that is outside the mesh (calculated using QEFFG). Subroutine RADTER, which determines the contribution to atmospheric heating from longwave radiation, is then called. The prescription used is that of Katayama (1974) as included in the 3-level UCLA GCM. The equations of radiative transfer are solved subject to the following boundary conditions: downward IR flux is zero at the top of the mesh and the upward flux at the earth's surface is black body at the surface temperature. The solution utilizes a bulk transmission function defined as the product of water vapor and carbon dioxide transmission functions. Absorption by ozone is neglected and the transmission function is assumed linear. RADSOL is then called from RADFLX, which calculates the radiant flux within the atmosphere due to solar insolation. A one-dimensional model is used, which takes into account: absorption of solar radiation by water vapor and the Rayleigh scattering by air molecules.

4.2.8 TENDV

As indicated in Figure 3, several subroutines occur within a do-loop ranging over all of the horizontal zones. These subroutines form quantities in which the coupling is in the vertical direction, and are easily and efficiently calculated a column at a time.

The first of the calculations in the TENDV Subroutine takes care of the vertical advection and the rate of change of the surface pressure. The vertical advection terms contain the vertical velocity ($\dot{\sigma}$), which is the effective velocity of the fluid with respect to the σ-coordinate. This quantity is obtained by employing the mass conservation equation subsequent to the calculation of the pressure-tendency rate.

The first calculation is that of the rate of change of pressure. This quantity is obtained by summing the pressure equation over all of the vertical zones. The as-yet-unknown- $\dot{\sigma}$ terms drop out of the equation in this procedure.

$$\bar{\pi}_t^t = - \sum_k \left(\overline{\pi u}_x^y + \overline{\pi v}_y^x \right) \Delta\sigma_k \qquad (4.9)$$

where the horizontal velocity divergence is formed from quantities known at the current time. These terms are centered in time and space, so that the pressure tendency is second-order accurate.

The second calculation in the TENDV Subroutine is that of $\dot{\sigma}$. This calculation employs the following equation in a sequential fashion.

$$\dot{\sigma}_{k+1} = \dot{\sigma}_k - \left(\bar{\pi}_t^t + \overline{\pi u}_x^y + \overline{\pi v}_y^x \right) \frac{\Delta\sigma_k}{\pi} \qquad (4.10)$$

The first term $\dot{\sigma}$ (k=1) is zero from the top-boundary condition. The $\dot{\sigma}$ values obtained in this step are used subsequently to evaluate the effects of vertical advection on all of the primary variables. This calculation can be considered to be an explicit one, since the velocities entering Eqs. (4.9) and (4.10) are known quantities at the current cycle.

TENDV calculates the contribution of vertical advection to u , v , θ , and C using the following relations respectively:

$$
\begin{aligned}
\overline{\pi u}^{t}_{t} &= -\delta\left(\overline{\dot{\sigma}}^{xy}\ \overline{\pi u}^{\sigma}\right)/\Delta\sigma \\
\overline{\pi v}^{t}_{t} &= -\delta\left(\overline{\dot{\sigma}}^{xy}\ \overline{\pi v}^{\sigma}\right)/\Delta\sigma \\
\overline{\pi\theta}^{t}_{t} &= -\delta\left(\dot{\sigma}\ \overline{\pi\theta}^{\sigma}\right)/\Delta\sigma \\
\overline{\pi C}^{t}_{t} &= -\delta\left(\dot{\sigma}\ \overline{\pi C}^{\sigma}\right)/\Delta\sigma
\end{aligned}
\tag{4.11}
$$

Finally, Coriolis contributions to the horizontal momentum equations are calculated in a straightforward manner:

$$
\begin{aligned}
\frac{\partial \pi u}{\partial t} &= f\ (\pi v - \pi v_g) \\
\frac{\partial \pi v}{\partial t} &= -f\ (\pi u - \pi u_g)
\end{aligned}
\tag{4.12}
$$

The components for the geostrophic velocity u_g and v_g are evaluated by interpolation from the ambient velocity table.

4.2.9 FRICV

In preparation for the calculation of the vertical diffusion resulting from turbulence, the boundary-layer diffusivities are calculated in the FRICV Subroutine. They will be used subsequently in DIFV to add the effects of turbulent transport to the tendency equations. The eddy diffusivities for momentum and heat are calculated according to the '2½ level' prescription of Mellor and Yamada (1974). In this formulation, a system of algebraic equations involving the local Richardson number, turbulent kinetic energy and the local shear is solved in order to evaluate the eddy diffusivities. The turbulent kinetic energy is described by a differential equation, which amounts to solving a tridiagonal system of equations in finite difference space. This is done in FRICV by calling subroutine TRIDG. Kinetic energy is the only quantity, which is completely updated to future-time level in FRICV.

4.2.10 RADFLX

The subroutine RADFLD, which updates the radiation fluxes, is not called every time step for computational economy; instead, subroutine RADFLX is used to update the fluxes by performing a linear extrapolation on radiative fluxes saved at two earlier time steps. Except during sunrise and sunset hours, this procedure has been found to work well.

4.2.11 DIFV

This important subroutine serves several functions:

- The vertical turbulent diffusion calculation is performed using the diffusion coefficients formed in the preceding subroutine FRICV.
- The heat balance equation at the soil surface is solved efficiently and accurately to take into account phenomena taking place on and near the ground surface. An option also is provided for water/land surface.

As previously dicussed, the vertical turbulent diffusion is a vigorous process, which acts across the small zone thickness corresponding to the vertical zoning. An explicit treatment of this diffusion would require that an additional numerical stability inequality be observed in the determination of the time interval. This diffusion-stability criterion would be very restrictive and would cause the time interval to be limited to even smaller values than determined by the other criteria discussed above.

In order to avoid such a costly restriction of the time interval a numerical formulation of the terms involving vertical diffusion that is not subject to the stability criterion was implemented. Such a formulation is that of partial implicitization, in which a linear system of simultaneous equations is solved. While the method required to evaluate these equations is somewhat more complicated (as described below) than that for the explicit formulation, the computational penalty is minimal. The solution is formed cellwise column by column and makes use of diffusivities calculated previously for the column of cells in question. This solution sequence dictates that the vertical calculations of the FRICV and DIFV subroutines be performed in the inner-most loop of the flowchart as indicated in Figure 3.

The terms of the differential equations that are evaluated in DIFV are

$$\frac{\partial \pi \phi}{\partial t} = \frac{g^2}{\pi^2} \frac{\partial}{\partial \sigma} \left(\rho^2 K_V \frac{\partial \pi \phi}{\partial \sigma}\right) + \dot{\phi} \qquad (4.13)$$

where $\phi = (u , v , h , C)$, and $\dot{\phi}$ is a symbolic representation of the other terms in the governing equations, which have been evaluated explicitly in the subroutines discussed above. The difference formulation evaluates the diffusion terms at the advanced time of the three levels of the leapfrog scheme. This formulation has been shown to be unconditionally stable against numerical error growth. Taking into account the variation in vertical cell size with height, the difference equation corresponding to Eq. (4.13) is:

$$\frac{\pi\phi_k^{n+1} - \pi\phi_k^{n-1}}{2\Delta t} = \frac{g^2}{\pi\Delta\sigma}\left[\frac{(\rho^2 K_V)_+}{\Delta\sigma_+}\left(\pi\phi_{k+1}^{n+1} - \pi\phi_k^{n+1}\right) - \frac{(\rho^2 K_V)_-}{\Delta\sigma_-}\left(\pi\phi_k^{n+1} - \pi\phi_{k-1}^{n+1}\right)\right] \quad (4.14)$$

In Eq. (4.14) the subscript (k) indicates the vertical index, the superscript (n) indicates the time index, and the subscripts (+) and (-) designate quantities to be evaluated at the interface of the zone having larger and smaller k-indices, respectively. All of the indices indicating horizontal position, which are not involved with these terms, have been suppressed.

Eq. (4.14) is a system of linear equations for the unknown quantities $\pi\phi_k^{n+1}$, which is tridiagonal (the matrix of the coefficients of $\pi\phi_k^{n+1}$ contains nonzero elements only on the main diagonal and the bordering diagonals immediately above and below the main diagonal). This system of equations is closed with boundary conditions at the top and bottom of the computational region. At the top of the mesh, zero turbulent flux is assumed. At the bottom the velocity at the surface is matched to a wall layer solution, and the temperature boundary value at the surface is assumed to be known.

The formulation of the surface-heat balance in DIFV takes into account, 1) the sensible and latent heat fluxes to incorporate the logrithmic dependence of windspeed on height, 2) the heat flux into soil, 3) the evaporation and condensation of moisture from the soil surface, and 4) simultaneous solution of the heat balance, atmospheric temperature and water vapor equations and the soil temperature equation. The soil moisture is defined by a wetness parameter.

4.2.12 UPDATE

Subroutine UPDATE updates all the primary variables to the advanced time level. In addition, a frequency filter to the time integration (Asselin 1972) is applied at each time step in order to remove the high-frequency noise from the solution. The filter also helps to damp the $2\Delta t$ oscillations inherent in leapfrog time differencing by averaging the solution over three consecutive time steps. If the superscript (n) denotes the time level and $(\sim)$ denotes smoothed variable, then the filter is defined by the following operator:

$$\tilde{\phi}^n = \phi^n + \tfrac{1}{2}\alpha\left(\tilde{\phi}^{n-1} - 2\phi^n + \phi^{n+1}\right) \tag{4.15}$$

Note that the filter is applied only to old quantities $\left(\phi^n\right)$ after the new quantities $\left(\phi^{n+1}\right)$ are calculated. The parameter $\alpha = 0$ corresponds to no filtering and $\alpha = 1$ corresponds to a heavily biased filter. A value of $\alpha = .5$ completely suppresses $2\Delta t$ oscillations.

4.2.13 UVBOUN

Values of velocity at external positions on the computational mesh have not been supplied in the previous subroutines and are added in the UVBOUN subroutine. These quantities are obtained in accordance with whether the wind velocity is directed into or out of the computational region at the boundary location in question. If the wind is directed inward, a specified boundary value of velocity is utilized. On the other hand, if the wind is directed out of the mesh, the boundary values of the wind components are obtained by extrapolation from the interior of the mesh.

4.2.14 OUTPUT

The OUTPUT subroutine controls the manipulation of several forms of data that flow from the computer to peripheral devices. This output may take the form of printer edits, tape dumps, printer plots or film plots. The user determines the frequency and mode of data output by specifying parameters in the INITAL subroutine. These parameters specify how many cycles of calculation are to elapse between the successive edits of a specified type.

The printer edits provide the user with a record of the progress of the calculation in terms of the values of selected variables. These variables are chosen to provide maximum information about the quantities of greatest interest during the normal course of the calculation. More complete data output is obtained by requesting the so-called "debug" edits. These edits contain a large amount of information on intermediate quantities, which are normally not edited. They would be too expensive to print, too time consuming to inspect, and too cumbersome to archive. The debug edits find their greatest use in diagnosing abnormal behavior of a calculation by providing detailed information on the suspected quantities.

The tape dumps are transfers to mass storage of all of the variables required to start the calculation. This feature is provided to enable the user to reinitiate a calculation, to change data part way through a calculation, or to edit information about a calculation selectively or in greater detail. These dumps can be stored compactly for a long period of time and thus constitute a convenient and permanent record of completed calculations.

The printer plots provide the user with various graphical representations of the data. This form of output is invaluable in evaluating the results of the calculation. The printer plots are made during the calculation and provide greater user overview of the calculation's progress and insight into the behavior and accuracy of the results.

The most important and impressive form of plot is on microfiche. Contour plots of velocity magnitude, temperature, moisture, turbulent kinetic energy, vector plot of the horizontal velocity vectors at different σ levels, contour and prospective plots of the terrain, etc., can be generated at any frequency through calls to other plot routines from OUTPUT. The plot routines are however machine and computer center specific, and would require modifications to utilize this capability.

5. USAGE AND INPUT

As might be imagined, the SIGMET code which implements all of the methods and models described above is quite sophisticated and requires knowledge and experience with such techniques to operate in a useful manner. It is being continually updated in terms of its physical and numerical methods to enhance its usefulness and efficiency. It, however, remains quite expensive to run; presently requiring on the order of 10's of minutes on a CDC 7600 computer for a typical 3D mesoscale simulation of approximately one hour real time duration. It also makes extensive use of computer storage; in its present configuration it requires 151.5_8K words to load and 130.5_8K words of small core memory to execute as well as 404.0_8K words of large core memory (LCM). In this configuration, it can perform a calculation with a finite difference grid design of (25 x 21 x 15) zones in the XxYxZ directions respectively. The sample calculation in the next section provides coding changes of array sizes required for performing a slightly larger calculation. It is recommended that similar changes be made for a smaller calculation to minimize computational costs. Because of the extensive storage and time requirements, it goes without saying that conservative computing practice be followed when exercising the SIGMET code.

Most input variables required for a SIGMET calculation are read through two NAMELIST records: OUTPT and START. This input uses standard CDC NAMELIST format. The variables to be included in each record are listed with a functional description in the Appendix. Variables in record OUTPT control execution and variables in START provide numerous parameters for numerical and physical description of a calculation. For an initial run, both records should be read, and for a restart run (ISTART = 1), when previously generated dump tape is used to initialize the calculation, only record OUTPT is read. SIGMET automatically generates a dump tape on logical Unit 8 for any run.

A typical input deck for an initial run is provided in the next section and has the following form:

NAMELIST OUTPT	: (see Appendix)
NAMELIST START	: (see Appendix)
UADAT	: (80 character field containing upper air data description)
TRDAT	: (80 character field containing terrain data description)
((HM(I,J), I=1, IE), J=1, JE)	: terrain data in format 8F 10.0)

As suggested in the above input list, terrain data must be provided for each finite difference cell starting at the southwest corner (I,J) = (1,1) for each row in turn proceeding in an easterly (I=1, IE) direction. This data is read according to the following coding in subroutine INITAL:

```
          DO 180 J = 1, JE
          READ (5,615) (HM(I,J), I = 1, IE)
      180 CONTINUE
      615 FORMAT (8F10.0)
```

As mentioned above, a restart run requires only the input contained in NAMELIST OUTPT to control execution as well as a dump tape of a previous run mounted on logical Unit 7.

A concrete example of the input for an initial run is now provided in the next section.

6. SAMPLE CALCULATION

The set up and example output for a sample SIGMET calculation are presented in this section for demonstration purposes. The test problem is the flow over a three-dimensional Guassian Hill. with a prevailing uniform westerly wind in a neutrally stratified (lapse rate = - .01 $^{O}K/m$ = - GAM) atmosphere.

Table 3, below lists the coding changes that are required to run this problem with the present code configuration. These changes include alteration in the array dimensions for various parameters and a code update to calculate the hill terrain directly rather than read the terrain data from cards. All other input parameters are defined from NAMELIST as described below. It is noted that the radiation arrays (RADTN, RADTO, RADSN, RADSO) are dimensioned to 1, since radiation is neglected in this test case (IRAD = 0). This results in a substantial reduction in computer

Table 3. Code Updates for Gaussian Hill Sample Case

```
*IDENT GAUSS
*DELETE COMMON.2,COMMON.12
      COMMON/ARRAY1/UA(25,25,15),VA(25,25,15),CA(24,24,15),TA(24,24,15)
      COMMON/ARRAY2/UB(25,25,15),VB(25,25,15),CB(24,24,15),TB(24,24,15)
      COMMON/ARRAY3/U(25,25,15),V(25,25,15),C(24,24,15),T(24,24,15)
      COMMON/ARRAY4/UTEN(25,25,15),VTEN(25,25,15),CTEN(25,25,15),
     .             TTEN(25,25,15)
      COMMON/ARRAY5/PHI(24,24,15),PHIPIX(24,24,15),PHIPIY(24,24,15)
      COMMON/ARRAY6/QA(24,24,16),QB(24,24,16),HDIFH(24,24,15)
      COMMON/ARRAY7/PSA(24,24),PSB(24,24),PTEN(24,24),HM(24,24),
     .             PUA(25,25),PUB(25,25),PCUT(25,25),CONVAR(25,25)
      COMMON/ARRAY8/TH(24,24),TG(24,24),STEMP(24,24),CG(24,24),
     .             GW(24,24),HRATE(24,24),IJCOLM(24,24)
*DELETE COMMON.15,COMMON.18
      COMMON/RADIAT/RADTN(1,1,1),RADTO(1,1,1),RADSN(1,1,1),RADSO(1,1,1)
      COMMON/SCRACH/FACTOR(24,16),FACTOL(24,16),SDOTR(24,16),
     .             SDOTL(24,16)
*DELETE COMEQV.4,COMEQV.6
      DIMENSION COM1(36030),COM2(36030),COM3(36030),COM4(37500),
     .          COM5(25920),COM6(27072),COM7(4804),COM8(4032),
     .          COM9(256),COM10(4),COM11(1536),COM12(96),
*DELETE INITAL.172,INITAL.176
      SIGC=3000.0
      DO 180 J=1,JE
      DO 180 I=1,IE
      XI=(I-12.5)*DX
      YI=(J-12.5)*DY
      HM(I,J)=1000.0*EXP(-(XI**2+YI**2)/SIGC**2)
  180 CONTINUE
```

core storage requirements. Further relief could be obtained by deleting the plotting subroutines (CONTUR, PLOTER and VELVER) and adding the dummy subroutines MODESG and EXITG since microfiche plots are not to be generated (IFILM = 0). This step should be implemented if SIGMET is to be run at a computer facility where the identical plotting package is not available.

The following are the NAMELIST input parameters and description caption input for this sample calculation:

```
$OUTPT
 NCYCLS=5,     IEDIT=5,     IOUT=0,     IPLOT=0,     ISTART=0,
 KBEG=15,      KBGPL=15,    IFILM=0,
$END

$START
 GMTMHR=7.25,        KMAX=15,            DELTAT=1.6,          DLAT=0.0,
 ANGLE=0.0,          DLONG=106.4,        PTOP=650.0,          PBOT=1000.,
 W=0.0,              IRAD=0,             XORG=0.0,            XCLAY=0.0,
 XSAND=1.0,          RPDIF=0.608,        CPDIF=2.0,           TRANS=1.0,
 VEG=0.0,            ZNAUT=0.1,          QEFFG=0.5,           GDEPTH=0.1,
 TDEEP=15.0,         TWATER=15.0,        ZBL=200.0,           SUREM=1.0,
 IE=24,              JE=24,              DX=750.0,            DY=750.0,
 TG=576*15.0,        GW=576*0.2,         HRATE=576*0.0,       IJCOLM=576*0,
 GAM=0.01,           GAM1=0.005,         PINV=500.0,          TNPRS=15.0
 NPRES=21,           DAYBEG=1.0,         ZBOT=0.0,            ALPH=0.0,
 PPRES=600.,620.,640.,660.,680.,700.,720.,740.,760.,780.,800.,
   820.,840.,860.,880.,900.,920.,940.,960.,980.,1000.,
 UPRES=21*10.0,      VPRES=21*0.0,       CPRES=21*0.0,        THETA=21*270.,
 RELHM=21*0.0,       CX3=0.013882,       CX4=1.2,
$END

SIMULATED INITIAL CONDITIONS
3-D GAUSSIAN HILL
```

If this were a usual SIGMET run in which terrain data were to be input via cards, the above input would be followed by terrain data cards in an 8F 10.0 format.

The above completes the coding changes and data input for the sample case. Table 4 is a reproduction of the raw computer output for this calculation. The numbers and tables provided there are relatively self explanatory. For illustration and check-out purposes, only the first two pages of output for the initial conditions and for the results at the end of the 5-cycle run are provided.

Table 4. Sample Output (Initial Condition)

```
XXXX  X  XXXX  X   X  XXXX XXXXX
X     X  X     XX XX  X      X
XXXX  X  X XX  X X X  XXXX   X
   X  X  X  X  X   X  X      X
XXXX  X  XXXX  X   X  XXXX   X
```

INITIAL CONDITIONS

CONSTANTS

GMTMHR=	7.2500E+00	DLAT=	0.	PTOP=	6.5000E+01	IRAD=	0	IEDIT=	5
DAYBEG=	1.0000E+00	DLONG=	1.0640E+02	PBOT=	1.0000E+02	ZBL=	2.0000E+02	IPLOT=	0
DELTAT=	1.5000E+00	ANGLE=	0.	PINV=	5.0000E+01	TNPRS=	1.5000E+01	ISTART=	0
NCYCLS=	5	XORG=	0.	RPDIF=	6.0800E-01	XSAND=	5.7000E-01	ALPH=	0.
NPRES=	21	DX=	7.5000E+02	CPDIF=	2.0000E+00	TRANS=	1.0000E+00	IFILM=	0
W=	0.	IE=	24	KMAX=	15	VEG=	0.	XCLAY=	0.
CAM=	1.0000E-02	ZNAUT=	1.0000E-01	QEFFG=	5.0000E-01	CDEPTH=	1.0000E-01	ZBOT=	0.
TDEEP=	2.8816E+02	TWATER=	2.8816E+02	JE=	24	DY=	7.5000E+02	CAM1=	5.0000E-03

SIMULATED INITIAL CONDITIONS

K	PPRES	UPRES	VPRES	TPRES	CPRES	R.H.	THETA
1	6.0000E+01	1.0000E+01	0.	-2.5006E+01	0.	0.0	2.7000E+02
2	6.2000E+01	1.0000E+01	0.	-2.2614E+01	0.	0.0	2.7000E+02
3	6.4000E+01	1.0000E+01	0.	-2.0276E+01	0.	0.0	2.7000E+02
4	6.6000E+01	1.0000E+01	0.	-1.7988E+01	0.	0.0	2.7000E+02
5	6.8000E+01	1.0000E+01	0.	-1.5750E+01	0.	0.0	2.7000E+02
6	7.0000E+01	1.0000E+01	0.	-1.3537E+01	0.	0.0	2.7000E+02
7	7.2000E+01	1.0000E+01	0.	-1.1408E+01	0.	0.0	2.7000E+02
8	7.4000E+01	1.0000E+01	0.	-9.3016E+00	0.	0.0	2.7000E+02
9	7.6000E+01	1.0000E+01	0.	-7.2346E+00	0.	0.0	2.7000E+02
10	7.8000E+01	1.0000E+01	0.	-5.2058E+00	0.	0.0	2.7000E+02
11	8.0000E+01	1.0000E+01	0.	-3.2134E+00	0.	0.0	2.7000E+02
12	8.2000E+01	1.0000E+01	0.	-1.2560E+00	0.	0.0	2.7000E+02
13	8.4000E+01	1.0000E+01	0.	6.6797E-01	0.	0.0	2.7000E+02
14	8.6000E+01	1.0000E+01	0.	2.5598E+00	0.	0.0	2.7000E+02
15	8.8000E+01	1.0000E+01	0.	4.4207E+00	0.	0.0	2.7000E+02
16	9.0000E+01	1.0000E+01	0.	6.2520E+00	0.	0.0	2.7000E+02
17	9.2000E+01	1.0000E+01	0.	8.0547E+00	0.	0.0	2.7000E+02
18	9.4000E+01	1.0000E+01	0.	9.8299E+00	0.	0.0	2.7000E+02
19	9.6000E+01	1.0000E+01	0.	1.1579E+01	0.	0.0	2.7000E+02
20	9.8000E+01	1.0000E+01	0.	1.3302E+01	0.	0.0	2.7000E+02
21	1.0000E+02	1.0000E+01	0.	1.5000E+01	0.	0.0	2.7000E+02

3-D GAUSSIAN HILL

I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	21	22	23	24																
J=24	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.																
J=23	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	1.	1.	1.	1.	1.	0.	0.	0.	0.	0.
	0.	0.	0.	0.																
J=22	0.	0.	0.	0.	0.	0.	1.	1.	2.	2.	3.	3.	3.	3.	2.	2.	1.	1.	0.	0.
	0.	0.	0.	0.																
J=21	0.	0.	0.	0.	0.	1.	2.	3.	5.	7.	10.	11.	11.	10.	7.	5.	3.	2.	1.	0.
	0.	0.	0.	0.																
J=20	0.	0.	0.	0.	1.	2.	4.	8.	14.	20.	26.	29.	29.	26.	20.	14.	8.	4.	2.	1.
	0.	0.	0.	0.																
J=19	0.	0.	0.	1.	2.	5.	11.	20.	33.	48.	62.	70.	70.	62.	48.	33.	20.	11.	5.	2.
	1.	0.	0.	0.																
J=18	0.	0.	1.	2.	4.	11.	23.	43.	70.	102.	131.	149.	149.	131.	102.	70.	43.	23.	11.	4.
	2.	1.	0.	0.																
J=17	0.	0.	1.	3.	8.	20.	43.	80.	131.	191.	245.	278.	278.	245.	191.	131.	80.	43.	20.	8.

Table 4. Sample Output (Initial Conditions continued)

	3.	1.	0.	0.																
J=16	0.	0.	2.	3.	14.	33.	70.	131.	216.	315.	404.	458.	458.	404.	315.	216.	131.	70.	33.	14.
	5.	2.	0.	0.																
J=15	0.	1.	2.	7.	20.	48.	102.	191.	315.	458.	588.	666.	666.	588.	458.	315.	191.	102.	48.	20.
	7.	2.	1.	0.																
J=14	0.	1.	3.	10.	26.	62.	131.	245.	404.	588.	755.	855.	855.	755.	588.	404.	245.	131.	62.	26.
	10.	3.	1.	0.																
J=13	0.	1.	3.	11.	29.	70.	149.	278.	458.	666.	855.	969.	969.	855.	666.	458.	278.	149.	70.	29.
	11.	3.	1.	0.																
J=12	0.	1.	3.	11.	29.	70.	149.	278.	458.	666.	855.	969.	969.	855.	666.	458.	278.	149.	70.	29.
	11.	3.	1.	0.																
J=11	0.	1.	3.	10.	26.	62.	131.	245.	404.	588.	755.	855.	855.	755.	588.	404.	245.	131.	62.	26.
	10.	3.	1.	0.																
J=10	0.	1.	2.	7.	20.	48.	102.	191.	315.	458.	588.	666.	666.	588.	458.	315.	191.	102.	48.	20.
	7.	2.	1.	0.																
J= 9	0.	0.	2.	5.	14.	33.	70.	131.	216.	315.	404.	458.	458.	404.	315.	216.	131.	70.	33.	14.
	5.	2.	0.	0.																
J= 8	0.	0.	1.	3.	8.	20.	43.	80.	131.	191.	245.	278.	278.	245.	191.	131.	80.	43.	20.	8.
	3.	1.	0.	0.																
J= 7	0.	0.	1.	2.	4.	11.	23.	43.	70.	102.	131.	149.	149.	131.	102.	70.	43.	23.	11.	4.
	2.	1.	0.	0.																
J= 6	0.	0.	0.	1.	2.	5.	11.	20.	33.	48.	62.	70.	70.	62.	48.	33.	20.	11.	5.	2.
	1.	0.	0.	0.																
J= 5	0.	0.	0.	0.	1.	2.	4.	8.	14.	20.	26.	29.	29.	26.	20.	14.	8.	4.	2.	1.
	0.	0.	0.	0.																
J= 4	0.	0.	0.	0.	0.	1.	2.	3.	5.	7.	10.	11.	11.	10.	7.	5.	3.	2.	1.	0.
	0.	0.	0.	0.																
J= 3	0.	0.	0.	0.	0.	0.	1.	1.	2.	2.	3.	3.	3.	3.	2.	2.	1.	1.	0.	0.
	0.	0.	0.	0.																
J= 2	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	1.	1.	1.	1.	1.	0.	0.	0.	0.	0.
	0.	0.	0.	0.																
J= 1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.																

PCUT BASED ON ZBL

I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	21	22	23	24	25															
J=25	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8
	98.8	98.8	98.8	98.8	98.8															
J=24	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8
	98.8	98.8	98.8	98.8	98.8															
J=23	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8
	98.8	98.8	98.8	98.8	98.8															
J=22	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.7	98.7	98.7	98.8	98.8	98.8	98.8	98.8	98.8
	98.8	98.8	98.8	98.8	98.8															
J=21	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.7	98.7	98.6	98.6	98.6	98.6	98.6	98.7	98.7	98.8	98.8	98.8
	98.8	98.8	98.8	98.8	98.8															
J=20	98.8	98.8	98.8	98.8	98.8	98.8	98.8	98.7	98.6	98.5	98.4	98.3	98.3	98.3	98.4	98.5	98.6	98.7	98.8	98.8
	98.8	98.8	98.8	98.8	98.8															
J=19	98.8	98.8	98.8	98.8	98.8	98.8	98.7	98.6	98.4	98.1	97.9	97.7	97.6	97.7	97.9	98.1	98.4	98.6	98.7	98.8
	98.8	98.8	98.8	98.8	98.8															
J=18	98.8	98.8	98.8	98.8	98.8	98.7	98.6	98.3	97.9	97.5	97.0	96.7	96.5	96.7	97.0	97.5	97.9	98.3	98.6	98.7
	98.8	98.8	98.8	98.8	98.8															
J=17	98.8	98.8	98.8	98.8	98.7	98.6	98.4	97.9	97.3	96.5	95.7	95.1	94.9	95.1	95.7	96.5	97.3	97.9	98.4	98.6
	98.7	98.8	98.8	98.8	98.8															
J=16	98.8	98.8	98.8	98.8	98.7	98.5	98.1	97.5	96.5	95.3	92.7	91.8	91.5	91.8	92.7	95.3	96.5	97.5	98.1	98.5
	98.7	98.8	98.8	98.8	98.8															
J=15	98.8	98.8	98.8	98.8	98.6	98.4	97.9	97.0	95.7	92.7	91.1	90.0	89.6	90.0	91.1	92.7	95.7	97.0	97.9	98.4
	98.6	98.8	98.8	98.8	98.8															
J=14	98.8	98.8	98.8	98.7	98.6	98.3	97.7	96.7	95.1	91.8	90.0	88.6	88.1	88.6	90.0	91.8	95.1	96.7	97.7	98.3
	98.6	98.7	98.8	98.8	98.8															
J=13	98.8	98.8	98.8	98.7	98.6	98.3	97.6	96.5	94.9	91.5	89.6	88.1	87.6	88.1	89.6	91.5	94.9	96.5	97.6	98.3
	98.6	98.7	98.8	98.8	98.8															
J=12	98.8	98.8	98.8	98.7	98.6	98.3	97.7	96.7	95.1	91.8	90.0	88.6	88.1	88.6	90.0	91.8	95.1	96.7	97.7	98.3
	98.6	98.7	98.8	98.8	98.8															
J=11	98.8	98.8	98.8	98.8	98.6	98.4	97.9	97.0	95.7	92.7	91.1	90.0	89.6	90.0	91.1	92.7	95.7	97.0	97.9	98.4

Table 4. Sample Output (5 Cycle Run)

COMPLETED 5 CYCLES. ELAPSED SIMULATION TIME IS 8.0000E+00 SEC DAY= 1 TIME= 0.15. 8

U VELOCITY TIME CYCLE= 5 K LEVEL= 15

I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
J=25	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62
J=24	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62
J=23	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.63	7.63	7.63	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62
J=22	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.63	7.63	7.63	7.63	7.63	7.63	7.63	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62
J=21	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.63	7.64	7.65	7.65	7.65	7.64	7.63	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62
J=20	7.64	7.62	7.62	7.62	7.62	7.62	7.61	7.62	7.62	7.64	7.66	7.68	7.69	7.68	7.67	7.65	7.63	7.62	7.61	7.62	7.62	7.62	7.62	7.62	7.62
J=19	7.64	7.62	7.62	7.62	7.61	7.61	7.61	7.61	7.63	7.66	7.70	7.74	7.76	7.75	7.71	7.67	7.64	7.61	7.61	7.61	7.61	7.62	7.62	7.62	7.62
J=18	7.64	7.62	7.62	7.62	7.61	7.60	7.59	7.60	7.64	7.70	7.77	7.85	7.88	7.86	7.79	7.71	7.65	7.61	7.60	7.60	7.61	7.61	7.62	7.62	7.62
J=17	7.64	7.62	7.62	7.61	7.60	7.59	7.58	7.59	7.64	7.73	7.90	8.02	8.08	8.04	7.94	7.77	7.66	7.60	7.58	7.58	7.60	7.61	7.62	7.62	7.62
J=16	7.64	7.62	7.62	7.60	7.59	7.57	7.56	7.58	7.64	7.87	6.88	6.93	6.99	6.94	6.78	7.79	7.68	7.59	7.56	7.57	7.58	7.60	7.61	7.62	7.62
J=15	7.64	7.62	7.61	7.60	7.58	7.55	7.54	7.56	7.70	6.74	6.87	7.06	7.15	7.09	6.92	6.67	7.62	7.58	7.54	7.55	7.57	7.60	7.61	7.62	7.62
J=14	7.64	7.62	7.61	7.60	7.57	7.54	7.52	7.55	7.70	6.79	6.92	7.16	7.28	7.20	6.97	6.72	7.64	7.58	7.52	7.54	7.57	7.59	7.61	7.62	7.62
J=13	7.64	7.62	7.61	7.59	7.57	7.54	7.51	7.55	7.71	6.80	6.95	7.20	7.34	7.25	7.00	6.73	7.65	7.57	7.52	7.53	7.56	7.59	7.61	7.62	7.62
J=12	7.64	7.62	7.61	7.60	7.57	7.54	7.52	7.55	7.70	6.78	6.92	7.15	7.27	7.19	6.97	6.72	7.64	7.57	7.52	7.54	7.57	7.59	7.61	7.62	7.62
J=11	7.64	7.62	7.61	7.60	7.58	7.55	7.53	7.56	7.69	6.74	6.86	7.04	7.13	7.07	6.91	6.66	7.61	7.58	7.54	7.55	7.57	7.60	7.61	7.62	7.62
J=10	7.64	7.62	7.62	7.60	7.59	7.57	7.55	7.57	7.64	7.86	6.87	6.91	6.97	6.93	6.78	7.79	7.67	7.59	7.56	7.57	7.59	7.60	7.61	7.62	7.62
J= 9	7.64	7.62	7.62	7.61	7.60	7.59	7.58	7.59	7.64	7.72	7.89	8.01	8.07	8.03	7.93	7.76	7.66	7.60	7.58	7.58	7.60	7.61	7.62	7.62	7.62
J= 8	7.64	7.62	7.62	7.62	7.61	7.60	7.60	7.60	7.63	7.69	7.76	7.84	7.87	7.85	7.78	7.71	7.64	7.61	7.60	7.60	7.61	7.61	7.62	7.62	7.62
J= 7	7.64	7.62	7.62	7.62	7.61	7.61	7.61	7.61	7.63	7.66	7.70	7.73	7.75	7.74	7.70	7.67	7.64	7.62	7.61	7.61	7.61	7.62	7.62	7.62	7.62
J= 6	7.64	7.62	7.62	7.62	7.62	7.62	7.61	7.62	7.63	7.64	7.66	7.68	7.69	7.68	7.67	7.65	7.63	7.62	7.61	7.62	7.62	7.62	7.62	7.62	7.62
J= 5	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.63	7.64	7.65	7.65	7.65	7.64	7.63	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62
J= 4	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.63	7.63	7.63	7.63	7.63	7.63	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62
J= 3	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.63	7.63	7.63	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62
J= 2	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62
J= 1	7.64	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62	7.62

V VELOCITY TIME CYCLE= 5 K LEVEL= 15

I= 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Table 4. Sample Output (5-Cycle Run continued)

	21	22	23	24	25															
J=25	0.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00
	-.00	-.00	-.00	-.00	-.00															
J=24	0.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00
	-.00	-.00	-.00	-.00	-.00															
J=23	0.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00
	-.00	-.00	-.00	-.00	-.00															
J=22	0.00	.00	.00	.00	.00	.00	.00	.01	.01	.01	.01	.00	.00	-.00	-.01	-.01	-.01	-.01	-.00	-.00
	-.00	-.00	-.00	-.00	-.00															
J=21	0.00	.00	.00	.00	.00	.00	.01	.01	.02	.02	.02	.01	.00	-.01	-.02	-.02	-.02	-.01	-.01	-.01
	-.00	-.00	-.00	-.00	-.00															
J=20	0.00	.00	.00	.00	.00	.01	.02	.03	.03	.04	.04	.02	.00	-.02	-.03	-.04	-.04	-.03	-.02	-.01
	-.01	-.00	-.00	-.00	-.00															
J=19	0.00	.00	.00	.00	.01	.02	.03	.05	.06	.07	.06	.04	.00	-.03	-.06	-.07	-.06	-.05	-.03	-.02
	-.01	-.00	-.00	-.00	-.00															
J=18	0.00	.00	.00	.01	.01	.03	.05	.07	.10	.11	.10	.06	.01	-.05	-.09	-.11	-.10	-.08	-.05	-.03
	-.01	-.01	-.00	-.00	-.00															
J=17	0.00	.00	.00	.01	.02	.03	.06	.09	.13	.16	.15	.08	.01	-.07	-.14	-.16	-.13	-.10	-.07	-.04
	-.02	-.01	-.00	-.00	-.00															
J=16	0.00	.00	.00	.01	.02	.04	.07	.10	.16	.18	.15	.09	.01	-.08	-.14	-.18	-.16	-.11	-.07	-.04
	-.02	-.01	-.00	-.00	-.00															
J=15	0.00	.00	.00	.01	.02	.03	.06	.09	.14	.15	.12	.08	.01	-.07	-.11	-.14	-.15	-.10	-.06	-.04
	-.02	-.01	-.00	-.00	-.00															
J=14	0.00	.00	.00	.00	.01	.02	.03	.06	.07	.09	.08	.05	.00	-.04	-.08	-.09	-.08	-.06	-.04	-.02
	-.01	-.00	-.00	-.00	-.00															
J=13	0.00	.00	-.00	-.00	.00	.00	.00	.00	.00	.00	.00	.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00
	-.00	.00	.00	-.00	-.00															
J=12	0.00	-.00	-.00	-.00	-.01	-.02	-.03	-.06	-.07	-.09	-.08	-.05	-.00	.04	.08	.09	.08	.06	.04	.02
	.01	.00	.00	.00	.00															
J=11	0.00	-.00	-.00	-.01	-.02	-.03	-.06	-.09	-.14	-.15	-.12	-.08	-.01	.07	.11	.14	.15	.10	.06	.04
	.02	.01	.00	.00	.00															
J=10	0.00	-.00	-.00	-.01	-.02	-.04	-.07	-.10	-.16	-.18	-.15	-.09	-.01	.08	.14	.18	.16	.11	.07	.04
	.02	.01	.00	.00	.00															
J= 9	0.00	-.00	-.00	-.01	-.02	-.03	-.06	-.09	-.13	-.16	-.15	-.08	-.01	.07	.14	.16	.13	.10	.07	.04
	.02	.01	.00	.00	.00															
J= 8	0.00	-.00	-.00	-.01	-.01	-.03	-.05	-.07	-.10	-.11	-.10	-.06	-.01	.05	.09	.11	.10	.08	.05	.03
	.01	.01	.00	.00	.00															
J= 7	0.00	-.00	-.00	-.00	-.01	-.02	-.03	-.05	-.06	-.07	-.06	-.04	-.00	.03	.06	.07	.06	.05	.03	.02
	.01	.00	.00	.00	.00															
J= 6	0.00	-.00	-.00	-.00	-.00	-.01	-.02	-.03	-.03	-.04	-.04	-.02	-.00	.02	.03	.04	.04	.03	.02	.01
	.01	.00	.00	.00	.00															
J= 5	0.00	-.00	-.00	-.00	-.00	-.00	-.01	-.01	-.02	-.02	-.02	-.01	-.00	.01	.02	.02	.02	.01	.01	.01
	.00	.00	.00	.00	.00															
J= 4	0.00	-.00	-.00	-.00	-.00	-.00	-.00	-.01	-.01	-.01	-.01	-.00	-.00	.00	.01	.01	.01	.01	.00	.00
	.00	.00	.00	.00	.00															
J= 3	0.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	.00	.00	.00	.00	.00	.00	.00
	.00	.00	.00	.00	.00															
J= 2	0.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	.00	.00	.00	.00	.00	.00	.00
	.00	.00	.00	.00	.00															
J= 1	0.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	-.00	.00	.00	.00	.00	.00	.00	.00
	.00	.00	.00	.00	.00															

TEMPERATURE TIME CYCLE= 5 . K LEVEL= 15

I=	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	16	17	18	19	20	21	22	23	24						
J=24	287.876	287.873	287.873	287.873	287.873	287.873	287.873	287.872	287.872	287.871	287.871	287.871	287.871	287.871	287.872
	287.872	287.873	287.873	287.873	287.873	287.873	287.873	287.873	287.873						
J=23	287.876	287.873	287.873	287.873	287.873	287.873	287.872	287.870	287.868	287.866	287.864	287.863	287.863	287.865	287.867
	287.869	287.870	287.872	287.872	287.873	287.873	287.873	287.873	287.873						
J=22	287.876	287.873	287.873	287.873	287.872	287.871	287.868	287.863	287.856	287.849	287.842	287.838	287.838	287.843	287.850
	287.857	287.863	287.868	287.871	287.872	287.873	287.873	287.873	287.873						
J=21	287.876	287.873	287.873	287.872	287.870	287.866	287.857	287.842	287.822	287.798	287.777	287.765	287.766	287.779	287.800

REFERENCES

1. Anthes, R.A., 1972: "The Development of Asymmetries in a Three-Dimensional Numerical Model of the Tropical Cyclone," Mon. Wea. Rev., 100, 461.

2. Anthes, R.A. and T.T. Warner, 1974: "Prediction of Mesoscale Flows Over Complex Terrain," U.S. Army Electronics Command Report ECOM-5532, AD77633, March.

3. Asselin, R., 1972: "Frequency Filter for Time Integration," Mon. Wea. Rev., 100, 487-490.

4. Freeman, B.E., P.C. Patnaik and G.T. Phillips, 1976: "Development of a Wind Energy Site Selection Methodology," DOE Report No. RLO/2440-76/4, December.

5. Gates, W.L., E.S. Batten, A.B. Kahle and A.B. Nelson, 1971: "A Documentation of the Mintz-Arakawa Two-Level Atmospheric General Circulation Model," Rand Corporation Report No. R-877-ARPA, ARPA Order No. 189-1, December.

6. Katayama, A., 1974: "A Simplified Scheme for Computing Radiative Transfer in the Troposphere," Technical Report No. 6, Department of Meteorology, UCLA.

7. Mellor, G.L., and T. Yamada, 1974: "A Hierarchy of Turbulence Closure Models for Planetary Boundary Layers," J. Atmos. Sci., 31, 1791-1806.

8. Shuman, F.G. and J.D. Stackpole, 1968: "Note on the Formulation of Finite Difference Equations Incorporating a Map Scale Factor," Mon. Wea. Rev., 96 (3), 157-161.

9. Smith, T.B. and S.M. Howard, 1972: "Methodology for Treating Diffusivity," Meteorology Research Inc., Report, September.

10. Traci, R.M., G.T. Phillips, P.C. Patnaik and B.E. Freeman, 1977: "Development of a Wind Energy Site Selection Methodology," DOE Report No. RLO/2440-77/11, June.

11. Traci, R.M., G.T. Phillips and P.C. Patnaik, 1978: "Developing a Site Selection Methodology for Wind Energy Conversion Systems," DOE Report No. RLO/2440-78/2, September.

APPENDIX A

Glossary of Input Variables

APPENDIX A

Glossary of Input Variables

The set of input variables required to define a SIGMET simulation are presented in the following table. Variables are input in standard CDC namelist format in either NAMELIST /OUTPT/ or NAMELIST /START/ as indicated.

NAMELIST /OUTPT/

Variable	Description
ISTART:	Starting parameter = 0 for initial run > 0 for restart runs. For restart runs save tape should be mounted on logical Unit 7
IEDIT:	Frequency of edit cycle
IPLOT:	Frequency of printer plot
IOUT:	Frequency of debug edit cycle
IFILM:	Frequency of film plot
NCYCLS:	Total number of time cycles
KBEG:	Starting edit level (If IEDIT is non-zero printer edit is provided level by level for all variables from K = KBEG to KMAX)
KBGPL:	If IFILM is non-zero plots on horizontal slices K=KBGPL to KMAX are to be prepared

NAMELIST /START/

Variable	Description
KMAX:	No. of vertical levels
IE:	No. of computational zones in x-direction
JE:	No. of computational zones in y-direction
DX:	Grid size in x-direction in meters
DY:	Grid size in y-direction in meters
DELTAT:	Time step size in seconds
GAM TNPRS GAM1 PINV	By assigning a non-negative value to GAM the measured profile of temperature (TPRES) can be replaced by one with TNPRS (in °C) at ground decreasing with altitude at lapse rate GAM (in °C per meter). An inversion layer can be simulated by having a smaller lapse rate GAM1 (in °C/meter) starting at a level where pressure is PINV (in millibar).
ANGLE:	x-axis rotation measured counter clockwise from East in degrees
ZBOT:	Ground elevation (AGL) of measurement location if available
TG:	A two-dimensional array at least IE x JE long representing ground temperature is °C. If measured values for TG are not available, they can be extrapolated from the measurement of temperature in the atmosphere. This is done by setting the value of GDEPTH negative.
GW:	A two dimensional array at least IE x JE long representing ground moisture expressed as fraction of the saturated humidity of air near the ground
HRATE:	A two-dimensional array at least IE x JE long representing external ground heat source, if any
TWATER:	Temperature in °K of water body, if any, present in the region of simulation
TDEEP:	Temperature in °K of underground soil assumed constant everywhere
XORG XCLAY XSAND	Composition of soil in the region of simulation, in fraction by mass of organic materials, clay and sand respectively
VEG:	Vegetated fraction of the total surface area
ZNAUT:	Roughness length in meters typical to the terrain over the region

Variable	Description
DLAT:	Latitude in degrees of region of simulation
DLONG:	Longitude in degrees of region of simulation
Q:	One dimensional array of at least KMAX +1 long for σ values
CX1, CX2:	Two parameters specifying the log-linear distribution of σ levels
CX3, CX4:	Two parameters specifying σ values according to a geometric progression series Note: If one of two parameters is zero σ values are not computed and preset values (read through Q) are used
GMTMHR:	GMT hour (in fraction notation) corresponding to initial simulation time
DAYBEG:	Julian day of initial simulation
PTOP:	Pressure in millibar corresponding to the top level of the model
PBOT:	Standard sea level pressure in millibar
UPRES, VPRES, TPRES, CPRES	Ambient (usually obtained from soundings) values of horizontal wind in meter/sec or in mph, temperature ^{o}C, humidity in Kg/Kg. These are one dimensional arrays of length 30 each
PPRES:	One dimensional array of size 30 representing pressure in millibar where UPRES, VPRES etc. are measured
NPRES:	No. of entries of UPRES, VPRES etc. ($\leq$ 30)
THETA:	A one dimensional array of at least NPRES length representing angle in degrees of wind vector measured clockwise from north. If the measurement is available in magnitude/direction format, UPRES should contain the magnitude in meters per sec and THETA the direction. If component measurements are to be entered, UPRES should contain the East-West component, VPRES should contain the North South component; both in miles per hour and all THETA should be set at zero.
RELHM:	An array of at least NPRES elements representing values of relative humidity at NPRES levels of the atmosphere. Expressed as percentage of saturated value for air at given temperature. If any of RELHM is non-zero, the measurement CPRES is replaced by new values calculated from RELHM and TPRES.

Variable	Description
ALPH:	Time smoothing parameter to damp out high frequency temporal oscillations, $0 \leq ALPH \leq 1$
ZBL:	Boundary layer height in meters above which an interpolated value from UPRES, VPRES is used for initializing the velocity field. Below ZBL a power law profile is used.
IRAD:	Frequency of cycles when radiation subroutines are called. If zero, radiation contribution is completely ignored.
W TRANS SUREM QEFFG	Radiation parameters
GDEPTH:	Length scale in meters required for solution of ground temperature
IJCOLM:	A two dimensional array of size IE x JE, each element corresponding to a given vertical column. A non-zero value will result in printout of a column $(1 \leq K \leq KMAX)$ of flow variables every IOUT cycles.

APPENDIX B

Computer Listing

```
      PROGRAM MAIN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7,TAPE8,   SIG3DE.2
     .             FILM,TAPE10=FILM)                                      SIG3DE.3
                                                                          SIG3DE.4
C-----THIS IS AN EXPLICIT VERSION OF SIGMET HYDROSTATIC CODE IN 3-D       SIG3DE.5
C-----WITH LATEST CHANGES MADE IN MAY,1980                                NEW.5
                                                                          SIG3DE.7
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15)  COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15)  COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)      COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),        COMMON.5
     .             TTEN(26,22,15)                                        COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)      COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)            COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),         COMMON.9
     .             PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)       COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),GG(25,21),          COMMON.11
     .             GW(25,21),HRATE(25,21),IJCOLM(25,21)                  COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),   COMMON.13
     .             PPRES(30),RELHM(30),UADAT(8),TRDAT(8)                 RATELE.1
      COMMON/RADIAT/RADTN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),     COMMON.15
     .             RADSO(25,21,16)                                       COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),            COMMON.17
     .             SDOTL(21,16)                                          COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)            COMMON.19
      COMMON/MISCLN/INT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .             ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),RI(16),       COMMON.21
     .             DVELDZ(16)                                            COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)    COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),             COMMON.24
     .             QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,    COMMON.25
     .             SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,       COMMON.26
     .             DAYNOW,AZMUTH,SIGMA,R,HRANGL,TANZEN,SINAZ,COSAZ,      COMMON.27
     .             SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,       COMMON.28
     .             XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,            COMMON.29
     .             HHHH,ELEE,RRRR,SSSS,NPLT,NPLTT,SUREM                  COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTMI,SIMTIM,NCYCLS,IEDIT,IPLOT,    COMMON.31
     .             ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,          COMMON.32
     .             CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,     COMMON.33
     .             CPDIF,RPDIF,ZBL,RHOFAC,GAM,IF,IEND,IEM1,IS,KS,DX,     COMMON.34
     .             RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISHNS,   COMMON.35
     .             GMTMHP,GMDYBG,DIFQ,DIF1,DY,RDY,RDYH,RDY4,RDY16,       COMMON.36
     .             RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,      COMMON.37
     .             KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4         COMMON.38
      COMMON/ADARAY/IDUM                                                 COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,   COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO          COMMON.41
                                                                          COMMON.42
      COMMON/IGSZZZ/Z(200)                                               COMMPL.2
      COMMON/TRANSF/ITRX,ITRY,TRN(26,22)                                 COMMPL.3
                                                                          SIG3DE.10
C*****************************************************************SIG3DE.11
C*          DEFINE THE INITIAL STATE OF THE SYSTEM                      *SIG3DE.12
C*****************************************************************SIG3DE.13
      CALL INITAL                                                        SIG3DE.14
C-----INITIALIZE FILM PLOTTER                                            SIG3DE.15
```

```
      IF(IFILM.GT.0) CALL MODESG(7,6,12HSIG3DE PLOTS)                   SIG3DE.16
      NCYCL=0                                                           SIG3DE.17
      LTAPE=0                                                           NEW.6
      CALL OUTPUT                                                       SIG3DE.18
      IF(IOUT.EQ.1) LOUT=1                                              SIG3DE.19
                                                                        SIG3DE.20
C***********************************************************************SIG3DE.21
C*          CALCULATIONAL LOOP                                         *SIG3DE.22
C***********************************************************************SIG3DE.23
      DO 400 N=1,NCYCLS                                                 SIG3DE.24
      NCYCL=N                                                           SIG3DE.25
      ICYCL=ICYCL+1                                                     NEW.7
                                                                        SIG3DE.26
C-----VISCOUS EFFECTS - HORIZONTAL - EVALUATE AT T=T(N-1)               SIG3DE.27
      CALL FRICH                                                        SIG3DE.28
                                                                        SIG3DE.29
C-----FORM U,V,T,C FIELDS                                               SIG3DE.30
      CALL FIELDS                                                       SIG3DE.31
                                                                        SIG3DE.32
C-----FORM HORIZONTAL TERMS                                             SIG3DE.33
      CALL TENDH                                                        SIG3DE.34
                                                                        SIG3DE.35
C-----FORM PRESSURE TERMS                                               SIG3DE.36
      CALL PRESUR                                                       SIG3DE.37
                                                                        SIG3DE.38
C-----OBTAIN RADIATIVE FLUX FIELDS                                      SIG3DE.39
      IF(IRAD.GT.0) CALL RADFLD                                         SIG3DE.40
                                                                        SIG3DE.41
C-----CALCULATE VERTICALLY COUPLED QUANTITIES                           SIG3DE.42
      DO 300 I=1,IE                                                     SIG3DE.43
      DO 300 J=1,JE                                                     SIG3DE.44
      IS=I                                                              SIG3DE.45
      JS=J                                                              SIG3DE.46
                                                                        SIG3DE.47
C-----VERTICAL ADVECTION AND CORIOLIS TERMS                             SIG3DE.48
      CALL TENDV                                                        SIG3DE.49
                                                                        SIG3DE.50
C-----DETERMINE VERTICAL MIXING COEFFICIENTS                            SIG3DE.51
      CALL FRICV                                                        SIG3DE.52
                                                                        SIG3DE.53
C-----DETERMINE EXTRAPOLATED RADIATIVE FLUX                             SIG3DE.54
      IF(IRAD.GT.0) CALL RADFLX                                         SIG3DE.55
                                                                        SIG3DE.56
C-----DO VERTICAL DIFFUSION IMPLICITLY ALONG WITH HEAT BALANCE BET. SOILSIG3DE.57
      CALL DIFFV                                                        SIG3DE.58
                                                                        SIG3DE.59
C-----CONTRIBUTION OF RADIATIVE FLUX TO ENERGY EQN.                     SIG3DE.60
      IF(IRAD.EQ.0) GO TO 112                                           SIG3DE.61
      DO 111 K=1,KMAX                                                   SIG3DE.62
  111 TTEN(IS,JS,K)=TTEN(IS,JS,K)+9.81*(FLUXL2(K+1)-FLUXL2(K)-RFLUX(K+1)SIG3DE.63
     .+RFLUX(K))*DQI(K)/(RHOFAC*1004.0*(1.0+CPOIF*C(IS,JS,K)))          SIG3DE.64
     .*TA(IS,JS,K)/(PSA(IS,JS)*T(IS,JS,K))                              NEW.8
  112 CONTINUE                                                          SIG3DE.65
                                                                        SIG3DE.66
C-----ADD ANY SURFACE HEAT SOURCES                                      SIG3DE.67
```

```
      TTEN(IS,JS,KMAX)=TTEN(IS,JS,KMAX)+HRATE(IS,JS)*TA(IS,JS,KMAX)     NEW.9
     ./T(IS,JS,KMAX)                                                    NEW.10
                                                                        SIG3DE.69
C-----UPDATE ALL QUANTITIES                                             SIG3DE.70
      CALL UPDATE                                                       SIG3DE.71
  300 CONTINUE                                                          SIG3DE.72
                                                                        SIG3DE.73
C-----SET LATERAL BOUNDARY CONDITIONS ON VECTORS                         SIG3DE.74
      CALL UVBOUN                                                       SIG3DE.75
                                                                        SIG3DE.76
C-----ADVANCE THE TIME                                                  SIG3DE.77
      SIMTIM=SIMTIM+DELTAT                                              SIG3DE.78
      CALL SSWTCH(6,KSTOP) $ GO TO (349,350),KSTOP                      SIG3DE.80
C-----PRODUCE ANY REQUESTED OUTPUT                                      SIG3DE.81
  350 CALL OUTPUT                                                       SIG3DE.82
  400 CONTINUE                                                          SIG3DE.83
                                                                        SIG3DE.84
C-----WRITE DUMP TAPE TO BE USED FOR RESTART                            SIG3DE.85
  349 LTAPE=1                                                           SIG3DE.86
      CALL OUTPUT                                                       SIG3DE.87
C-----CLOSE FILM PLOTTER OUTPUT FILE                                    SIG3DE.88
      IF(IFILM.GT.0) CALL EXITG(7)                                      SIG3DE.89
                                                                        SIG3DE.90
      STOP                                                              SIG3DE.91
      END                                                               SIG3DE.92
```

```
CARD NR. SEVERITY  DETAILS    DIAGNOSIS OF PROBLEM

   123      I                AN IF STATEMENT MAY BE MORE EFFICIENT THAN A 2 OR 3 BRANCH COMPUTED GO TO STATEMENT.
```

```
      SUBROUTINE CONTUR(VAR,IDIM,IMAX,JMAX,DX,DY,KS,ITYPE)                CONTUR.2
                                                                          CONTUR.3
C     THIS CONSTRUCTS CONTOUR PLOTS ON A HORIZONTAL SLICE USING BKY       CONTUR.4
C     IGSNEAR PLOT PACKAGE                                                CONTUR.5
                                                                          CONTUR.6
      COMMON/IGSZZZ/Z(200)                                                COMMPL.2
      COMMON/TRANSF/ITRX,ITRY,TRN(26,22)                                  COMMPL.3
      DIMENSION VAR(IDIM,JMAX)                                            CONTUR.8
      DIMENSION CCL(40)                                                   CONTUR.9
                                                                          CONTUR.10
C-----SET UP PLOTTER TO FIT PLOTS NICELY ON 8.5X11 PAPER                  CONTUR.11
      XDIM=0.9                                                            CONTUR.12
      YDIM=0.65                                                           CONTUR.13
      TOT=1.0                                                             CONTUR.14
      IF(ITYPE.LE.5) XDIM=XDIM*FLOAT(IMAX-1)/FLOAT(IMAX)                  CONTUR.15
      IF(ITYPE.LE.5) YDIM=YDIM*FLOAT(JMAX-1)/FLOAT(JMAX)                  CONTUR.16
      ASPECT=XDIM/YDIM                                                    CONTUR.17
      XX=(IMAX-1.0)*DX                                                    CONTUR.18
      YY=(JMAX-1.0)*DY                                                    CONTUR.19
      XYA=XX/YY                                                           CONTUR.20
      IF(XYA.LT.ASPECT) GO TO 101                                         CONTUR.21
      XIN=XDIM                                                            CONTUR.22
      YIN=XIN/XYA                                                         CONTUR.23
      GO TO 102                                                           CONTUR.24
  101 YIN=YDIM                                                            CONTUR.25
      XIN=YIN*XYA                                                         CONTUR.26
  102 XORIG=(TOT-XIN)/2.0                                                 CONTUR.27
      YORIG=(TOT-YIN)/2.0                                                 CONTUR.28
                                                                          CONTUR.29
C-----EXPLICITLY SET UP OBJECT AND SUBJECT SUBSPACE FOR CONTOURING        CONTUR.30
      X1=1.0                                                              CONTUR.31
      Y1=1.0                                                              CONTUR.32
      X2=FLOAT(IMAX)                                                      CONTUR.33
      Y2=FLOAT(JMAX)                                                      CONTUR.34
      TICK=AMAX1(X2,Y2)*0.01                                              CONTUR.35
      CALL SETSMG(Z,19,1.0)                                               CONTUR.36
      CALL SETSMG(Z,20,1.0)                                               CONTUR.37
      CALL OBJCTG(Z,0.0,0.0,1.0,1.0)                                      CONTUR.38
      CALL SET(XORIG,XORIG+XIN,YORIG,YORIG+YIN,X1,X2,Y1,Y2,1)             CONTUR.39
C-----DRAW GRID WITH NO INTERNAL LINES EXCEPT TICK MARKS                  CONTUR.40
      CALL SETSMG(Z,30,2.0)                                               CONTUR.41
      CALL BOXG(Z,0.0,X1,Y1,X2,Y1,X1,Y2,X2,Y2)                            CONTUR.42
      CALL MLTPLG(Z,IMAX-2,X1,Y1,X1,Y1+TICK,X2,Y1,X2,Y1+TICK)             CONTUR.43
      CALL MLTPLG(Z,IMAX-2,X1,Y2,X1,Y2-TICK,X2,Y2,X2,Y2-TICK)             CONTUR.44
      CALL MLTPLG(Z,JMAX-2,X1,Y1,X1+TICK,Y1,X1,Y2,X1+TICK,Y2)             CONTUR.45
      CALL MLTPLG(Z,JMAX-2,X2,Y1,X2-TICK,Y1,X2,Y2,X2-TICK,Y2)             CONTUR.46
      CALL SETSMG(Z,30,1.0)                                               CONTUR.47
                                                                          CONTUR.48
C-----CONTOUR LEVELS                                                      CONTUR.49
      NCL=9                                                               CONTUR.50
      OMAX=-1.E+8                                                         CONTUR.51
      OMIN= 1.E+8                                                         CONTUR.52
      DO 910 I=1,IMAX                                                     CONTUR.53
      DO 910 J=1,JMAX                                                     CONTUR.54
      OMAX=AMAX1(OMAX,VAR(I,J))                                           CONTUR.55
```

```
            QMIN=AMIN1(QMIN,VAR(I,J))                                       CONTUR.56
        910 CONTINUE                                                        CONTUR.57
            DQQ=(QMAX-QMIN)/FLOAT(NCL+1)                                    CONTUR.58
            DO 911 I=1,NCL                                                  CONTUR.59
            CCL(I)=QMIN+I*DQQ                                               CONTUR.60
        911 CONTINUE                                                        CONTUR.61
            WRITE (6,912) ITYPE,KS,QMIN,QMAX,DQQ,(CCL(I),I=1,NCL)           CONTUR.62
        912 FORMAT (23H CONTOUR LEVELS  ITYPE=,I2,3H K=,I2,3F10.4/(1X,9E10.4))NEW.11
                                                                            CONTUR.64
      C-----CONTOURING, NO COORDINATE TRANSFORMATION                        CONTUR.65
            ITRX=0                                                          CONTUR.66
            ITRY=0                                                          CONTUR.67
            CALL CONREG(VAR,IDIM,IMAX,JMAX,CCL,NCL,1, 1,0,1)                CONTUR.68
                                                                            CONTUR.69
      C-----SUPERIMPOSE THE TERRAIN CONTOUR ON THE V AND V**3 CONTOURS      CONTUR.70
            IF(ITYPE.LE.5) GO TO 110                                        CONTUR.71
            NCL=9                                                           CONTUR.72
            QMAX=-1.E+8                                                     CONTUR.73
            QMIN= 1.E+8                                                     CONTUR.74
            DO 920 I=1,IMAX                                                 CONTUR.75
            DO 920 J=1,JMAX                                                 CONTUR.76
            QMAX=AMAX1(QMAX,TRN(I,J))                                       CONTUR.77
            QMIN=AMIN1(QMIN,TRN(I,J))                                       CONTUR.78
        920 CONTINUE                                                        CONTUR.79
            DQQ=(QMAX-QMIN)/FLOAT(NCL+1)                                    CONTUR.80
            DO 930 I=1,NCL                                                  CONTUR.81
            CCL(I)=QMIN+I*DQQ                                               CONTUR.82
        930 CONTINUE                                                        CONTUR.83
            CALL SETSMG(7,30,0.3)                                           CONTUR.84
            ITRX=0                                                          CONTUR.85
            ITRY=0                                                          CONTUR.86
            CALL CONREC(TRN,IMAX,IMAX,JMAX,CCL,NCL,1,0,1736B,0)             CONTUR.87
            CALL SETSMG(7,30,1.0)                                           CONTUR.88
        110 CONTINUE                                                        CONTUR.89
            CALL PAGEG(7,0,1,1)                                             CONTUR.90
                                                                            CONTUR.91
            RETURN                                                          CONTUR.92
            END                                                             CONTUR.93
```

```
      SUBROUTINE DIFFV                                                  DIFFV.2
                                                                        DIFFV.3
C     "DIFV" EVALUATES THE VERTICAL DIFFUSION TERMS AT N+1 TO           DIFFV.4
C     REMOVE THE RESTRICTIVE EXPLICT STABILITY CONDITION. ALSO, THE     DIFFV.5
C     HEAT BALANCE EQN. BET. SOIL AND ATMOPHERE IS SOLVED SIMULTNLY.    DIFFV.6
C     FOR TG TO PROVIDE THE BOTTOM BOUNDARY CONDITION FOR TEMPERATURE EQDIFFV.7
                                                                        DIFFV.8
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATELF.1
      COMMON/RADIAT/RADIN(25,21,16),RADIO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/EG(16),ET(16),EU(16),EV(16),FT(16),FU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(15),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),RI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,PLAT,RLANG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,ELEE,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .     PHIPIX,PHIPIY,QA,QB,HDIFH,RADIN,RADIO,RADSN,RADSO            COMMON.41
                                                                        COMMON.42
      AMOIST(X)=0.38*EXP(17.27*(X-273.16)/(X-35.86))                    DIFFV.10
      DMOIST(X)=4098.2/((X-35.86)*(X-35.86))                            DIFFV.11
      ZNASAV=ZNAUT                                                      NEW.12
      IF (GW(IS,JS).LT.0.0) ZNAUT=0.1*ZNASAV                            NEW.13
                                                                        DIFFV.12
C-----CALCULATE FACTOR, STORE OLD FACTOR                                DIFFV.13
      IF(IS.EQ.1) GO TO 220                                             DIFFV.14
```

```
      DO 200 K=2,KMAX
      FACTOL(JS,K)=FACTOR(JS,K)
  200 CONTINUE

C-----ADD SQUARED DENSITY FACTOR TO DIFFUSIVITIES
  220 CONS=(9.81/PSA(IS,JS))**2*DT
      TEMP1=(ALT(KMAX)*0.5+ZNAUT)/ZNAUT
      FACLOG=ALOG(TEMP1)
      DO 300 K=2,KMAX
      CC=TWT(K,2)*C(IS,JS,K-1)+TWT(K,1)*C(IS,JS,K)
      TT=TWT(K,2)*T(IS,JS,K-1)+TWT(K,1)*T(IS,JS,K)
      RPRIM=287.0*(1.0+RPDIF*CC)
      RHO=(PSA(IS,JS)*Q(K)+PTOP)/(RPRIM*TT)
      FACTOR(JS,K)=RHO*RHO*CONS*AKI(K)
  300 CONTINUE

C***************************************
C*       SCALARS                       *
C***************************************
C-----FORWARD PASS - TOP ZONE -  NO-FLUX BOUNDARY CONDITION
      FACBOT=FACTOR(JS,2)*DIFH(2)
      AA=FACBOT*DQI(1)
      BI=1.0/(1.0+AA)
      ET(1)=AA*BI
      FT(1)=(TB(IS,JS,1)+DT*TTEN(IS,JS,1))*BI
      FC(1)=(CB(IS,JS,1)+DT*CTEN(IS,JS,1))*BI
C------INTERIOR ZONES
      DO  400 K=2,KE
      CC=FACBOT*DQI(K)
      FACBOT=FACTOR(JS,K+1)*DIFH(K+1)
      AA=FACBOT*DQI(K)
      BB=1.0+AA+CC
      DI=1.0/(BB-CC*ET(K-1))
      ET(K)=AA*DI
      FT(K)=(TB(IS,JS,K)+DT*TTEN(IS,JS,K)+CC*FT(K-1))*DI
      FC(K)=(CB(IS,JS,K)+DT*CTEN(IS,JS,K)+CC*FC(K-1))*DI
  400 CONTINUE

      JSP1=JS+1
      IF(JE.EQ.1) JSP1=1
      RPRIM=287.0*(1.0+RPDIF*CG(IS,JS))
      RHOG=(PSA(IS,JS)+PTOP)*RHOFAC/(TG(IS,JS)*RPRIM)
      UAV=0.25*SQRT( (U(IS,JS,KMAX)+U(IS+1,JS,KMAX)+U(IS,JSP1,KMAX)
     .                  +U(IS+1,JSP1,KMAX))**2 +(V(IS,JS,KMAX)+V(IS+1,JS,
     .                  KMAX)+V(IS,JSP1,KMAX)+V(IS+1,JSP1,KMAX))**2 )
      UR=0.35/FACLOG*UAV
      C6=UR*RHOG/FACLOG *0.473
      C1=1004.0*C6
      C2=0.0
      C3=2.46E3*C1
C-----NEXT-TO-BOTTOM ZONE --- EMPLOY THE LAW OF THE WALL
      CC=FACBOT*DQI(KE)
      AA=C6*9.81/(PSA(IS,JS)*RHOFAC)*DT*DQI(KMAX)
      BB=1.0+CC+AA
      DI=1.0/(BB-CC*ET(KE))
```

```
      ET(KMAX)=AA*DI                                                   DIFFV.70
      FT(KMAX)=(TB(IS,JS,KMAX)+DI*TTEN(IS,JS,KMAX)+CC*FT(KE))*DI       DIFFV.71
      FC(KMAX)=(CB(IS,JS,KMAX)+DI*CTEN(IS,JS,KMAX)+CC*FC(KE))*DI       DIFFV.72
      PRATIO=((PSA(IS,JS)+PTOP)/PBOT)**0.286                           DIFFV.73
C-----CHECK FOR WATER SURFACE                                          DIFFV.74
      IF(GW(IS,JS).GE.0.0) GO TO 500                                   DIFFV.75
      FTG=AMOIST(TWATER)/(PSA(IS,JS)+PTOP)                             DIFFV.76
      CG(IS,JS)=FTG                                                    DIFFV.77
      CCG=CG(IS,JS)                                                    DIFFV.78
      TG(IS,JS)=TWATER                                                 DIFFV.79
      C4=0.0                                                           NEW.14
      C5=0.0                                                           NEW.15
      ETC=ET(KMAX)                                                     DIFFV.80
      GO TO 478                                                        DIFFV.81
  500 CONTINUE                                                         DIFFV.82
C-----GET READY TO CALCULATE THE SOIL FLUX (SELLERS,1965)             DIFFV.83
C     ( LOCALLY 1-D MODEL )                                            DIFFV.84
      ETA=GW(IS,JS)*(1.0-XSOIL)                                        DIFFV.85
      IF(ETA.LT.0.07) FS=20.0*ETA+0.293                                DIFFV.86
      IF(ETA.GE.0.07) FS=1.207+(7.50-8.00*ETA)*ETA                     DIFFV.87
      IF(ETA.LT.0.156) FCC=6.93*ETA+0.293                              DIFFV.88
      IF(ETA.GE.0.156) FCC=2.10*ETA+1.045                              DIFFV.89
      FO=0.408*ETA+0.02                                                DIFFV.90
      XLAM=(FS*XSAND+FCC*XCLAY+FO*XORG)/XSOIL                          DIFFV.91
      CONS=1.926E6*(XSAND+XCLAY)+2.51E6*XORG+4.185E6*ETA               DIFFV.92
C-----SOIL FLUX FROM BROOKS, M.W.R.,104,PP.793                        DIFFV.93
      ALNGTH=2.0*XLAM/(CONS*WW)                                        DIFFV.94
      EPS=0.405*XLAM*DELTAT/(CONS*ALNGTH)                              DIFFV.95
      EPS1=1.0+2.0*EPS                                                 DIFFV.96
      EPS2=0.215*EPS                                                   DIFFV.97
      ALNGTH=SQRT(ALNGTH)                                              DIFFV.98
      EPS3=1.274*XLAM/ALNGTH                                           DIFFV.99
      C4=EPS3/EPS1                                                     DIFFV.100
      C5=(STEMP(IS,JS)+EPS2*TDEEP)*C4                                  DIFFV.101
      C4=C4*(1.0+EPS2)                                                 DIFFV.102
                                                                       DIFFV.103
C-----ITERATE TO FIND THE SURFACE TEMPERATURE                          DIFFV.104
      M=0                                                              DIFFV.105
      M1=0                                                             DIFFV.106
      FTG=AMOIST(TG(IS,JS))/(PSA(IS,JS)+PTOP)                          DIFFV.107
      IF(CG(IS,JS).GT.C(IS,JS,KMAX)) GO TO 430                         DIFFV.108
      IF(FTG.LT.C(IS,JS,KMAX)) GO TO 440                               DIFFV.109
C---REGION OF ZERO FLUX AT THE SURFACE                                 DIFFV.110
      C3=0.0                                                           DIFFV.111
      ETC=0.0                                                          DIFFV.112
      BB=1.0+CC                                                        DIFFV.113
      VAPFAC=0.0                                                       DIFFV.114
      GO TO 450                                                        DIFFV.115
C----EVAPORIZATION OF SURFACE WATER                                    DIFFV.116
  430 VAPFAC=GW(IS,JS)                                                 DIFFV.117
      GO TO 445                                                        DIFFV.118
C----CONDENSATION ON THE SURFACE                                       DIFFV.119
  440 VAPFAC=1.0                                                       DIFFV.120
  445 CONTINUE                                                         DIFFV.121
      ETC=ET(KMAX)                                                     DIFFV.122
```

```
          BB=1.0+AA+CC                                                   DIFFV.123
      450 DT=1.0/(BB-CC*FT(KE))                                          DIFFV.124
          FC(KMAX)=(CB(IS,JS,KMAX)+DT*CTEN(IS,JS,KMAX)+CC*FC(KE))*DT     DIFFV.125
      455 FTG=AMOIST(TG(IS,JS))/(PSA(IS,JS)+PTOP)                        DIFFV.126
          CCG=VAPFAC*FTG                                                 DIFFV.127
          DCDT=DMOIST(TG(IS,JS))*CCG                                     DIFFV.128
          TEMPT=(RFLUX(KMAXP)-FLUXL2(KMAXP)+C1/PSA(IS,JS)*FT(KMAX)+C5    DIFFV.129
         .+C3*(FC(KMAX)/PSA(IS,JS)+(1.0-FTC)*(DCDT*TG(IS,JS)-CCG)))      DIFFV.130
         ./((1.0-ET(KMAX))*C1/PRATIO+(1.0-FTC)*C3*DCDT+C4)               DIFFV.131
          TEMRAT=TEMPT/TG(IS,JS)                                         DIFFV.132
          TG(IS,JS)=TEMPT                                                DIFFV.133
C     ALLOW FEW ADDITIONAL ITERATIONS TO ENSURE NO OSCILLATION ASSOCIATEDDIFFV.134
C     WITH INITIAL TG AND FLUXL2(KMAXP)                                  DIFFV.135
          IF(ICYCL.GT.1) GO TO 515                                       NEW.16
          M1=M1+1                                                        DIFFV.137
          IF(M1.LE.5.AND.IRAD.GT.0) CALL RADTER                          NEW.17
      515 CONTINUE                                                       DIFFV.139
          M=M+1                                                          DIFFV.140
          IF (ABS(TEMRAT-1.0) .LT. 1.0E-04) GO TO 460                    NEW.18
          IF(M.LT.20) GO TO 455                                          DIFFV.142
          CG(IS,JS)=FTG*GW(IS,JS)                                        DIFFV.143
          WRITE (6,67) M,IS,JS,TG(IS,JS),CG(IS,JS)                       DIFFV.144
       67 FORMAT (4H  M=,I5,5X,4H IS=,I5,5X,4H JS=,I5,5X,3HTG=,1PE12.4,5X, DIFFV.145
         .         3HCG=,1PE12.4 )                                       DIFFV.146
          STOP "TG"                                                      DIFFV.147
      460 CONTINUE                                                       DIFFV.148
          EVAP=C3*(CCG*(1.0-ETC)-FC(KMAX)/PSA(IS,JS))/2.47E6             DIFFV.149
          GW(IS,JS)=GW(IS,JS)+DELTAT*(PRECIP-EVAP)/(1.0E3*GDEPTH*(1.-XSOIL))DIFFV.150
          GW(IS,JS)=AMAX1(GW(IS,JS),0.0)                                 DIFFV.151
          GW(IS,JS)=AMIN1(GW(IS,JS),1.0)                                 DIFFV.152
          CG(IS,JS)=FTG*GW(IS,JS)                                        DIFFV.153
C-----CALCULATE THE SOIL TEMPERATURE                                     DIFFV.154
          STEMP(IS,JS)=(STEMP(IS,JS)+EPS*(TG(IS,JS)*1.785+TDEEP*0.215))/EPS1DIFFV.155
      478 TBB=PSA(IS,JS)*TG(IS,JS)/PRATIO                                DIFFV.156
          CBB=CCG*PSA(IS,JS)                                             DIFFV.157
          SAVE1(KMAX)=ET(KMAX)*TBB+FT(KMAX)                              DIFFV.158
          SAVE2(KMAX)=ETC*CBB+FC(KMAX)                                   DIFFV.159
C----CALCULATE THE TEMPERATURE AND WATER VAPOR (BACK SUBSTITUTION)       DIFFV.160
          DO 480 KK=1,KE                                                 DIFFV.161
          K=KMAX-KK                                                      DIFFV.162
          SAVE1(K)=ET(K)*SAVE1(K+1)+FT(K)                                DIFFV.163
          SAVE2(K)=ET(K)*SAVE2(K+1)+FC(K)                                DIFFV.164
      480 CONTINUE                                                       DIFFV.165
C-----SAVE1 AND SAVE2 ARE UPDATED TA AND CA                              DIFFV.166
          DO 508 K=1,KMAX                                                DIFFV.167
          TTEN(IS,JS,K)=(SAVE1(K)-TB(IS,JS,K))/DT                        DIFFV.168
          CTEN(IS,JS,K)=(SAVE2(K)-CB(IS,JS,K))/DT                        DIFFV.169
      508 CONTINUE                                                       DIFFV.170
C-----EVALUATE FLUXES AND STORE THEM                                     DIFFV.171
          FLUXCG=C3*(CCG*(1.0-ETC)-FC(KMAX)/PSA(IS,JS))                  DIFFV.172
          HHHH=C1*(TG(IS,JS)/PRATIO-SAVE1(KMAX)/PSA(IS,JS))              DIFFV.173
          ELEE=FLUXCG                                                    DIFFV.174
          GGGG=C4*TG(IS,JS)-C5                                           DIFFV.175
          RRRR=FLUXL2(KMAXP)                                             DIFFV.176
          SSSS=RFLUX(KMAXP)                                              DIFFV.177
```

```
      IF(GW(IS,JS).LT.0.0) 6666=SSSS-RRRR-HHHH-FLEE

C-----UPDATE RADTO (USED FOR EXTRAPOLATION) CORRESPONDING TO NEW INITIAL
C     FLUXL2(KMAXP)
      IF(ICYCL.GT.0) GO TO 779
      DO 778 K=1,KMAXP
  778 RADTO(IS,JS,K)=FLUXL2(K)
  779 CONTINUE

C-----ADD DIFFUSIVITIES FOR LATER PROGRAM EFFICIENCY
      DO 475 K=2,KMAX
      FACTOR(JS,K)=FACTOR(JS,K)*DIFM(K)
  475 CONTINUE

C*************************************
C*    VECTORS                        *
C*************************************
C-----SKIP VELOCITY CALCULATION IF IS=1 OR JS=1
      IF(JE.EQ.1) GO TO 511
      IF(IS.EQ.1.OR.JS.EQ.1) GO TO 600
  511 IF(IS.EQ.1) GO TO 600

C-----FORWARD PASS - TOP ZONE - NO FLUX
      FACBOT=0.25*(FACTOR(JS,2)+FACTOR(JS-1,2)+FACTOL(JS,2)
     +FACTOL(JS-1,2))
      IF(JS.EQ.1) FACBOT=0.5*(FACTOR(1,2)+FACTOL(1,2))
      AA=DQI(1)*FACBOT
      BI=1.0/(1.0+AA)
      EU(1)=AA*BI
      FU(1)=(UB(IS,JS,1)+DT*UTEN(IS,JS,1))*BI
      FV(1)=(VB(IS,JS,1)+DT*VTEN(IS,JS,1))*BI

C-----FORWARD PASS - INTERNAL ZONES
      DO 550 K=2,KEM
      FACTOP=FACBOT
      FACBOT=0.5*(FACTOR(JS,K+1)+FACTOL(JS,K+1))
      CC=DQI(K)*FACTOP
      AA=DQI(K)*FACBOT
      BB=1.0+AA+CC
      DD=1.0/(BB-CC*EU(K-1))
      EU(K)=AA*DD
      FU(K)=(UB(IS,JS,K)+DT*UTEN(IS,JS,K)+CC*FU(K-1))*DD
      FV(K)=(VB(IS,JS,K)+DT*VTEN(IS,JS,K)+CC*FV(K-1))*DD
  550 CONTINUE

C-----FORWARD PASS - NEXT TO BOTTOM ZONE USING LAW OF THE WALL
      TEMP2=((ALT(KMAX)+ALT(KE))*0.5+ZNAUT)/ZNAUT
      BLOGZ=FACLOG/ALOG(TEMP2)
      FACTOP=FACBOT
      FACBOT=0.5*(FACTOR(JS,KMAX)+FACTOL(JS,KMAX))
      CC=DQI(KE)*FACTOP
      AA=DQI(KE)*FACBOT
      BB=1.0+CC+AA*(1.0-BLOGZ)
      DD=1.0/(BB-CC*EU(KEM))
      FU(KE)=0.0
```

```
      FU(KE)=(UB(IS,JS,KE)+DT*UTEN(IS,JS,KE)+CC*FU(KEM))*DD                   DIFFV.233
      FV(KE)=(VB(IS,JS,KE)+DT*VTEN(IS,JS,KE)+CC*FV(KEM))*DD                   DIFFV.234
                                                                              DIFFV.235
C-----BACK SUBSTITUTE                                                          DIFFV.236
      DO 580 KK=2,KMAX                                                        DIFFV.237
      K=KMAXP-KK                                                              DIFFV.238
      SAVE3(K)=EU(K)*SAVE3(K+1)+FU(K)                                         DIFFV.239
      SAVE4(K)=EU(K)*SAVE4(K+1)+FV(K)                                         DIFFV.240
  580 CONTINUE                                                                DIFFV.241
      SAVE3(KMAX)=SAVE3(KE)*RLOGZ                                             DIFFV.242
      SAVE4(KMAX)=SAVE4(KE)*RLOGZ                                             DIFFV.243
C-----SAVE3 AND SAVE4 ARE UPDATED UA AND VA                                    DIFFV.244
      DO 509 K=1,KMAX                                                         DIFFV.245
      UTEN(IS,JS,K)=(SAVE3(K)-UB(IS,JS,K))/DT                                 DIFFV.246
      VTEN(IS,JS,K)=(SAVE4(K)-VB(IS,JS,K))/DT                                 DIFFV.247
  509 CONTINUE                                                                DIFFV.248
  600 CONTINUE                                                                DIFFV.249
      ZNAUT=ZNASAV                                                            NEW.19
                                                                              DIFFV.250
C-----DEBUG PRINT                                                              DIFFV.251
      IF(LOUT.LE.0) RETURN                                                    DIFFV.252
      IF(IJCOLM(IS,JS).LE.0) RETURN                                           DIFFV.253
      WRITE(6,700) IS,JS                                                      DIFFV.254
      WRITE(6,705)                                                            DIFFV.255
      DO 605 K=1,KMAX                                                         DIFFV.256
      WRITE (6,710) K,UTEN(IS,JS,K),VTEN(IS,JS,K),TTEN(IS,JS,K),              DIFFV.257
     .                 CTEN(IS,JS,K)                                          DIFFV.258
  605 CONTINUE                                                                DIFFV.259
      WRITE(6,730)                                                            DIFFV.260
      DO 610 K=1,KMAX                                                         DIFFV.261
      WRITE(6,710) K,FACTOR(K),EU(K),FU(K),FV(K),ET(K),FT(K),FC(K)            DIFFV.262
  610 CONTINUE                                                                DIFFV.263
      WRITE (6,735) TG(IS,JS),CG(IS,JS),GW(IS,JS)                             DIFFV.264
  700 FORMAT(22H0DIFFV OUTPUT, COLUMN=,I2,1H,,I2)                             DIFFV.265
  705 FORMAT(5H    K,9X,4HUTEN,9X,4HVTEN,9X,4HTTEN,9X,4HCTEN)                 DIFFV.266
  710 FORMAT(I5,1P8E13.5)                                                     DIFFV.267
  730 FORMAT(5H    K,7X,6HFACTOR,11X,2HEU,11X,2HFU,11X,2HFV,                  DIFFV.268
     .11X,2HET,11X,2HFT,11X,2HFC)                                             DIFFV.269
  735 FORMAT(4H TG=,1PE13.5,5X,4H CG=,1PE13.5,5X,4H GW=,1PE13.5)              DIFFV.270
                                                                              DIFFV.271
      RETURN                                                                  DIFFV.272
      END                                                                     DIFFV.273
```

```
      FUNCTION ENTERP(PXXX,N)                                           ENTERP.2
C***********************************************************************ENTERP.3
C*                                                                     *ENTERP.4
C*        "ENTERP" PERFORMS A LINEAR INTERPOLATION IN PRESSURE SPACE   *ENTERP.5
C*    ON A PROFILE WHICH IS DEFINED AT NPRES PRESSURE LEVELS.          *ENTERP.6
C*    THE VALUE RETURNED IS THAT FOR A PRESSURE LEVEL "PXXX". THE      *ENTERP.7
C*    PARTICULAR PROFILE USED IS DETERMINED BY THE VALUE OF "N".       *ENTERP.8
C*                                                                     *ENTERP.9
C***********************************************************************ENTERP.10
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),G6(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATELE.1
      COMMON/RADIAT/RADTN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),RI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COFFE1,COFFE2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,ELEE,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTMI,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,HTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .       PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO          COMMON.41
                                                                        COMMON.42
                                                                        ENTERP.12
      AMOIST(X)=0.38*EXP(17.27*X/(X+237.3))                             NEW.20
C                                                                       BATELE.3
C     FIRST PASS THRU ENTERP TO INITIALISE UPRES,VPRES,TPRES ETC.       BATELE.4
      IF(N.NE.5) GO TO 113                                              BATELE.5
```

```
      C                                                                  NEW.33
      C     HORIZONTAL WIND UPRES,VPRES IN COMPONENT FORMAT AND IN MPS   NEW.34
      C     X-AXIS ROTATION (ANGLE) MEASURED COUNTERCLOCKWISE FROM EAST  NEW.35
            DO 103 K=1,NPRES                                             NEW.36
        103 IF(ABS(THETA(K)).GT.0.0) GO TO 105                           BATELE.6
            COSANG=COS(ANGLE*0.01745329)                                 NEW.39
            SINANG=SIN(ANGLE*0.01745329)                                 NEW.40
            DO 104 K=1,NPRES                                             NEW.41
            UPRESK=UPRES(K)                                              BATELE.7
            VPRESK=VPRES(K)                                              BATELE.8
            UPRES(K)=(UPRESK*COSANG+VPRESK*SINANG)                       NEW.44
        104 VPRES(K)=(VPRESK*COSANG+UPRESK*SINANG)                       NEW.45
            GO TO 107                                                    NEW.46
      C                                                                  NEW.47
      C     INPUT DATA IN MAGNITUDE (UPRES)- DIRECTION (THETA) FORMAT, IN MPS NEW.48
      C     THETA MEASURED CLOCKWISE FROM NORTH                          NEW.49
        105 DO 106 K=1,NPRES                                             NEW.50
            UPRESK=UPRES(K)                                              BATELE.9
            THK=THETA(K)                                                 BATELE.10
            ANGL=THK+ANGLE                                               NEW.53
            VPRES(K)=-UPRESK*COS(ANGL*0.01745329)                        NEW.54
            IF(ANGL.GT. 89.9.AND.ANGL.LE. 90.1) VPRES(K)=0.0             NEW.55
            IF(ANGL.GT.269.9.AND.ANGL.LE.270.1) VPRES(K)=0.0             NEW.56
            UPRES(K)=-UPRESK*SIN(ANGL*0.01745329)                        NEW.57
        106 CONTINUE                                                     NEW.58
      C                                                                  NEW.59
      C     TEMPERATURE AND HUMIDITY                                     NEW.60
        107 CONTINUE                                                     BATELE.11
            DO 109 K=1,NPRES                                             NEW.65
        109 IF(RELHM(K).LT.0.0) GO TO 110                                BATELE.12
            DO 112 K=1,NPRES                                             BATELE.13
            CPRES(K)=RELHM(K)*AMOIST(TPRES(K))/(PPRES(K)*100.)           NEW.73
        112 CONTINUE                                                     NEW.76
        110 CONTINUE                                                     BATELE.14
            ENTERP=0.0                                                   NEW.78
            IF(N.EQ.5) RETURN                                            NEW.79
      C                                                                  BATELE.15
        113 CONTINUE                                                     NEW.80
            PXX1=PXXX                                                    ENTERP.13
            IF(PXXX.GT.PPRES(NPRES)) PXX1=PPRES(NPRES)                   ENTERP.14
            IF(PXXX.LT.PPRES(1)) PXX1=PPRES(1)                           ENTERP.15
      C-----SELECT THE PROFILE TO BE INTERPOLATED                        ENTERP.16
            GO TO(200,300,400,500), N                                    ENTERP.17
                                                                         ENTERP.18
      C-----U VELOCITY COMPONENT                                         ENTERP.19
        200 CONTINUE                                                     ENTERP.20
            IF(PXXX.GT.PCUT(IS,JS)) PXX1=PCUT(IS,JS)                     ENTERP.21
            DO 201 K=2,NPRES                                             ENTERP.22
            IF(PXX1.GT.PPRES(K)) GO TO 201                               ENTERP.23
            DPP=PPRES(K)-PPRES(K-1)                                      ENTERP.24
            DXX=UPRES(K)-UPRES(K-1)                                      ENTERP.25
            ENTERP=UPRES(K-1)+(PXX1-PPRES(K-1))*DXX/DPP                  ENTERP.26
            IF(PXXX.GT.PCUT(IS,JS)) ENTERP=ENTERP*                       ENTERP.27
           .((PUA(IS,JS)+PTOP-PXXX)/(PUA(IS,JS)+PTOP-PCUT(IS,JS)))**0.17 ENTERP.28
            RETURN                                                       ENTERP.29
```

```
  201 CONTINUE
      STOP "ENTU"

C-----V-VELOCITY COMPONENT
  300 CONTINUE
      IF(PXXX.GT.PCUT(IS,JS)) PXX1=PCUT(IS,JS)
      DO 301 K=2,NPRES
      IF(PXX1.GT.PPRES(K)) GO TO 301
      DPP=PPRES(K)-PPRES(K-1)
      DXX=VPRES(K)-VPRES(K-1)
      ENTERP=VPRES(K-1)+(PXX1-PPRES(K-1))*DXX/DPP
      IF(PXXX.GT.PCUT(IS,JS)) ENTERP=ENTERP*
     .((PUA(IS,JS)+PTOP-PXXX)/(PUA(IS,JS)+PTOP-PCUT(IS,JS)))**0.17
      RETURN
  301 CONTINUE
      STOP "ENTV"

C-----MOISTURE
  400 CONTINUE
      DO 401 K=2,NPRES
      IF(PXX1.GT.PPRES(K)) GO TO 401
      DPP=PPRES(K)-PPRES(K-1)
      DXX=CPRES(K)-CPRES(K-1)
      ENTERP=CPRES(K-1)+(PXX1-PPRES(K-1))*DXX/DPP
      RETURN
  401 CONTINUE
      STOP "ENTC"

C-----TEMPERATURE
  500 CONTINUE
      DO 501 K=2,NPRES
      IF(PXX1.GT.PPRES(K)) GO TO 501
      DPP=PPRES(K)-PPRES(K-1)
      DXX=TPRES(K)-TPRES(K-1)
      ENTERP=TPRES(K-1)+(PXX1-PPRES(K-1))*DXX/DPP+273.16
      RETURN
  501 CONTINUE
      STOP "ENTT"

      END
```

```
      SUBROUTINE FIELDS                                                 FIELDS.2
                                                                        FIELDS.3
C-----IN THIS ROUTINE FIELD VARIABLES SUCH AS U,V,C,T ARE CALCULATED    FIELDS.4
C-----FROM THE PRIMARY VARIABLES IN THE MODEL WHICH ARE UA,VA,CA AND TA FIELDS.5
                                                                        FIELDS.6
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATELE.1
      COMMON/RADIAT/RADIN(25,21,16),RADIO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),R(16),SCR(16),DIFN(16),DIFH(16),RI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XOPG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,ELEE,RRRR,SSSS,NPLT,NPLTT,SHREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KF,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,RINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IOHM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADIN,RADIO,RADSN,RADSO         COMMON.41
                                                                        COMMON.42
                                                                        FIELDS.8
C****************************************                               FIELDS.9
C*    SCALARS                          *                                FIELDS.10
C****************************************                               FIELDS.11
      DO 100 J=1,JE                                                     FIELDS.12
      DO 100 I=1,IE                                                     FIELDS.13
      TEMP=1.0/PSA(I,J)                                                 FIELDS.14
      DO 100 K=1,KMAX                                                   FIELDS.15
      PSIG=PSA(I,J)*A(K)+PTOP                                           FIELDS.16
```

```
      C(I,J,K)=CA(I,J,K)*TEMF
      T(I,J,K)=TA(I,J,K)*TEMF*(PSIG/PBOT)**0.286
  100 CONTINUE

C*************************************
C*    VELOCITY                       *
C*************************************
C-----INTERNAL POINTS
      DO 150 I=1,IEND
      DO 150 J=1,JEND
      DO 150 K=1,KMAX
      U(I,J,K)=UA(I,J,K)/PUA(I,J)
      V(I,J,K)=VA(I,J,K)/PUA(I,J)
  150 CONTINUE

C-----DEBUG PRINT
      IF(LOUT.LE.0) RETURN
      DO 500 J=1,JE
      DO 500 I=1,IE
      IF(IJCOLM(I,J).LE.0) GO TO 500
      WRITE(6,600) I,J
      WRITE(6,610)
      DO 450 K=1,KMAX
      WRITE(6,620) K,U(I,J,K),V(I,J,K),T(I,J,K),C(I,J,K)
  450 CONTINUE
      WRITE (6,630) PUA(I,J)
  500 CONTINUE
  600 FORMAT(23H0FIELDS OUTPUT, COLUMN=,I2,1H,,I2)
  610 FORMAT(6H    K,2*X,1HU,24X,1HV,24X,1HT,24X,1HC)
  620 FORMAT(I5,1P4E25.15)
  630 FORMAT(5H PUA=,1PE12.4)

      RETURN
      END
```

```
      SUBROUTINE FRICH                                                  FRICH.2
C***********************************************************************FRICH.3
C*                                                                     *FRICH.4
C*        "FRICH" EVALUATES THE HORIZONTAL DIFFUSION TERMS OCCURING IN *FRICH.5
C*     THE CONSERVATION EQUATIONS AND STORES THEM IN TEMPORARY STORAGE. *FRICH.6
C*     BOUNDARY CONDITIONS FOR THE SCALARS DEPEND ON THE SIGN OF THE    *FRICH.7
C*     BOUNDARY VELOCITY. IF THE FLOW IS ZERO OR DIRECTED OUT OF THE    *FRICH.8
C*     MESH. A NO FLUX CONDITION IS APPLIED. IF THE FLOW IS INWARD. THEN*FRICH.9
C*     A STEADY STATE AMBIENT VALUE (FUNCTION OF SIGMA) IS USED IN THE  *FRICH.10
C*     BOUNDARY FLUX CALCULATION. THE TURBULENT EXCHANGE COEFFICIENTS   *FRICH.11
C*     ARE ASSUMED A FUNCTION OF SIGMA ONLY. NOTE THAT THE "PI" FACTOR  *FRICH.12
C*     IS REMOVED FROM INSIDE THE LAPLACIAN OPERATOR A LA ANTHES. NOTE  *FRICH.13
C*     ALSO THAT FOR STABILITY REASONS. THE DIFFUSION TERMS MUST BE     *FRICH.14
C*     EVALUATED AT T=T(N-1).                                          *FRICH.15
C*                                                                     *FRICH.16
C***********************************************************************FRICH.17
      COMMON/ARRAY1/UA(26.22.15).VA(26.22.15).CA(25.21.15).TA(25.21.15) COMMON.2
      COMMON/ARRAY2/UB(26.22.15).VB(26.22.15).CB(25.21.15).TB(25.21.15) COMMON.3
      COMMON/ARRAY3/U(26.22.15).V(26.22.15).C(25.21.15).T(25.21.15)     COMMON.4
      COMMON/ARRAY4/UTEN(26.22.15).VTEN(26.22.15).CTEN(26.22.15).       COMMON.5
     .              TTEN(26.22.15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25.21.15).PHIPIX(25.21.15).PHIPIY(25.21.15)     COMMON.7
      COMMON/ARRAY6/QA(25.21.16).QB(25.21.16).HDIFH(25.21.15)           COMMON.8
      COMMON/ARRAY7/PSA(25.21).PSB(25.21).PTEN(25.21).HM(25.21).        COMMON.9
     .              PUA(26.22).PUB(26.22).PCUT(26.22).CONVAR(26.22)     COMMON.10
      COMMON/ARRAY8/TH(25.21).TG(25.21).STEMP(25.21).CG(25.21).         COMMON.11
     .              GW(25.21).HRATE(25.21).IJCOLM(25.21)                COMMON.12
      COMMON/ARRAY9/UPRES(30).VPRES(30).CPRES(30).TPRES(30).THETA(30).  COMMON.13
     .              PPRES(30).RELHM(30).UADAT(8).TRDAT(8)               BATELE.1
      COMMON/RADIAT/RADTN(25.21.16).RADTO(25.21.16).RADSN(25.21.16).    COMMON.15
     .              RADSO(25.21.16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21.16).FACTOL(21.16).SDOTR(21.16).           COMMON.17
     .              SDOTL(21.16)                                        COMMON.18
      COMMON/MATRIX/FC(16).FT(16).FU(16).FV(16).ET(16).EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16.2).Q(16).A(16).DQ(16).DQI(16).AKI(16).DZ(16).COMMON.20
     .              ALT(16).P(16).SCR(16).DIFM(16).DIFH(16).PI(16).     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16).SAVE2(16).SAVE3(16).SAVE4(16).SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16).FLUXL2(16).T4SIG(16).QEFF(16).            COMMON.24
     .              QEFFO.VEG.TRANS.COSANG.SINANG.RLAT.RLONG.DLONG.SS.  COMMON.25
     .              SA.HHH.FACTOM.COEFF1.COEFF2.COSZEN.DAYBEG.DLAT.     COMMON.26
     .              DAYNOW.AZMUTH.SIGMA.B.HRANGL.TANZEN.SINAZ.COSAZ.    COMMON.27
     .              SOLCON.SOLARN.IRAD.RTOLD.RTNEW.PRECIP.WW.XSAND.     COMMON.28
     .              XCLAY.XORG.XSOIL.GDEPTH.TDEEP.TWATER.GGGG.          COMMON.29
     .              HHHH.ELEF.RRRR.SSSS.NPLT.NPLTT.SUREM                COMMON.30
      COMMON/SCALAR/DT.DELTAT.ICYCL.SIMTMI.SIMTIM.NCYCLS.IEDIT.IPLOT.   COMMON.31
     .              ITAPE.KMAX.KMAXP.KE.KEM.PTOP.PBOT.UTOP.VTOP.        COMMON.32
     .              CTOP.TTOP.ZBOT.ZNAUT.XLN.IOUT.IFILM.LOUT.NPRES.F.   COMMON.33
     .              CPDIF.RPDIF.ZBL.RHOFAC.GAM.IE.IEND.IEM1.IS.KS.DX.   COMMON.34
     .              RDX.RDXH.RDX4.RDX16.RDXSQ.NOWDAY.ISUNH.ISUNM.ISUNS. COMMON.35
     .              GMTMHR.GMDYBG.DIF0.DIF1.DY.RDY.RDYH.RDY4.RDY16.     COMMON.36
     .              RDYSQ.JE.JEND.JEM1.JS.JBEG.ALPH.PINV.TNPRS.GAM1.    COMMON.37
     .              KBEG.KBGPL.ISTART.NCYCL.LTAPE.CX1.CX2.CX3.CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2.UA.VA.CA.TA.UB.VB.CB.TB.U.V.C.T.UTEN.VTEN.CTEN.TTEN.PHI.  COMMON.40
```

```
     .          PHIPIX,PHIPIY,QA,QB,HDIFH,RADIN,RADTO,RADSN,RADSO               COMMON.41
                                                                                COMMON.42
                DIMENSION ISTAB(7),DIFDAT(8,7),HT(8)                            FRICH.19
                                                                                FRICH.20
                DATA STBCLS/3./                                                 FRICH.21
                DATA ((DIFDAT(L,M),L=1,8),M=1,7)     /                          FRICH.22
               1   1.62,1.62,3.12,4.83,7.31,9.90,12.4,24.8,                     FRICH.23
               2   1.25,1.25,2.37,3.62,5.55,7.57,9.07,18.1,                     FRICH.24
               3   .848,.848,1.61,2.40,3.64,4.97,6.13,12.3,                     FRICH.25
               4   .527,.527,.956,1.34,2.09,2.78,3.43,6.86,                     FRICH.26
               5   .220,.220,.421,.554,.882,1.21,1.51,3.02,                     FRICH.27
               6   .088,.088,.161,.207,.321,.445,.559,1.12,                     FRICH.28
               7   .032,.032,.057,.069,.109,.150,.186,.372  /                   FRICH.29
                DATA HT/0.0,10.0,20.0,30.0,50.0,75.0,100.0,30000.0/             FRICH.30
                                                                                FRICH.31
                DO 252 K=1,KMAX                                                 FRICH.32
                DO 251 I=1,IE                                                   FRICH.33
                DO 251 J=1,JE                                                   FRICH.34
                UTEN(I,J,K)=0.0                                                 FRICH.35
                VTEN(I,J,K)=0.0                                                 FRICH.36
                TTEN(I,J,K)=0.0                                                 FRICH.37
                QTEN(I,J,K)=0.0                                                 FRICH.38
            251 CONTINUE                                                        FRICH.39
                DO 253 I=1,IEND                                                 FRICH.40
                UTEN(I,JEND)=0.0                                                FRICH.41
            253 VTEN(I,JEND)=0.0                                                FRICH.42
                DO 252 J=1,JE                                                   FRICH.43
                UTEN(IEND,J)=0.0                                                FRICH.44
                VTEN(IEND,J)=0.0                                                FRICH.45
            252 CONTINUE                                                        FRICH.46
                                                                                FRICH.47
          C-----SET UP FIELD OF DIFFUSIVITIES                                   FRICH.48
                IF(ICYCL.GT.1) GO TO 99                                         NEW.81
                ALTMAX=0.0                                                      FRICH.50
                DO 250 J=1,JE                                                   FRICH.51
                JP1=J+1                                                         FRICH.52
                IF(JE.EQ.1) JP1=1                                               FRICH.53
                DO 250 I=1,IE                                                   FRICH.54
                IP1=I+1                                                         FRICH.55
                TEMP=29.26*PSA(I,J)                                             FRICH.56
                ALT(KMAXP)=0.0                                                  FRICH.57
                DO 1 KK=1,KMAX                                                  FRICH.58
                K=KMAXP-KK                                                      FRICH.59
                RPRIM=1.0+RPDIF*Q(I,J,K)                                        FRICH.60
                ALT(K)=ALT(K+1)+TEMP*RPRIM*T(I,J,K)/(PSA(I,J)*A(K)+PTOP)*DQ(K)  FRICH.61
                ALTMAX=AMAX1(ALTMAX,ALT(K))                                     FRICH.62
              1 CONTINUE                                                        FRICH.63
                DO 2 K=1,KMAX                                                   FRICH.64
              2 DZ(K) = ALT(K)-ALT(K+1)                                         FRICH.65
                IS1 = STBCLS+0.1                                                FRICH.66
                FACT=1.0                                                        FRICH.67
                IF(IS1.GT.5) FACT=25.0                                          FRICH.68
                IF(IS1.EQ.5) FACT=10.0                                          FRICH.69
                IL = 1                                                          FRICH.70
                ZB = HT(IL)                                                     FRICH.71
```

```
      TBB = DIFDAT(IL,IS1)                                                 FRICH.72
      ZT = HT(IL+1)                                                        FRICH.73
      TDOP = DIFDAT(IL+1,IS1)                                              FRICH.74
      SLOPE = (TDOP-TBB)/(ZT-ZB)                                           FRICH.75
      DO 8 KK=1,KMAX                                                       FRICH.76
      K=KMAXP-KK                                                           FRICH.77
      ZZZ=ALT(K)-0.5*DZ(K)                                                 FRICH.78
    4 IF(ZZZ.GT.ZT) GO TO 6                                                FRICH.79
      DUMXX=SQRT((U(I,J,K)+U(IP1,J,K)+U(I,JP1,K)+U(IP1,JP1,K))**2          FRICH.80
     .      +(V(I,J,K)+V(IP1,J,K)+V(I,JP1,K)+V(IP1,JP1,K))**2)  *0.25      FRICH.81
      HDIFH(I,J,K)=TBB+(ZZZ-ZB)*SLOPE                                      FRICH.82
      HDIFH(I,J,K)=HDIFH(I,J,K)*DUMXX*FACT                                 FRICH.83
C-----HDIFH IS CENTERED AT (T,C) POINTS                                    FRICH.84
      HDIFH(I,J,K)=AMAX1(HDIFH(I,J,K),200.0)                               FRICH.85
      GO TO 8                                                              FRICH.86
    6 TBB = TDOP                                                           FRICH.87
      ZB = ZT                                                              FRICH.88
      IL = IL+1                                                            FRICH.89
      IF(IL.GT.7) STOP "FRH1"                                              FRICH.90
      ZT = HT(IL+1)                                                        FRICH.91
      TDOP = DIFDAT(IL+1,IS1)                                              FRICH.92
      SLOPE =(TDOP-TBB)/(ZT-ZB)                                            FRICH.93
      GO TO 4                                                              FRICH.94
    8 CONTINUE                                                             FRICH.95
  250 CONTINUE                                                             FRICH.96
                                                                           FRICH.97
C-----MAX. TIME STEP BASED ON THE MAX. SPEED OF THE GRAVITY WAVE           FRICH.98
      DTFRAC=0.80                                                          FRICH.99
      DXX=AMAX1(DX,DY)                                                     FRICH.100
      DTMAX=DTFRAC*0.5*DXX/SQRT(9.81*ALTMAX)                               FRICH.101
      IF(DELTAT.LE.DTMAX) GO TO 99                                         FRICH.102
      WRITE (6,98) DELTAT,DTMAX                                            FRICH.103
   98 FORMAT(/67H ****TIME STEP TOO LARGE BASED ON LINEARISED GRAVITY WAFRICH.104
     .VE CRITERION/5X,19HORIGINAL TIME STEP=,F6.2,5X,19HADJUSTED TIME STEFRICH.105
     .EP=,F6.2,4H****/)                                                    FRICH.106
      DELTAT=DTMAX                                                         FRICH.107
      DT=2.0*DELTAT                                                        FRICH.108
   99 CONTINUE                                                             FRICH.109
                                                                           FRICH.110
C****************************************                                  FRICH.111
C*    HORIZONTAL MOMENTUM DIFFUSION     *                                  FRICH.112
C****************************************                                  FRICH.113
      DO 400 K=1,KMAX                                                      FRICH.114
C-----X DIRECTION                                                          FRICH.115
      DO 350 J=JBEG,JE                                                     FRICH.116
      JM1=J-1                                                              FRICH.117
      IF(J.EQ.1) JM1=1                                                     FRICH.118
      DIFR=0.5*(HDIFH(1,J,K)+HDIFH(1,JM1,K))                               FRICH.119
      FLX1=DIFR*(U(2,J,K)-U(1,J,K))                                        FRICH.120
      FLX3=DIFR*(V(2,J,K)-V(1,J,K))                                        FRICH.121
      DO 350 I=2,IE                                                        FRICH.122
      DIFR=0.5*(HDIFH(I,J,K)+HDIFH(I,JM1,K))                               FRICH.123
      FLX2=DIFR*(U(I+1,J,K)-U(I,J,K))                                      FRICH.124
      FLX4=DIFR*(V(I+1,J,K)-V(I,J,K))                                      FRICH.125
      UTEN(I,J,K)=UTEN(I,J,K)+PUB(I,J)*(FLX2-FLX1)*RDXSQ                   FRICH.126
```

```
      VTEN(I,J,K)=VTEN(I,J,K)+PUF(I,J)*(FLX4-FLX3)*RDXSQ                FRICH.127
      FLX1=FLX2                                                         FRICH.128
      FLX3=FLX4                                                         FRICH.129
  350 CONTINUE                                                          FRICH.130
                                                                        FRICH.131
C-----Y DIRECTION                                                       FRICH.132
      IF(JE.EQ.1) GO TO 301                                             FRICH.133
      DO 300 I=2,IE                                                     FRICH.134
      DIFR=0.5*(HDIFH(I,1,K)+HDIFH(I-1,1,K))                            FRICH.135
      FLX1=DIFR*(U(I,2,K)-U(I,1,K))                                     FRICH.136
      FLX3=DIFR*(V(I,2,K)-V(I,1,K))                                     FRICH.137
      DO 300 J=2,JE                                                     FRICH.138
      DIFR=0.5*(HDIFH(I,J,K)+HDIFH(I-1,J,K))                            FRICH.139
      FLX2=DIFR*(U(I,J+1,K)-U(I,J,K))                                   FRICH.140
      FLX4=DIFR*(V(I,J+1,K)-V(I,J,K))                                   FRICH.141
      UTEN(I,J,K)=UTEN(I,J,K)+PUB(I,J)*(FLX2-FLX1)*RDYSQ                FRICH.142
      VTEN(I,J,K)=VTEN(I,J,K)+PUB(I,J)*(FLX4-FLX3)*RDYSQ                FRICH.143
      FLX1=FLX2                                                         FRICH.144
      FLX3=FLX4                                                         FRICH.145
  300 CONTINUE                                                          FRICH.146
  301 CONTINUE                                                          FRICH.147
  400 CONTINUE                                                          FRICH.148
                                                                        FRICH.149
C****************************************                               FRICH.150
C*      HORIZONTAL SCALAR DIFFUSION         *                           FRICH.151
C****************************************                               FRICH.152
      DO 600 K=1,KMAX                                                   FRICH.153
      DO 551 I=1,IE                                                     FRICH.154
      DO 551 J=1,JE                                                     FRICH.155
      TH(I,J)=TB(I,J,K)/PSB(I,J)                                        FRICH.156
  551 CONTINUE                                                          FRICH.157
C-----X DIRECTION                                                       FRICH.158
      DO 550 J=1,JE                                                     FRICH.159
      JP1=J+1                                                           FRICH.160
      IF(JE.EQ.1) JP1=1                                                 FRICH.161
      FLX1=0.0                                                          FRICH.162
      FLX3=0.0                                                          FRICH.163
      IF(UB(1,J,K)+UB(1,JP1,K).LE.0.0) GO TO 510                        FRICH.164
      PSIG=PSB(1,J)*A(K)+PTOP                                           FRICH.165
      KS=K                                                              FRICH.166
      JS=J                                                              FRICH.167
      IS=1                                                              FRICH.168
      CBB=ENTERP(PSIG,3)                                                FRICH.169
      TBB=ENTERP(PSIG,4)                                                FRICH.170
      TBB=TBB*(PRDT/PSIG)**0.286                                        FRICH.171
      FLX1=HDIFH(1,J,K)*(C(1,J,K)-CBB)                                  FRICH.172
      FLX3=HDIFH(1,J,K)*(TH(1,J)-TBB)                                   FRICH.173
  510 DO 520 I=1,IEM1                                                   FRICH.174
      IP1=I+1                                                           FRICH.175
      DIFR=0.5*(HDIFH(IP1,J,K)+HDIFH(I,J,K))                            FRICH.176
      FLX2=DIFR*(C(IP1,J,K)-C(I,J,K))                                   FRICH.177
      FLX4=DIFR*(TH(IP1,J)-TH(I,J))                                     FRICH.178
      CTEN(I,J,K)=CTEN(I,J,K)+PSB(I,J)*(FLX2-FLX1)*RDXSQ                FRICH.179
      TTEN(I,J,K)=TTEN(I,J,K)+PSB(I,J)*(FLX4-FLX3)*RDXSQ                FRICH.180
      FLX1=FLX2                                                         FRICH.181
```

```
      FLX3=FLX4
  520 CONTINUE
      FLX2=0.0
      FLX4=0.0
      IF(UB(IEND,J,K)+UB(IEND,JP1,K).GE.0.0) GO TO 530
      PSIG=PSB(IE,J)*A(K)+PTOP
      IS=IE
      CBB=ENTERP(PSIG,3)
      TBB=ENTERP(PSIG,4)
      TBB=TBB*(PBOT/PSIG)**0.286
      FLX2=HDIFH(IE   ,J,K)*(CBB-C(IE,J,K))
      FLX4=HDIFH(IE   ,J,K)*(TBB-TH(IE,J))
  530 CTEN(IE,J,K)=CTEN(IE,J,K)+PSB(IE,J)*(FLX2-FLX1)*RDXSQ
      TTEN(IE,J,K)=TTEN(IE,J,K)+PSB(IE,J)*(FLX4-FLX3)*RDXSQ
  550 CONTINUE

C-----Y DIRECTION
      IF(JE.EQ.1) GO TO 501
      DO 500 I=1,IE
      IP1=I+1
      FLX1=0.0
      FLX3=0.0
      IF(VB(I,1,K)+VB(IP1,1,K).LE.0.0) GO TO 450
      PSIG=PSB(I,1)*A(K)+PTOP
      IS=I
      JS=1
      CBB=ENTERP(PSIG,3)
      TBB=ENTERP(PSIG,4)
      TBB=TBB*(PBOT/PSIG)**0.286
      FLX1=HDIFH(I,1,K)*(C(I,1,K)-CBB)
      FLX3=HDIFH(I,1,K)*(TH(I,1)-TBB)
  450 DO 460 J=1,JEM1
      JP1=J+1
      DIFR=0.5*(HDIFH(I,JP1,K)+HDIFH(I,J,K))
      FLX2=DIFR*(C(I,JP1,K)-C(I,J,K))
      FLX4=DIFR*(TH(I,JP1)-TH(I,J))
      CTEN(I,J,K)=CTEN(I,J,K)+PSB(I,J)*(FLX2-FLX1)*RDYSQ
      TTEN(I,J,K)=TTEN(I,J,K)+PSB(I,J)*(FLX4-FLX3)*RDYSQ
      FLX1=FLX2
      FLX3=FLX4
  460 CONTINUE
      FLX2=0.0
      FLX4=0.0
      IF(VB(I,JEND,K)+VB(IP1,JEND,K).GE.0.0) GO TO 465
      PSIG=PSB(I,JE)*A(K)+PTOP
      JS=JE
      CBB=ENTERP(PSIG,3)
      TBB=ENTERP(PSIG,4)
      TBB=TBB*(PBOT/PSIG)**0.286
      FLX2=HDIFH(I,JE   ,K)*(CBB-C(I,JE,K))
      FLX4=HDIFH(I,JE   ,K)*(TBB-TH(I,JE))
  465 CTEN(I,JE,K)=CTEN(I,JE,K)+PSB(I,JE)*(FLX2-FLX1)*RDYSQ
      TTEN(I,JE,K)=TTEN(I,JE,K)+PSB(I,JE)*(FLX4-FLX3)*RDYSQ
  500 CONTINUE
  501 CONTINUE
```

```
  600 CONTINUE

C-----DEBUG PRINT
      IF(LOUT.LE.0) RETURN
      DO 700 J=1,JE
      DO 700 I=1,IE
      IF(IJCOLM(I,J).LE.0) GO TO 700
      WRITE(6,705) I,J
      WRITE(6,710)
      DO 690 K=1,KMAX
      WRITE(6,720) K,UTEN(I,J,K),VTEN(I,J,K),TTEN(I,J,K),CTEN(I,J,K),
     .UB(I,J,K),VB(I,J,K),TB(I,J,K),CB(I,J,K),HDIFH(I,J,K)
  690 CONTINUE
      WRITE (6,730) PSB(I,J),PUB(I,J)
  700 CONTINUE
  705 FORMAT(22H FRICH OUTPUT, COLUMN=,I2,1H,,I2)
  710 FORMAT(6H     K,8X,4HUTEN,9X,4HVTEN,9X,4HTTEN,9X,4HCTEN,11X,2HUB,
     .11X,2HVB,11X,2HTB,11X,2HCB,9X,4HHDIF)
  720 FORMAT(I5,1P9E13.5)
  730 FORMAT(5H PSB=,1PE12.4,4HPUB=,1PE12.4)
      RETURN
      END
```

```
      SUBROUTINE FRICV                                                  FRICV.2
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .             TTEN(26,22,15)                                       COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .             PUA(26,22),PUB(26,22),PGHT(26,22),CONVAR(26,22)      COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .             GW(25,21),HRATE(25,21),IJCOLM(25,21)                 COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .             PPRES(30),RELHM(30),UADAT(8),TRDAT(8)                BATELE.1
      COMMON/RADIAT/RADTN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .             RADSO(25,21,16)                                      COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .             SDOTL(21,16)                                         COMMON.18
      COMMON/MATRIX/EC(16),ET(16),EU(16),FV(16),FT(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .             ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),PI(16),      COMMON.21
     .             DVELDZ(16)                                           COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .             QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,   COMMON.25
     .             SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,      COMMON.26
     .             DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,     COMMON.27
     .             SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,      COMMON.28
     .             XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,           COMMON.29
     .             HHHH,ELEF,RRRR,SSSS,NPLT,NPLTT,SUREM                 COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTIM,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .             ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,         COMMON.32
     .             CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,    COMMON.33
     .             CPDIF,PPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,    COMMON.34
     .             RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS,  COMMON.35
     .             GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,      COMMON.36
     .             RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,     COMMON.37
     .             KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4        COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO         COMMON.41
                                                                        COMMON.42
C----------VERTICAL EDDY COEFFICIENTS ACCORDING TO MELLOR - YAMADA      FRICV.4
C          LEVEL   2 1/2 FORMULATION    (J.A.S.,32,PP.2309)             FRICV.5
                                                                        FRICV.6
      ZNASAV=ZNAUT                                                      NEW.82
      IF (GW(IS,JS).LT.0.0) ZNAUT=0.1*ZNASAV                            NEW.83
      DO 201 K=1,KMAXP                                                  NEW.84
      DIFM(K)=DIF0                                                      NEW.85
      DIFH(K)=DIF1                                                      NEW.86
  201 CONTINUE                                                          NEW.87
      IF(LOUT.GT.0.AND.IJCOLM(IS,JS).GT.0) WRITE(6,800) IS,JS           FRICV.7
C-----CALCULATE ALTITUDES                                               FRICV.8
      TEMP=29.26*PSA(IS,JS)                                             FRICV.9
      ALT(KMAXP)=0.0                                                    FRICV.10
```

```
      DO 20 KK=1,KMAX                                                 FRICV.11
      K=KMAXP-KK                                                      FRICV.12
      RPRIM=1.0+RPDIF*C(IS,JS,K)                                      FRICV.13
      ALT(K)=ALT(K+1)+TEMP*RPRIM*T(IS,JS,K)/(PSA(IS,JS)*A(K)+PTOP)*DQ(K)FRICV.14
   20 CONTINUE                                                        FRICV.15
                                                                      FRICV.16
      FACTK=0.35                                                      FRICV.17
      ISP1=IS+1                                                       FRICV.18
      JSP1=JS+1                                                       FRICV.19
      IF(JE.EQ.1) JSP1=1                                              FRICV.20
                                                                      FRICV.21
C-----DETERMINE THE VERTICAL SHEAR AND RICHARDSON NUMBER               FRICV.22
      DO 260 K=2,KMAX                                                 FRICV.23
      PP=PSA(IS,JS)*Q(K)+PTOP                                         FRICV.24
      TT=TWT(K,1)*T(IS,JS,K)+TWT(K,2)*T(IS,JS,K-1)                    FRICV.25
      CC=TWT(K,1)*C(IS,JS,K)+TWT(K,2)*C(IS,JS,K-1)                    FRICV.26
      FACT=0.0342*PP/(TT*PSA(IS,JS))                                  FRICV.27
      DUDZ=(U(IS,JS,K-1)+U(ISP1,JS,K-1)+U(IS,JSP1,K-1)+U(ISP1,JSP1,K-1)-FRICV.28
     .U(IS,JS,K)-U(ISP1,JS,K)-U(IS,JSP1,K)-U(ISP1,JSP1,K))/(A(K-1)-A(K))FRICV.29
      DVDZ=(V(IS,JS,K-1)+V(ISP1,JS,K-1)+V(IS,JSP1,K-1)+V(ISP1,JSP1,K-1)-FRICV.30
     .V(IS,JS,K)-V(ISP1,JS,K)-V(IS,JSP1,K)-V(ISP1,JSP1,K))/(A(K-1)-A(K))FRICV.31
      DVELDZ(K)=FACT*SQRT(DUDZ*DUDZ+DVDZ*DVDZ)*0.25                   FRICV.32
      DVELDZ(K)=AMAX1(1.0E-20,DVELDZ(K))                              FRICV.33
      DTDZ=0.01/(1.0+0.88*CC)+FACT*(T(IS,JS,K-1)-T(IS,JS,K)+0.61*TT   FRICV.34
     .*(C(IS,JS,K-1)-C(IS,JS,K)))/(A(K)-A(K-1))                       FRICV.35
      RI(K)=9.81*DTDZ/(DVELDZ(K)*DVELDZ(K)*TT)                        FRICV.36
  260 CONTINUE                                                        FRICV.37
C-----RICHARDSON NUMBER AT THE TOP                                    FRICV.38
      FACT=0.0342*PTOP/(TTOP*PSA(IS,JS))                              FRICV.39
      DUDZ=(U(IS,JS,1)+U(ISP1,JS,1)+U(IS,JSP1,1)+U(ISP1,JSP1,1)-4.0*UTOPFRICV.40
     .)/A(1)                                                          FRICV.41
      DVDZ=(V(IS,JS,1)+V(ISP1,JS,1)+V(IS,JSP1,1)+V(ISP1,JSP1,1)-4.0*VTOPFRICV.42
     .)/A(1)                                                          FRICV.43
      DVELDZ(1)=FACT*SQRT(DUDZ*DUDZ+DVDZ*DVDZ)*0.25                   FRICV.44
      DVELDZ(1)=AMAX1(1.0E-20,DVELDZ(1))                              FRICV.45
      DTDZ=0.01/(1.0+0.88*CTOP)+FACT*(TTOP-T(IS,JS,1)+0.61*TTOP*(CTOP FRICV.46
     .-C(IS,JS,1)))/A(1)                                              FRICV.47
      RI(1)=9.81*DTDZ/(DVELDZ(1)*DVELDZ(1)*TTOP)                      FRICV.48
C-----RICHARDSON NUMBER AT THE BOTTOM                                 FRICV.49
      FACT=0.0342*(PSA(IS,JS)+PTOP)/(TG(IS,JS)*PSA(IS,JS))            FRICV.50
      DUDZ=(U(IS,JS,KMAX)+U(ISP1,JS,KMAX)+U(IS,JSP1,KMAX)+U(ISP1,JSP1,FRICV.51
     .KMAX))/(A(KMAX)-1.0)                                            FRICV.52
      DVDZ=(V(IS,JS,KMAX)+V(ISP1,JS,KMAX)+V(IS,JSP1,KMAX)+V(ISP1,JSP1,FRICV.53
     .KMAX))/(A(KMAX)-1.0)                                            FRICV.54
      DVELDZ(KMAXP)=FACT*SQRT(DUDZ*DUDZ+DVDZ*DVDZ)*0.25               FRICV.55
      DVELDZ(KMAXP)=AMAX1(1.0E-20,DVELDZ(KMAXP))                      FRICV.56
      DTDZ=0.01/(1.0+0.88*CG(IS,JS))+FACT*(T(IS,JS,KMAX)-TG(IS,JS)+0.61*FRICV.57
     .TG(IS,JS)*(C(IS,JS,KMAX)-CG(IS,JS)))/(1.0-A(KMAX))              FRICV.58
      RI(KMAXP)=9.81*DTDZ/(DVELDZ(KMAXP)*DVELDZ(KMAXP)*TG(IS,JS))     FRICV.59
                                                                      FRICV.60
C----CALCULATE THE MACROLENGTH                                        FRICV.61
      DO 280 K=1,KMAX                                                 FRICV.62
      SAVE1(K)=ALT(K)-ALT(K+1)                                        FRICV.63
      SAVE4(K)=0.5*(ALT(K)+ALT(K+1))                                  FRICV.64
  280 CONTINUE                                                        FRICV.65
```

```
      X1=0.0                                                            FRICV.66
      X2=0.0                                                            FRICV.67
      DO 266 K=1, KMAX                                                  FRICV.68
      TEMP = (QA(IS,JS,K)+QA(IS,JS,K+1))*0.5                            FRICV.69
      SAVE2(K) = SQRT(TEMP)                                             FRICV.70
      X1=X1+SAVE4(K)*TEMP                                               FRICV.71
      X2=X2+TEMP                                                        FRICV.72
  266 CONTINUE                                                          FRICV.73
      IF (X2.LE.0.0) GO TO 267                                          FRICV.74
C-----RATIO OF MOMENTS IS 0.10 (YAMADA AND MELLOR)                      FRICV.75
      XLN= 0.1*X1/X2                                                    FRICV.76
  267 XLN = AMAX1(10.0,XLN)                                             FRICV.77
      ZLAMI = 1.0/XLN                                                   FRICV.78
                                                                        FRICV.79
C-----EVALUATE DIFFUSIVITIES FROM QSQD                                  FRICV.80
      FLXIN=0.0                                                         FRICV.81
      DO 400 K=2,KMAX                                                   FRICV.82
      FACT=FACTK*(ALT(K)+ZNAUT)                                         FRICV.83
      ZL=FACT/(1.0+FACT/XLN)                                            FRICV.84
      D2=ZL*ZL                                                          FRICV.85
      IF(QA(IS,JS,K).GT. 1.0E-20) GO TO 350                             FRICV.86
      SAVE1(K)=0.0                                                      FRICV.87
      SAVE3(K)=0.0                                                      FRICV.88
      SAVE4(K)=0.0                                                      FRICV.89
      DIFH(K)=0.0                                                       FRICV.90
      DIFM(K)=0.0                                                       FRICV.91
      GO TO 400                                                         FRICV.92
  350 YCOEF=QA(IS,JS,K)/(DVELDZ(K)*DVELDZ(K)*D2)                        FRICV.93
      D0 = RI(K)/YCOEF                                                  FRICV.94
      IF(D0 .LT.-0.337) D0 = -0.337                                     FRICV.95
      D1 = DVELDZ(K)*SQRT(YCOEF)                                        FRICV.96
      SAVE4(K)=D1                                                       FRICV.97
      SAVE3(K)=D1*D2                                                    FRICV.98
      ALF1 = 1.0/(1.0+12.64*D0)                                         FRICV.99
      ALF2 = 1.0/(1.0+3.88206*D0)                                       FRICV.100
      ALF3 = 1.0-3.93183*ALF1*D0                                        FRICV.101
      TT1 =       0.333333+ 0.2044224*ALF2/YCOEF                        FRICV.102
      TT2 = 1.04/D1                                                     FRICV.103
      TT3 = 3.6504*(ALF2*ALF3/YCOEF + ALF1*D0*1.418)                    FRICV.104
      ZKP = ALT(K+1)                                                    FRICV.105
      ZKM = ALT(K-1)                                                    FRICV.106
      FLXOUT=0.0                                                        FRICV.107
      IF(K.EQ.KMAX) GO TO 346                                           FRICV.108
      FACTP = FACTK*(ZKP+ZNAUT)                                         FRICV.109
      ZLP = FACTP/(1.0+FACTP/XLN)                                       FRICV.110
      ALAM = 0.115*(ZL+ZLP)                                             FRICV.111
      ZK = ALT(K)                                                       FRICV.112
      FLXOUT = SAVE2(K)*ALAM*(QA(IS,JS,K+1)-QA(IS,JS,K))/(ZKP-ZK)       FRICV.113
  346 DFUNCT = 2.0*(FLXOUT-FLXIN)/(ZKP-ZKM)                             FRICV.114
      WSQD = (TT1*QA(IS,JS,K)+TT2*DFUNCT)/(1.0+TT3)                     FRICV.115
      WSQD = AMAX1(0.0,AMIN1(WSQD,0.5*QA(IS,JS,K)))                     FRICV.116
      DIFH(K)= ALF1*WSQD*2.37/D1                                        FRICV.117
      DIFM(K)= 2.34*ALF2*(ALF3*WSQD-0.056*QA(IS,JS,K))/D1               FRICV.118
      SAVE1(K) = WSQD                                                   FRICV.119
      SAVE2(K) = DFUNCT                                                 FRICV.120
```

```
      SAVE5(K) = YCOEF                                                 FRICV.121
      FLXIN = FLXOUT                                                   FRICV.122
  400 CONTINUE                                                         FRICV.123
                                                                       FRICV.124
C-----SET UP COEFFS FOR THE TRIDIAGONAL SOLUTION                       FRICV.125
      FC(2) = 0.0                                                      FRICV.126
      DO 500 K=2,KE                                                    FRICV.127
      TEMP = 2.0*DT/(ALT(K+1)-ALT(K-1))                                FRICV.128
      FU(K) = 0.1917*(SAVE3(K+1)+SAVE3(K))/(ALT(K)-ALT(K+1))           FRICV.129
      FC(K+1) = FU(K)                                                  FRICV.130
      FU(K) = FU(K)*TEMP                                               FRICV.131
      FC(K) = FC(K)*TEMP                                               FRICV.132
      FT(K) = 1.0+0.2667*DELTAT*SAVE4(K)- FC(K) - FU(K)                FRICV.133
      FV(K) = QB(IS,JS,K)+2.0*DT*(DVELDZ(K)*DVELDZ(K)*                 FRICV.134
     .     (DIFM(K)-RI(K)*DIFH(K)))                                    FRICV.135
  500 CONTINUE                                                         FRICV.136
C-----EVALUATE LOWER BOUNDARY VALUES FOR QA                             FRICV.137
      TEMP = -2.0*DT/ALT(KE)                                           FRICV.138
      FC(KMAX) = FC(KMAX)*TEMP                                         FRICV.139
      TEMP1 = (ALT(KMAX)*0.5 + ZNAUT)/ZNAUT                            FRICV.140
      FACLOG = ALOG(TEMP1)                                             FRICV.141
      URSQ=0.00765625/(FACLOG*FACLOG)*((U(IS,JS,KMAX)+U(ISP1,JS,KMAX)  FRICV.142
     .+U(IS,JSP1,KMAX)+U(ISP1,JSP1,KMAX))**2 +(V(IS,JS,KMAX)+V(ISP1,JS, FRICV.143
     .KMAX)+V(IS,JSP1,KMAX)+V(ISP1,JSP1,KMAX))**2)                     FRICV.144
      QA(IS,JS,KMAXP) = 6.10*URSQ                                      FRICV.145
      FACT=FACTK*TEMP1*ZNAUT                                           FRICV.146
      ZL=FACT/(1.0+FACT/XLN)                                           FRICV.147
      FU(KMAX)=0.3833*ZL*SQRT(QA(IS,JS,KMAXP))/ALT(KMAX)*TEMP          FRICV.148
      FT(KMAX)=1.0+0.2667*DELTAT*SAVE4(KMAX)-FC(KMAX)-FU(KMAX)         FRICV.149
      FV(KMAX)=QB(IS,JS,KMAX)+2.0*DT*DVELDZ(KMAX)*DVELDZ(KMAX)         FRICV.150
     .*(DIFM(KMAX)-RI(KMAX)*DIFH(KMAX))-FU(KMAX)*QA(IS,JS,KMAXP)       FRICV.151
      FU(KMAX)=0.0                                                     FRICV.152
                                                                       FRICV.153
C-----INVERT MATRIX WITH SUBROUTINE                                    FRICV.154
      CALL TRIDG(1,FC,FT,FU,FV,ET,2,KMAX)                              FRICV.155
      DO 510 K=2, KMAX                                                 FRICV.156
      IF(SAVE4(K).NE.0.0) SAVE3(K)=SQRT(SAVE3(K)/SAVE4(K))             FRICV.157
      SAVE4(K) = QA(IS,JS,K)                                           FRICV.158
      QSAVE = QA(IS,JS,K)                                              FRICV.159
      QA(IS,JS,K) = AMAX1(ET(K), 0.0)                                  FRICV.160
      QB(IS,JS,K)=QSAVE+0.5*ALPH*(QB(IS,JS,K)-2.0*QSAVE+QA(IS,JS,K))   FRICV.161
  510 CONTINUE                                                         FRICV.162
                                                                       FRICV.163
      DO 405 K=1,KMAXP                                                 FRICV.164
      IF(DIFM(K).LT.DIF0) DIFM(K)=DIF0                                 FRICV.165
      IF(DIFH(K).LT.DIF1) DIFH(K)=DIF1                                 FRICV.166
  405 CONTINUE                                                         FRICV.167
                                                                       FRICV.168
C-----SPATIAL SMOOTHING OF DIFFUSIVITIES                               FRICV.169
      TEMP1=DIFM(1)                                                    FRICV.170
      TEMP3=DIFH(1)                                                    FRICV.171
      DO 270 K=2,KMAX                                                  FRICV.172
      TEMP2=DIFM(K)                                                    FRICV.173
      TEMP4=DIFH(K)                                                    FRICV.174
      DIFM(K)=0.5*DIFM(K)+0.25*(DIFM(K+1)+TEMP1)                       FRICV.175
```

```
      DIFH(K)=0.5*DIFH(K)+0.25*(DIFH(K+1)+TEMP3)                        FRICV.176
      TEMP1=TEMP2                                                       FRICV.177
      TEMP3=TEMP4                                                       FRICV.178
  270 CONTINUE                                                          FRICV.179
                                                                        FRICV.180
      ZNAUT=ZNASAV                                                      NEW.88
C-----DEBUG PRINT                                                       FRICV.181
      IF(LOUT.LE.0) RETURN                                              FRICV.182
      IF(IJCOLM(IS,JS).LE.0) RETURN                                     FRICV.183
      WRITE(6,805)                                                      FRICV.184
      DO 801 K=1,KMAXP                                                  FRICV.185
      WRITE(6,810) K,ALT(K),DIFH(K),DIFM(K),DVELDZ(K),RI(K),SAVE1(K),   FRICV.186
     .             SAVE2(K),SAVE3(K),QA(IS,JS,K),SAVE5(K)               FRICV.187
  801 CONTINUE                                                          FRICV.188
  800 FORMAT(22H0FRICV OUTPUT, COLUMN=,I2,1H,,I2)                       FRICV.189
  805 FORMAT (5H    K,9X,3HALT,8X,4HDIFH,8X,4HDIFM,6X,6HDVELDZ,10X,     FRICV.190
     .        2HRI,8X,4HWSOD,6X,6HDFUNCT,8X,4HQSOD,7X,5HYCOEF)          FRICV.191
  810 FORMAT(I5,1P10E12.5)                                              FRICV.192
                                                                        FRICV.193
      RETURN                                                            FRICV.194
      END                                                               FRICV.195
```

```
      FUNCTION FX(X,Y)                                          FX.2
      COMMON/IGSZZZ/Z(200)                                      COMMPL.2
      COMMON/TRANSF/ITRX,ITRY,TRN(26,22)                        COMMPL.3
      IF(ITRX.GT.0) GO TO 100                                   FX.4
      FX=X                                                      FX.5
      RETURN                                                    FX.6
  100 CONTINUE                                                  FX.7
      RETURN                                                    FX.8
      END                                                       FX.9
```

FUNCTION FY 76/76 OPT=2 FTN 4.8+508/045 30 JUN 80 14.44.44 PAGE 1

```
 1            FUNCTION FY(X,Y)                                    FY.2
              COMMON/IGSZZZ/Z(200)                                COMMPL.2
              COMMON/TRANSF/ITRX,ITRY,TRN(26,22)                  COMMPL.3
              IF(ITRY.GT.0) GO TO 100                             FY.4
 5            FY=Y                                                FY.5
              RETURN                                              FY.6
          100 CONTINUE                                            FY.7
              RETURN                                              FY.8
              END                                                 FY.9
```

81

```
      SUBROUTINE INITAL                                                 INITAL.2
C***********************************************************************INITAL.3
C*                                                                     *INITAL.4
C*        "INITAL" DEFINES MISCELLANEOUS VARIABLES SO THAT EXECUTION OF*INITAL.5
C*    LATER ROUTINES IS MORE EFFICIENT. "INITAL" ALSO DEFINES THE      *INITAL.6
C*    FOLLOWING - THE INITIAL HORIZONTAL WIND FIELD, THE BOTTOM        *INITAL.7
C*    TOPOGRAPHY, COEFFICIENTS FOR VISCOUS AND CORIOLIS TERMS, THE     *INITAL.8
C*    SIGMA LEVELS, THE INITIAL PROFILES OF MOISTURE AND TEMPERATURE,  *INITAL.9
C*    AND THE INITIAL SURFACE MOISTURE CONFIGURATION.                  *INITAL.10
C*                                                                     *INITAL.11
C***********************************************************************INITAL.12
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .             TTEN(26,22,15)                                       COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .             PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)      COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .             GW(25,21),HRATE(25,21),TJCOLM(25,21)                 COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .             PPRES(30),RELHM(30),UADAT(8),TRDAT(8)                BATELE.1
      COMMON/RADIAT/RADTN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .             RADSO(25,21,16)                                      COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .             SDOTL(21,16)                                         COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .             ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),PI(16),      COMMON.21
     .             DVELDZ(16)                                           COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .             QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,   COMMON.25
     .             SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,      COMMON.26
     .             DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,     COMMON.27
     .             SOLCON,SOLARN,IRAD,PTOLD,PTNEW,PRECIP,WW,XSAND,      COMMON.28
     .             XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,           COMMON.29
     .             HHHH,EEEE,RRRR,SSSS,NPLT,NPLTT,SUREM                 COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .             ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,         COMMON.32
     .             CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,    COMMON.33
     .             CPDIF,RPDIF,ZBL,PHOFAC,GAM,IF,IEND,IEM1,IS,KS,DX,    COMMON.34
     .             RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS,  COMMON.35
     .             GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,      COMMON.36
     .             RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,     COMMON.37
     .             KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4        COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO         COMMON.41
                                                                        COMMON.42
C-----SIZE OF COM1,COM2 ETC SHOULD BE IDENTICAL TO THOSE OF /ARRAY1/,   COMEQV.2
C-----/ARRAY2/ ETC, AND THEY SHOULD BE EQUIVALENCED.                    COMEQV.3
      DIMENSION COM1(32910),COM2(32910),COM3(32190),COM4(34320),        COMEQV.4
```

```
     .          COM5(23625),COM6(24675),COM7(4388),COM8(3675),          COMEQV.5
     .          COM9(226),COM10(33600),COM11(1344),COM12(96),           BATELE.2
     .          COM13(240),COM14(80),COM15(111),COM16(75)               COMEQV.7
     .         ,COM17(1)                                                COMEQV.8
      LEVEL 2,COM1,COM2,COM3,COM4,COM5,COM6,COM10                       COMEQV.9
      EQUIVALENCE (UA(1),COM1(1)),(UB(1),COM2(1)),(U(1),COM3(1)),       COMEQV.10
     .            (UTEN(1),COM4(1)),(PHI(1),COM5(1)),(QA(1),COM6(1)),   COMEQV.11
     .            (PSA(1),COM7(1)),(TH(1),COM8(1)),(UPRES(1),COM9(1)),  COMEQV.12
     .            (RADTN(1),COM10(1)),(FACTOR(1),COM11(1)),(FC(1),      COMEQV.13
     .            COM12(1)),(TWT(1),COM13(1)),(SAVE1(1),COM14(1)),      COMEQV.14
     .            (REFLUX(1),COM15(1)),(DT,COM16(1))                    COMEQV.15
     .          ,(IDUM,COM17(1))                                        COMEQV.16
      NAMELIST/START/GMTMHR,KMAX,DELTAT,DLAT,Q,ANGLE,                   INITAL.15
     .               HRATE,PTOP,ZBOT,SURFM,JF,DY,PINV,TNPRS,GAM1,ALPH,  NEW.89
     .               W,TG,PBOT,IRAD,XORG,XCLAY,XSAND,DAYBEG,            INITAL.17
     .               GW,DLONG,RPDIF,CPDIF,GAM,TRANS,VEG,ZNAUT,QEFFG,    INITAL.18
     .               GDEPTH,TDEEP,TWATER,IF,DX,IJCOLM,ZBL,NPRES,PPRES,  INITAL.19
     .               CX1,CX2,CX3,CX4,TJUMP,UPRES,VPRES,TPRES,CPRES,     BATELE.16
     .               THETA,RELHM                                        BATELE.17
      NAMELIST/OUTPT/ISTART,IEDIT,IPLOT,IOUT,IFILM,NCYCLS,KBEG,KBGPL    INITAL.22
                                                                        INITAL.23
                                                                        INITAL.25
C-----READ THE CONTENTS OF THE CARD INPUT                               INITAL.26
      READ (5,OUTPT)                                                    INITAL.27
      IF(ISTART.GT.0) GO TO 460                                         INITAL.28
      READ (5,START)                                                    INITAL.29
      READ (5,21) UADAT                                                 INITAL.30
      READ (5,21) TRDAT                                                 INITAL.31
                                                                        INITAL.32
C-----TIMING PARAMETERS                                                 INITAL.33
      ICYCL=0                                                           INITAL.34
      DT=DELTAT+DELTAT                                                  INITAL.35
      SIMTIM=0.0                                                        INITAL.36
      LAGHR=DLONG/15.0+0.5                                              INITAL.37
      SIMTM1=GMTMHR-LAGHR                                               INITAL.38
      IF(SIMTM1.LT.0.0) SIMTM1=24.0+SIMTM1                              INITAL.39
      SIMTM1=SIMTM1*3600.0                                              INITAL.40
      SIMTM2=GMTMHR-DLONG/15.0                                          INITAL.41
      IF(SIMTM2.LT.0.0) SIMTM2=24.0+SIMTM2                              INITAL.42
      GMDYBG=DAYBEG+(SIMTM2+DLONG/15.0)*0.0416666667                    INITAL.43
                                                                        INITAL.44
C-----CONVERT PRESSURE QUANTITIES TO DECIBARS                           INITAL.45
      PTOP=PTOP*0.10                                                    INITAL.46
      PBOT=PBOT*0.10                                                    INITAL.47
      PINV=PINV*0.10                                                    INITAL.48
      RHOFAC=1000.0                                                     INITAL.49
      DO 201 K=1,NPRES                                                  INITAL.50
  201 PPRES(K)=PPRES(K)*0.10                                            INITAL.51
                                                                        INITAL.52
C-----OPTION TO DEFINE Q ON LOG-LINEAR OR GEOMETRIC PROGRESSION SCALE   INITAL.129
  240 Q(1)=0.0                                                          INITAL.130
      IF(CX1*CX2.EQ.0.0) GO TO 242                                      INITAL.131
      RANGE=1.0+CX2*ALOG((1.0+CX1)/CX1)                                 INITAL.132
      DRANGE=RANGE/KMAX                                                 INITAL.133
      TEMP=RANGE                                                        INITAL.134
```

```
          DO 245 K=2,KMAX                                            INITAL.135
          TEMP=TEMP-DRANGE                                           INITAL.136
          Q(K)=Q(K-1)                                                INITAL.137
          NN=0                                                       INITAL.138
      243 RRANGE=1.0-Q(K)+CX2*ALOG((1.0-Q(K)+CX1)/CX1)               INITAL.139
          NN=NN+1                                                    INITAL.140
          DRDQ=-1.0-CX2/(1.0-Q(K)+CX1)                               INITAL.141
          DELQ=(TEMP-RRANGE)/DRDQ                                    INITAL.142
          Q(K)=Q(K)+DELQ                                             INITAL.143
          IF(Q(K).GE.1.0) Q(K)=1.0                                   INITAL.144
          IF(NN.GT.20) STOP "QMESH"                                  INITAL.145
          IF(ABS(DELQ).GT.0.00001) GO TO 243                         INITAL.146
      245 CONTINUE                                                   INITAL.147
          GO TO 250                                                  INITAL.148
      242 IF(CX3*CX4.EQ.0.0) GO TO 250                               INITAL.149
          DO 241 K=1,KMAX                                            INITAL.150
          KK=KMAX-K+1                                                INITAL.151
          DQ(KK)=CX3*(CX4)**(K-1)                                    INITAL.152
      241 CONTINUE                                                   INITAL.153
          Q(KMAX+1)=1.0                                              INITAL.154
          DO 244 K=1,KMAX                                            INITAL.155
          KK=KMAX-K+1                                                INITAL.156
          Q(KK)=Q(KK+1)-DQ(KK)                                       INITAL.157
      244 CONTINUE                                                   INITAL.158
      250 Q(KMAX+1)=1.0                                              INITAL.159
                                                                     INITAL.160
          KMAXP=KMAX+1                                               INITAL.161
          KE=KMAX-1                                                  INITAL.162
          KEM=KE-1                                                   INITAL.163
          IEND=IE+1                                                  INITAL.164
          JEND=JE+1                                                  INITAL.165
          IEM1=IE-1                                                  INITAL.166
          JEM1=JE-1                                                  INITAL.167
          IF(JE.EQ.1) JEND=1                                         INITAL.168
          JBEG=2                                                     INITAL.169
          IF(JE.EQ.1) JBEG=1                                         INITAL.170
                                                                     INITAL.171
C-----READ TERRAIN ELEVATION                                          NEW.92
          DO 180 J=1,JE                                              INITAL.173
          READ (5,615) (HH(I,J),I=1,IE)                              INITAL.174
      615 FORMAT(8F10.0)                                             INITAL.175
      180 CONTINUE                                                   INITAL.176
                                                                     INITAL.177
C-----GRID PARAMETERS                                                 INITAL.178
          RDX=1.0/DX                                                 INITAL.179
          RDY=1.0/DY                                                 INITAL.180
          RDX4=0.25*RDX                                              INITAL.181
          RDY4=0.25*RDY                                              INITAL.182
          RDX16=RDX/16.0                                             INITAL.183
          RDY16=RDY/16.0                                             INITAL.184
          RDXSQ=RDX*RDX                                              INITAL.185
          RDYSQ=RDY*RDY                                              INITAL.186
          RDXH=0.5*RDX                                               INITAL.187
          RDYH=0.5*RDY                                               INITAL.188
                                                                     INITAL.189
```

```
C-----FORM CORIOLIS CONSTANTS                                INITAL.190
      RLAT=DLAT*0.01745329                                   INITAL.191
      RLONG=DLONG*0.01745329                                 INITAL.192
      F=1.458E-4*SIN(RLAT)                                   INITAL.193
                                                             INITAL.194
C-----FORM SIGMA TERMS FOR LATER USE IN TRANSPORT SOLUTIONS   INITAL.195
      DO 425 K=1,KMAX                                        INITAL.196
      DQ(K)=Q(K+1)-Q(K)                                      INITAL.197
      DQI(K)=1.0/DQ(K)                                       INITAL.198
      A(K)=0.5*(Q(K+1)+Q(K))                                 INITAL.199
  425 CONTINUE                                               INITAL.200
      A(KMAXP)=0.0                                           INITAL.201
      DO 426 K=1,KE                                          INITAL.202
      AKI(K+1)=1.0/(A(K+1)-A(K))                             INITAL.203
  426 CONTINUE                                               INITAL.204
      AKI(1)=1.0/A(1)                                        INITAL.205
      AKI(KMAXP)=1.0/(1.0-A(KMAX))                           INITAL.206
                                                             INITAL.207
C-----FORM INTERPOLATIVE FUNCTION "TWT"                      INITAL.208
      TWT(1,1)=0.0                                           INITAL.209
      TWT(1,2)=0.0                                           INITAL.210
      DO 430 K=2,KMAX                                        INITAL.211
      TWT(K,1)=(Q(K)-A(K-1))/(A(K)-A(K-1))                   INITAL.212
      TWT(K,2)=1.0-TWT(K,1)                                  INITAL.213
  430 CONTINUE                                               INITAL.214
                                                             INITAL.215
      IF(GAM.LT.0.0) GO TO 394                               NEW.93
      GGR1=29.26*GAM1                                        NEW.94
      GGR2=29.26*GAM                                         NEW.95
      TPRES(NPRES)=TNPRS+273.16                              NEW.96
      IONCE=0                                                NEW.97
      DO 393 NN=2,NPRES                                      NEW.98
      N=NPRES+1-NN                                           NEW.99
      IF(PPRES(N).LT.PINV) GO TO 392                         NEW.100
      GGR=GGR2                                               NEW.101
  391 TPRES(N)=TPRES(N+1)*(PPRES(N)/PPRES(N+1))**GGR         NEW.102
      GO TO 393                                              NEW.103
  392 IF(IONCE.EQ.0) TPRES(N+1)=TPRES(N+1)+TJUMP             NEW.104
      IONCE=1                                                NEW.105
      GGR=GGR1                                               NEW.106
      GO TO 391                                              NEW.107
  393 CONTINUE                                               NEW.108
      DO 481 K=1,NPRES                                       NEW.109
  481 TPRES(K)=TPRES(K)-273.16                               BATFLE.18
  394 CONTINUE                                               NEW.112
C                                                            NEW.113
C-----INITIALIZE UPRES,VPRES,TPRES AND CPRES                 NEW.114
      XXX=ENTERP(0.0,5)                                      NEW.115
                                                             INITAL.250
      PRECIP=0.0                                             INITAL.251
      WW=7.27E-5                                             INITAL.252
      XSAND=0.57*XSAND                                       INITAL.253
      XCLAY=0.57*XCLAY                                       INITAL.254
      XORG=0.10*XORG                                         INITAL.255
      XSOIL=XSAND+XCLAY+XORG                                 INITAL.256
```

```
      DIF0=0.5                                                  INITAL.257
      DIF1=0.5                                                  INITAL.258
      TDEEP=TDEEP+273.16                                        INITAL.259
      DO 491 I=1,IE                                             INITAL.260
      DO 491 J=1,JE                                             INITAL.261
      STEMP(I,J)=TDEEP                                          INITAL.262
  491 CONTINUE                                                  INITAL.263
                                                                INITAL.264
C-----INITIALIZE SOLAR AND TERRESTRIAL RADIATION FLUX            NEW.116
      DO 492 K=1,KMAXP                                          NEW.117
       RFLUX(K)=0.0                                             NEW.118
      FLUXL2(K)=0.0                                             NEW.119
  492 CONTINUE                                                  NEW.120
C-----INITIALIZE U,V,T,C,TG,CG ETC.                              NEW.121
      CALL NESTAR                                               INITAL.266
                                                                INITAL.267
      GO TO 461                                                 INITAL.299
                                                                INITAL.300
C-----READ THE DUMP TAPE FOR RESTARTING                          INITAL.301
  460 CONTINUE                                                  INITAL.302
      JSTART=ISTART                                             INITAL.303
      JEDIT=IEDIT                                               INITAL.304
      JPLOT=IPLOT                                               INITAL.305
      JOUT=IOUT                                                 INITAL.306
      JFILM=IFILM                                               INITAL.307
      JCYCLS=NCYCLS                                             INITAL.308
      JKREG=KREG                                                INITAL.309
      JKBGPL=KBGPL                                              INITAL.310
      REWIND 7                                                  INITAL.311
      READ (7) COM1                                             INITAL.312
      READ (7) COM2                                             INITAL.313
      READ (7) COM3                                             INITAL.314
      READ (7) COM4                                             INITAL.315
      READ (7) COM5                                             INITAL.316
      READ (7) COM6                                             INITAL.317
      READ (7) COM7                                             INITAL.318
      READ (7) COM8                                             INITAL.319
      READ (7) COM9                                             INITAL.320
      READ (7) COM10                                            INITAL.321
      READ (7) COM11                                            INITAL.322
      READ (7) COM12                                            INITAL.323
      READ (7) COM13                                            INITAL.324
      READ (7) COM14                                            INITAL.325
      READ (7) COM15                                            INITAL.326
      READ (7) COM16                                            INITAL.327
      READ (7) COM17                                            INITAL.328
      ISTART=JSTART                                             INITAL.329
      IEDIT=JEDIT                                               INITAL.330
      IPLOT=JPLOT                                               INITAL.331
      IOUT=JOUT                                                 INITAL.332
      IFILM=JFILM                                               INITAL.333
      NCYCLS=JCYCLS                                             INITAL.334
      KREG=JKREG                                                INITAL.335
      KBGPL=JKBGPL                                              INITAL.336
      REWIND 7                                                  INITAL.337
```

```
C-----PRINT OF INITIAL STATE OF SYSTEM                                INITAL.338
  461 CONTINUE                                                        INITAL.339
      WRITE (6,1)                                                     INITAL.340
      WRITE (6,2)                                                     INITAL.341
      WRITE (6,3)                                                     INITAL.342
      WRITE (6,4)                                                     INITAL.343
      WRITE (6,5) GMTMHR,DLAT,PTOP,IRAD,IEDIT,DAYBEG,DLONG,PROT,ZBL,  INITAL.344
     .            IPLOT                                               INITAL.345
      WRITE (6,6) DELTAT,ANGLE,PINV,INPRS,ISTART,NCYCLS,XORG,RPDIF,XSANDINITAL.346
     .            ,ALPH                                               INITAL.347
      WRITE (6,7) NPRES,DX,CPDIF,TRANS,IFILM,W,IF,KMAX,VEG,XCLAY      INITAL.348
      WRITE (6,1005) GAM,ZNAUT,QEFFG,GDEPTH,ZBOT,TDEEP,TWATER,JE,DY,GAMIINITAL.349
      WRITE (6,900) UADAT,GMTMHR                                      BATELE.19
      WRITE (6,9)                                                     INITAL.351
      WRITE (6,10)                                                    INITAL.352
      WRITE (6,9)                                                     INITAL.353
      DO 20 K=1,NPRES                                                 INITAL.354
      WRITE (6,11) K,PPRES(K),UPRES(K),VPRES(K),TPRES(K),CPRES(K),    BATELE.20
     .RELHM(K),THETA(K)                                               BATELE.21
   20 CONTINUE                                                        INITAL.357
      WRITE (6,9)                                                     NEW.126
  721 CONTINUE                                                        NEW.127
      WRITE (6,21) TRDAT                                              INITAL.358
      WRITE (6,14) (I,I=1,IE)                                         INITAL.359
      DO 45 JJ=1,JE                                                   INITAL.360
      J=JE+1-JJ                                                       INITAL.361
      WRITE (6,46) J,(HM(I,J),I=1,IE)                                 INITAL.362
   45 CONTINUE                                                        INITAL.363
      WRITE (6,13)                                                    INITAL.364
      WRITE (6,14) (I,I=1,IEND)                                       INITAL.365
      DO 30 JJ=1,JEND                                                 INITAL.366
      J=JEND+1-JJ                                                     INITAL.367
      WRITE (6,40) J,(PCUT(I,J),I=1,IEND)                             INITAL.368
   30 CONTINUE                                                        INITAL.369
      WRITE (6,41)                                                    INITAL.370
      WRITE (6,14) (I,I=1,IE)                                         INITAL.371
      DO 31 JJ=1,JE                                                   INITAL.372
      J=JE+1-JJ                                                       INITAL.373
      WRITE (6,42) J,(IJCOLM(I,J),I=1,IE)                             INITAL.374
   31 CONTINUE                                                        INITAL.375
      WRITE (6,43)                                                    INITAL.376
      WRITE (6,14) (I,I=1,IE)                                         INITAL.377
      DO 32 JJ=1,JE                                                   INITAL.378
      J=JE+1-JJ                                                       INITAL.379
      WRITE (6,44) J,(HRATE(I,J),I=1,IE)                              INITAL.380
   32 CONTINUE                                                        INITAL.381
      WRITE (6,15) (Q(K),K=1,KMAXP)                                   INITAL.382
      WRITE (6,17) (A(K),K=1,KMAX)                                    INITAL.383
                                                                      INITAL.384
C-----I/O FORMATS                                                     INITAL.385
    1 FORMAT (1H1)                                                    INITAL.386
    2 FORMAT (49X, 33HXXXX  X  XXXX  X   X  XXXX XXXXX /              INITAL.387
     1        49X, 33HX     X  X     XX XX  X       X  /              INITAL.388
     2        49X, 33HXXXX  X  X XX  X X X  XXXX    X  /              INITAL.389
     3        49X, 33H   X  X  X  X  X   X  X       X  /              INITAL.390
```

```
      4      49X, 33HXXXX  X  XXXX  X    X  XXXX    X    //   )               INITAL.391
    3  FORMAT (47X,35HI N I T I A L   C O N D I T I O N S //)                 INITAL.392
    4  FORMAT ( 60X,9HCONSTANTS /)                                            INITAL.393
    5  FORMAT (5X,8HGMTMHR= ,1PE12.4,4X,8HDLAT=    ,1PE12.4,4X,8HPTOP=     ,INITAL.394
      1         1PE12.4,4X,8HIRAD=    ,I12,4X,8HIEDIT=   ,I12 /               INITAL.395
      3      5X,8HDAYBEG= ,1PE12.4,4X,8HDLONG=   ,1PE12.4,4X,8HPROT=     ,INITAL.396
      4         1PE12.4,4X,8HZBL=     ,1PE12.4,4X,8HIPLOT=   ,I12  )        INITAL.397
    6  FORMAT (5X,8HDELTAT= ,1PE12.4,4X,8HANGLE=   ,1PE12.4,4X,8HPINV=     ,INITAL.398
      1         1PE12.4,4X,8HTNPRS=   ,1PE12.4,4X,8HISTART= ,I12 /            INITAL.399
      2      5X,8HNCYCLS= ,I12,4X,8HXORG=    ,1PE12.4,4X,8HRPDIF=   ,        INITAL.400
      3         1PE12.4,4X,8HXSAND=   ,1PE12.4,4X,8HALPH=    ,1PE12.4)       INITAL.401
    7  FORMAT (5X,8HNPRES=  ,I12,4X,8HDX=      ,1PE12.4,4X,8HCPDIF=   ,      INITAL.402
      1         1PE12.4,4X,8HTRANS=   ,1PE12.4,4X,8HIFILM=   ,I12 /           INITAL.403
      2      5X,8HW=      ,1PE12.4,4X,8HIF=      ,I12,4X,8HKMAX=    ,         INITAL.404
      3      I12,4X,8HVEG=     ,1PE12.4,4X,8HXCLAY=   ,1PE12.4)               INITAL.405
    9  FORMAT (1X,128(1H-))                                                   INITAL.406
 1005  FORMAT (5X,8HGAM=     ,1PE12.4,4X,8HZNAUT=   ,1PE12.4,4X,              INITAL.407
      1      8HOEFFG=  ,1PE12.4,4X,8HGDEPTH= ,1PE12.4,4X,8HZBOT=    ,         INITAL.408
      2     1PE12.4/5X,8HTDEEP=   ,1PE12.4,4X,8HTWATER= ,1PE12.4,4X,           INITAL.409
      3     8HJE=     ,I12,4X,8HDY=      ,1PE12.4,4X,8HGAM1=    ,              INITAL.410
      4     1PE12.4//)                                                        INITAL.411
   10  FORMAT (7X,1HK,9X,5HPPRES,15X,5HUPRES,13X,5HVPRES,13X,5HTPRES,13X,INITAL.412
      1      5HCPRES,5X,4HR.H.,8X,5HTHETA )                                   INITAL.413
   11  FORMAT (6X,I3,5X,1PE12.4,4(6X,1PE12.4),2X,0PF5.1,5X,1PE12.4)           INITAL.414
   13 FORMAT(/18H PCUT BASED ON ZBL)                                          INITAL.415
   14 FORMAT(5H I=   ,20I6/(5X,20I6))                                         INITAL.416
   40 FORMAT(3H J=,I2,20F6.1/(5X,20F6.1))                                     INITAL.417
   41 FORMAT(/13H IJCOLM ARRAY)                                               INITAL.418
   42 FORMAT(3H J=,I2,20I6/(5X,20I6))                                         INITAL.419
   43 FORMAT(/33H HEATING RATE FROM SURFACE SOURCE)                           INITAL.420
   44 FORMAT(3H J=,I2,20F6.1/(5X,20F6.1))                                     INITAL.421
   46 FORMAT(3H J=,I2,20F6.0/(5X,20F6.0))                                     INITAL.422
   15  FORMAT (/5X,5HQ(K)=,10F10.6/(10X,10F10.6))                             INITAL.423
   17  FORMAT(5X,5HA(K)=,10F10.6/(10X,10F10.6))                               INITAL.424
   21  FORMAT (8A10)                                                          INITAL.425
  900 FORMAT(8A10/5X,57HSOUNDING TIME RELATIVE TO INITAL SIMULATION TIMENEW.128
     . IN SEC =,F12.2)                                                        NEW.129
                                                                              INITAL.426
       RETURN                                                                 INITAL.427
       END                                                                    INITAL.428
```

```
      SUBROUTINE LOCTIM                                                 LOCTIM.2
                                                                        LOCTIM.3
C-----THIS ROUTINE CALCULATES THE LOCAL TIME AND THE JULLIAN DAY        LOCTIM.4
C     CORRESPONDING TO THE PRESENT SIMULATION TIME                      LOCTIM.5
                                                                        LOCTIM.6
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJGOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATELE.1
      COMMON/RADIAT/RADIN(25,21,16),RADIO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/EC(16),ET(16),EU(16),EV(16),FT(16),FU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),RI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,ELEF,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IF,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADIN,RADIO,RADSN,RADSO         COMMON.41
                                                                        COMMON.42
      DAYNOW=DAYBEG+SIMTIM*1.157407407E-5                               LOCTIM.8
      SUNTIM=SIMTM1+SIMTIM                                              LOCTIM.9
      DAYNOW=DAYBEG+SUNTIM*1.1574074074E-5                              LOCTIM.10
      SUNSEC=AMOD(SUNTIM,60.0) +1.E-8                                   LOCTIM.11
      SUNMIN=AMOD(SUNTIM,3600.0)/60.0 +1.E-8                            LOCTIM.12
      SUNHR=AMOD(SUNTIM,86400.0)/3600.0 +1.E-8                          LOCTIM.13
      ISUNH=SUNHR                                                       LOCTIM.14
      ISUNM=SUNMIN                                                      LOCTIM.15
      ISUNS=SUNSEC                                                      LOCTIM.16
```

```
      NOWDAY=DAYNOW                                                     LOCTIM.17
                                                                        LOCTIM.18
      RETURN                                                            LOCTIM.19
      END                                                               LOCTIM.20
```

90

```
      SUBROUTINE NESTAR                                                 NESTAR.2
                                                                        NESTAR.3
C-----THIS SUBROUTINE WHEN CALLED BY INITAL PROVIDES INITIAL VALUES FOR NESTAR.4
C     THE PRIMARY VARIABLES OBTAINED THRU INTERPOLATION OF INPUT        NESTAR.5
C     AMBIENT DATA (UPRES,VPRES ETC.). THE LATTER PART IS DONE BY       NESTAR.6
C     CALLING ENTERP                                                    NESTAR.7
                                                                        NESTAR.8
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PGUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATELE.1
      COMMON/RADIAT/RADTN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/EC(16),ET(16),EU(16),EV(16),FT(16),FU(16)           COMMON.19
      COMMON/MISCLN/IWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),RI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),OFFF(16),            COMMON.24
     .              OFFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,ELEE,RRRR,SSSS,NPLT,NPLTT,SURFM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              GPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .      PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO           COMMON.41
                                                                        COMMON.42
      AMOIST(X)=0.38*EXP(17.27*X/(X+237.3))                             NEW.130
                                                                        NESTAR.10
C-----CALCULATE PI FACTOR                                               NESTAR.11
      DO 150 J=1,JE                                                     NESTAR.12
      DO 150 I=1,IE                                                     NESTAR.13
      IS=I                                                              NESTAR.14
      JS=J                                                              NESTAR.15
```

```
      KS=KMAX                                                          NESTAR.16
      ZEE=ZBOT                                                         NESTAR.17
      PSIG = PBOT                                                      NESTAR.18
      TOLL = ENTERP(PSIG,4)                                            NESTAR.19
      PDELT = 0.025 * ( PBOT-PTOP )                                    NESTAR.20
    1 PSIG = PSIG - PDELT                                              NESTAR.21
      IF ( PSIG.LE.PTOP )     STOP "NEST1"                             NESTAR.22
      TNEW = ENTERP(PSIG,4)                                            NESTAR.23
      B1=(TOLL-TNEW)/PDELT                                             NESTAR.24
      AA=TOLL-B1*(PSIG+PDELT)                                          NESTAR.25
      DZEE=29.26*(AA*ALOG((PSIG+PDELT)/PSIG)+B1*PDELT)                 NESTAR.26
      TOLL = TNEW                                                      NESTAR.27
      ZEE = ZEE + DZEE                                                 NESTAR.28
      IF ( HM(I,J).GT.ZEE )  GO TO 1                                   NESTAR.29
      ZEE = ZEE - DZEE                                                 NESTAR.30
      DP = (HM(I,J)-ZEE) / DZEE * PDELT                                NESTAR.31
    2 PRESS=(PSIG+PDELT)*EXP((B1*DP-(HM(I,J)-ZEE)/29.26)/AA)           NESTAR.32
      DPNEW = PSIG + PDELT - PRESS                                     NESTAR.33
      IF(ABS(DPNEW-DP).LE.1.E-5) GO TO 10                              NEW.131
      DP = DPNEW                                                       NESTAR.35
      GO TO 2                                                          NESTAR.36
   10 PSA(I,J) = PRESS - PTOP                                          NESTAR.37
      PSB(I,J)=PSA(I,J)                                                NESTAR.38
  150 CONTINUE                                                         NESTAR.39
                                                                       NESTAR.40
C-----INITIALIZE T AND C FIELDS                                        NEW.132
      DO 210 J=1,JE                                                    NESTAR.42
      DO 210 I=1,IE                                                    NESTAR.43
      DO 210 K=1,KMAX                                                  NESTAR.44
      IS=I                                                             NESTAR.45
      JS=J                                                             NESTAR.46
      KS=K                                                             NESTAR.47
      PSIG=A(K)*PSA(I,J)+PTOP                                          NESTAR.48
      C(I,J,K)=ENTERP(PSIG,3)                                          NESTAR.49
      CA(I,J,K)=C(I,J,K)*PSA(I,J)                                      NESTAR.50
      CB(I,J,K)=CA(I,J,K)                                              NEW.133
      T(I,J,K)=ENTERP(PSIG,4)                                          NESTAR.51
      TA(I,J,K)=T(I,J,K)*PSA(I,J)*(PBOT/PSIG)**0.286                   NESTAR.52
      TB(I,J,K)=TA(I,J,K)                                              NEW.134
  210 CONTINUE                                                         NESTAR.53
                                                                       NESTAR.54
C----DETERMINE PCUT BASED ON ZBL ABOVE THE TERRAIN SURFACE             NESTAR.55
C-----BELOW ZBL BOUNDARY-LAYER PROFILE IS USED, ABOVE ZBL INTERPOLATED NESTAR.56
C     VALUES ARE USED FROM THE TABLE FOR WIND PROFILE                  NESTAR.57
      DO 103 I=1,IEND                                                  NESTAR.58
      II=I                                                             NESTAR.59
      IM=I-1                                                           NESTAR.60
      IF(I.EQ.1) IM=1                                                  NESTAR.61
      IF(I.EQ.IEND) II=IE                                              NESTAR.62
      DO 103 J=1,JEND                                                  NESTAR.63
      JJ=J                                                             NESTAR.64
      JM=J-1                                                           NESTAR.65
      IF(J.EQ.1) JM=1                                                  NESTAR.66
      IF(J.EQ.JEND) JJ=JE                                              NESTAR.67
      PSOM=0.25*(PSA(II,JJ)+PSA(IM,JJ)+PSA(II,JM)+PSA(IM,JM))          NESTAR.68
```

```
      PUA(I,J)=PSOM                                             NESTAR.69
      PUB(I,J)=PSOM                                             NESTAR.70
      ALT1=0.0                                                  NESTAR.72
      DO 101 KK=1,KMAX                                          NESTAR.73
      K=KMAXP-KK                                                NESTAR.74
      P1=PSOM*Q(K+1)+PTOP                                       NEW.135
      P2=PSOM*Q(K  )+PTOP                                       NEW.136
      CSUM=0.25*(C(II,JJ,K)+C(IM,JJ,K)+C(II,JM,K)+C(IM,JM,K))   NESTAR.75
      TSUM=0.25*(T(II,JJ,K)+T(IM,JJ,K)+T(II,JM,K)+T(IM,JM,K))   NESTAR.76
      ALT2=ALT1+29.26*(1.0+RPDIF*CSUM)*TSUM*ALOG(P1/P2)         NEW.137
      IF(ALT2.GE.ZBL) GO TO 102                                 NESTAR.78
      ALT1=ALT2                                                 NESTAR.79
  101 CONTINUE                                                  NESTAR.80
      PCUT(I,J)=AMAX1(PTOP,PPRES(1))                            NESTAR.81
      GO TO 103                                                 NESTAR.82
  102 PCUT(I,J)=PSOM*Q(K+1)+PTOP +PSOM*DQ(K)*(ZBL-ALT1)/(ALT2-ALT1) NESTAR.83
      PCUT(I,J)=AMIN1(PCUT(I,J),PUA(I,J)+PTOP)                  NESTAR.84
  103 CONTINUE                                                  NESTAR.85
                                                                NESTAR.86
C-----INITIAL VELOCITY FIELD                                     NEW.138
      DO 200 J=1,JEND                                           NEW.139
      JS=J                                                      NESTAR.89
      DO 200 I=1,IEND                                           NEW.140
      IS=I                                                      NESTAR.91
      PS=PUA(I,J)                                               NESTAR.92
      DO 200 K=1,KMAX                                           NESTAR.93
      KS=K                                                      NESTAR.94
      PSIG = A(K)*PS+PTOP                                       NESTAR.95
      U(I,J,K)=ENTERP(PSIG,1)                                   NESTAR.96
      V(I,J,K)=ENTERP(PSIG,2)                                   NESTAR.97
      UA(I,J,K)=U(I,J,K)*PS                                     NESTAR.98
      VA(I,J,K)=V(I,J,K)*PS                                     NESTAR.99
      UB(I,J,K)=UA(I,J,K)                                       NEW.141
      VB(I,J,K)=VA(I,J,K)                                       NEW.142
  200 CONTINUE                                                  NESTAR.100
                                                                NESTAR.101
C-----BOUNDARY VALUES                                           NESTAR.102
      CALL UVBOUN                                               NESTAR.103
      DO 202 J=1,JEND                                           NESTAR.104
      PS1=PUA(1,J)                                              NESTAR.105
      PS2=PUA(IEND,J)                                           NESTAR.106
      DO 202 K=1,KMAX                                           NESTAR.107
      U(1,J,K)=UA(1,J,K)/PS1                                    NESTAR.108
      V(1,J,K)=VA(1,J,K)/PS1                                    NESTAR.109
      U(IEND,J,K)=UA(IEND,J,K)/PS2                              NESTAR.110
      V(IEND,J,K)=VA(IEND,J,K)/PS2                              NESTAR.111
  202 CONTINUE                                                  NESTAR.112
      DO 203 I=2,IE                                             NESTAR.113
      PS1=PUA(I,1)                                              NESTAR.114
      PS2=PUA(I,JEND)                                           NESTAR.115
      DO 203 K=1,KMAX                                           NESTAR.116
      U(I,1,K)=UA(I,1,K)/PS1                                    NESTAR.117
      V(I,1,K)=VA(I,1,K)/PS1                                    NESTAR.118
      U(I,JEND,K)=UA(I,JEND,K)/PS2                              NESTAR.119
      V(I,JEND,K)=VA(I,JEND,K)/PS2                              NESTAR.120
```

```
  203 CONTINUE                                                    NESTAR.121
                                                                  NESTAR.122
C-----SURFACE TEMPERATURE                                         NEW.143
      TWATER=TWATER+273.16                                        NEW.144
      IF (GDEPTH.LT.0.0) GO TO 421                                NEW.145
      DO 420 I=1,IE                                               NEW.146
      DO 420 J=1,JE                                               NEW.147
  420 TG(I,J)=TG(I,J)+273.16                                      NEW.148
      GO TO 424                                                   NEW.149
  421 GDEPTH=-GDEPTH                                              NEW.150
      DO 423 I=1,IE                                               NEW.151
      DO 423 J=1,JE                                               NEW.152
      IS=I                                                        NEW.153
      JS=J                                                        NEW.154
      KS=KMAX                                                     NEW.155
      TG(I,J)=TWATER                                              NEW.156
      IF (GW(I,J).LT.0.0) GO TO 422                               NEW.157
      PSIG=PSA(I,J)+PTOP                                          NEW.158
      TG(I,J)=ENTERP(PSIG,4)                                      NEW.159
  422 CONTINUE                                                    NEW.160
  423 CONTINUE                                                    NEW.161
  424 CONTINUE                                                    NEW.162
C-----INITIAL SURFACE VAPOR PRESSURE                              NEW.163
      DO 465 J=1,JE                                               NEW.164
      DO 465 I=1,IE                                               NEW.165
      TGG=TG(I,J)-273.16                                          NEW.166
      FTG=AMOIST(TGG)/(PSA(I,J)+PTOP)                             NEW.167
      IF (GW(I,J).LT.0.0) GO TO 464                               NEW.168
      CG(I,J)=FTG*GW(I,J)                                         NEW.169
      GO TO 465                                                   NEW.170
  464 CG(I,J)=FTG                                                 NEW.171
  465 CONTINUE                                                    NEW.172
      PSIG=PTOP                                                   NESTAR.137
      TTOP=ENTERP(PSIG,4)                                         NESTAR.138
      CTOP=ENTERP(PSIG,3)                                         NESTAR.139
      IS=1                                                        NESTAR.140
      JS=1                                                        NESTAR.141
      KS=1                                                        NESTAR.142
      A(1)=0.0                                                    NESTAR.143
      UTOP=ENTERP(PSIG,1)                                         NESTAR.144
      VTOP=ENTERP(PSIG,2)                                         NESTAR.145
      A(1)=0.5*(Q(1)+Q(2))                                        NESTAR.146
                                                                  NESTAR.147
C-----CALCULATE INITIAL VALUES OF TURB KE FROM LEVEL 2 APPROX     NESTAR.148
                                                                  NESTAR.149
      FACTK=0.35                                                  NESTAR.150
      DO 500 J=1,JE                                               NESTAR.151
      JS=J                                                        NESTAR.152
      JSP1=JS+1                                                   NESTAR.153
      IF(JE.EQ.1) JSP1=1                                          NESTAR.154
      DO 500 I=1,IE                                               NESTAR.155
      IS=I                                                        NESTAR.156
      ISP1=IS+1                                                   NESTAR.157
      TEMP=29.26*PSA(IS,JS)                                       NESTAR.158
      ALT(KMAXP)=0.0                                              NESTAR.159
```

```
      ZNASAV=ZNAUT                                                      NEW.173
      IF (GW(IS,JS).LT.0.0) ZNAUT=0.1*ZNASAV                            NEW.174
      DO 213 KK=1,KMAX                                                  NESTAR.160
      K=KMAXP-KK                                                        NESTAR.161
      RPRIM=1.0+PPDIF*C(IS,JS,K)                                        NESTAR.162
  213 ALT(K)=ALT(K+1)+TEMP*RPRIM*T(IS,JS,K)/(PSA(IS,JS)*A(K)+PTOP)*DQ(K)NESTAR.163
                                                                        NESTAR.164
      DO 260 K=2,KMAX                                                   NESTAR.165
      PP=PSA(IS,JS)*Q(K)+PTOP                                           NESTAR.166
      TT=TWT(K,1)*T(IS,JS,K)+TWT(K,2)*T(IS,JS,K-1)                      NESTAR.167
      CC=TWT(K,1)*C(IS,JS,K)+TWT(K,2)*C(IS,JS,K-1)                      NESTAR.168
      FACT=0.0342*PP/(TT*PSA(IS,JS))                                    NESTAR.169
      DUDZ=(U(IS,JS,K-1)+U(ISP1,JS,K-1)+U(IS,JSP1,K-1)+U(ISP1,JSP1,K-1)-NESTAR.170
     .U(IS,JS,K)-U(ISP1,JS,K)-U(IS,JSP1,K)-U(ISP1,JSP1,K))/(A(K-1)-A(K))NESTAR.171
      DVDZ=(V(IS,JS,K-1)+V(ISP1,JS,K-1)+V(IS,JSP1,K-1)+V(ISP1,JSP1,K-1)-NESTAR.172
     .V(IS,JS,K)-V(ISP1,JS,K)-V(IS,JSP1,K)-V(ISP1,JSP1,K))/(A(K-1)-A(K))NESTAR.173
      DVELDZ(K)=FACT*SQRT(DUDZ*DUDZ+DVDZ*DVDZ)*0.25                     NESTAR.174
      DVELDZ(K)=AMAX1(1.0E-20,DVELDZ(K))                                NESTAR.175
      DTDZ=0.01/(1.0+0.88*CC)+FACT*(T(IS,JS,K-1)-T(IS,JS,K)+0.61*TT     NESTAR.176
     .*(C(IS,JS,K-1)-C(IS,JS,K)))/(A(K)-A(K-1))                         NESTAR.177
      RI(K)=9.81*DTDZ/(DVELDZ(K)*DVELDZ(K)*TT)                          NESTAR.178
  260 CONTINUE                                                          NESTAR.179
C----RICHARDSON NUMBER AT THE TOP                                       NESTAR.180
      FACT=0.0342*PTOP/(TTOP*PSA(IS,JS))                                NESTAR.181
      DUDZ=(U(IS,JS,1)+U(ISP1,JS,1)+U(IS,JSP1,1)+U(ISP1,JSP1,1)-4.0*UTOPNESTAR.182
     .)/A(1)                                                            NESTAR.183
      DVDZ=(V(IS,JS,1)+V(ISP1,JS,1)+V(IS,JSP1,1)+V(ISP1,JSP1,1)-4.0*VTOPNESTAR.184
     .)/A(1)                                                            NESTAR.185
      DVELDZ(1)=FACT*SQRT(DUDZ*DUDZ+DVDZ*DVDZ)*0.25                     NESTAR.186
      DVELDZ(1)=AMAX1(1.0E-20,DVELDZ(1))                                NESTAR.187
      DTDZ=0.01/(1.0+0.88*CTOP)+FACT*(TTOP-T(IS,JS,1)+0.61*TTOP*(CTOP   NESTAR.188
     .-C(IS,JS,1)))/A(1)                                                NESTAR.189
      RI(1)=9.81*DTDZ/(DVELDZ(1)*DVELDZ(1)*TTOP)                        NESTAR.190
C----RICHARDSON NUMBER AT THE BOTTOM                                    NESTAR.191
      FACT=0.0342*(PSA(IS,JS)+PTOP)/(TG(IS,JS)*PSA(IS,JS))              NESTAR.192
      DUDZ=(U(IS,JS,KMAX)+U(ISP1,JS,KMAX)+U(IS,JSP1,KMAX)+U(ISP1,JSP1,  NESTAR.193
     .KMAX))/(A(KMAX)-1.0)                                              NESTAR.194
      DVDZ=(V(IS,JS,KMAX)+V(ISP1,JS,KMAX)+V(IS,JSP1,KMAX)+V(ISP1,JSP1,  NESTAR.195
     .KMAX))/(A(KMAX)-1.0)                                              NESTAR.196
      DVELDZ(KMAXP)=FACT*SQRT(DUDZ*DUDZ+DVDZ*DVDZ)*0.25                 NESTAR.197
      DVELDZ(KMAXP)=AMAX1(1.0E-20,DVELDZ(KMAXP))                        NESTAR.198
      DTDZ=0.01/(1.0+0.88*CG(IS,JS))+FACT*(T(IS,JS,KMAX)-TG(IS,JS)+0.61*NESTAR.199
     .TG(IS,JS)*(C(IS,JS,KMAX)-CG(IS,JS)))/(1.0-A(KMAX))                NESTAR.200
      RI(KMAXP)=9.81*DTDZ/(DVELDZ(KMAXP)*DVELDZ(KMAXP)*TG(IS,JS))       NESTAR.201
                                                                        NESTAR.202
C----CALCULATE THE MACROLENGTH                                          NESTAR.203
      DO 280 K=1,KMAX                                                   NESTAR.204
      SAVE1(K)=ALT(K)-ALT(K+1)                                          NESTAR.205
      SAVE4(K)=0.5*(ALT(K)+ALT(K+1))                                    NESTAR.206
  280 CONTINUE                                                          NESTAR.207
      DO 266 K=1,KMAXP                                                  NESTAR.208
      IF(RI(K).LT.0.2412) GO TO 265                                     NESTAR.209
      SAVE2(K)=0.0                                                      NESTAR.210
      GO TO 266                                                         NESTAR.211
  265 RCOEF=16.775*RI(K)-3.042                                          NESTAR.212
```

```
      CCOEF=(115.19*RI(K)-27.79)*RI(K)                                   NESTAR.213
      YCOEF=-BCOEF+SQRT(BCOEF*BCOEF-CCOEF)                               NESTAR.214
      SAVE2(K)=SQRT(YCOEF)*DVELDZ(K)                                     NESTAR.215
  266 CONTINUE                                                           NESTAR.216
      DO 267 K=1,KMAX                                                    NESTAR.217
      SAVE2(K)=0.5*(SAVE2(K)+SAVE2(K+1))                                 NESTAR.218
      SAVE3(K)=SAVE2(K)*(ZNAUT+SAVE4(K))*SAVE1(K)                        NESTAR.219
      SAVE2(K)=SAVE3(K)*SAVE4(K)                                         NESTAR.220
      SAVE1(K)=(SAVE4(K)+ZNAUT)*FACTK                                    NESTAR.221
  267 CONTINUE                                                           NESTAR.222
      XLO=50.0                                                           NESTAR.223
      NN=0                                                               NESTAR.224
  310 X1=0.0                                                             NESTAR.225
      X2=0.0                                                             NESTAR.226
      DO 230 K=1,KMAX                                                    NESTAR.227
      DENOM=1.0/(XLO+SAVE1(K))                                           NESTAR.228
      X1=X1+SAVE2(K)*DENOM                                               NESTAR.229
      X2=X2+SAVE3(K)*DENOM                                               NESTAR.230
  230 CONTINUE                                                           NESTAR.231
      NN=NN+1                                                            NESTAR.232
      XLN=0.0                                                            NESTAR.233
      IF(X1*X2.LT.1.E-20) GO TO 350                                      NESTAR.234
      XLN=0.1*X1/X2                                                      NESTAR.235
      IF(LOUT.GT.0) WRITE(6,66) IS,JS,NN,XLN                             NESTAR.236
   66 FORMAT(4H IS=,I3,4H JS=,I3,4H NN=,I3,5X,4HXLN=,1PE12.4)            NESTAR.237
      IF(ABS(XLN/XLO-1.0).LT.0.0001) GO TO 350                           NESTAR.238
      XLO=AMAX1(XLN,1.0E-20)                                             NESTAR.239
      IF(NN.LT.20 ) GO TO 310                                            NESTAR.240
      WRITE (6,66) IS,JS,NN,XLN                                          NESTAR.241
      STOP "XLO"                                                         NESTAR.242
  350 CONTINUE                                                           NESTAR.243
      XLN=AMAX1(1.0E-20,XLN)                                             NESTAR.244
      ZLAMI=1.0/XLN                                                      NESTAR.245
                                                                         NESTAR.246
      DO 400 K=1,KMAXP                                                   NESTAR.247
      QA(I,J,K)=0.0                                                      NESTAR.248
      QB(I,J,K)=0.0                                                      NESTAR.249
      IF(RI(K).GT.0.2412) GO TO 400                                      NESTAR.250
      FACT=FACTK*(ALT(K)+ZNAUT)                                          NESTAR.251
      ZL=FACT/(1.0+FACT*ZLAMI)                                           NESTAR.252
      BCOEF=16.775*RI(K)-3.042                                           NESTAR.253
      CCOEF=(115.19*RI(K)-27.79)*RI(K)                                   NESTAR.254
      YCOEF=-BCOEF+SQRT(BCOEF*BCOEF-CCOEF)                               NESTAR.255
      QA(I,J,K) = YCOEF*DVELDZ(K)*DVELDZ(K)*ZL*ZL                        NESTAR.256
      QB(I,J,K)=QA(I,J,K)                                                NESTAR.257
  400 CONTINUE                                                           NESTAR.258
      ZNAUT=ZNASAV                                                       NEW.175
  500 CONTINUE                                                           NESTAR.259
                                                                         NESTAR.260
      RETURN                                                             NESTAR.261
      END                                                                NESTAR.262
```

```
      SUBROUTINE OUTPUT                                                 OUTPUT.2
C*********************************************************************OUTPUT.3
C*                                                                    *OUTPUT.4
C*          "OUTPUT" HONORS REQUESTS FOR OUTPUT DURING THE SIMULATION. *OUTPUT.5
C*    OUTPUT TAKES THE FORM OF TAPE DUMPS, PRINTER EDITS, AND PRINTER *OUTPUT.6
C*    PLOTS. VARIABLES DUMPED TO TAPE MAY BE USED LATER FOR A PROBLEM *OUTPUT.7
C*    RESTART. OUTPUT IS ON CYCLES. FREQUENCY AND MODE OF OUTPUT      *OUTPUT.8
C*    IS CONTROLLED BY THE USER WITH THE VARIABLES IN NAMELIST        *OUTPUT.9
C*    "OUTPT" WHICH IS READ IN ROUTINE "INITAL".                      *OUTPUT.10
C*                                                                    *OUTPUT.11
C*********************************************************************OUTPUT.12
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TPDAT(8)               BATELE.1
      COMMON/RADIAT/RADIN(25,21,16),RADIO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),PI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4S16(16),QEFF(16),            COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,FLEF,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADIN,RADIO,RADSN,RADSO         COMMON.41
                                                                        COMMON.42
C-----SIZE OF COM1,COM2 ETC SHOULD BE IDENTICAL TO THOSE OF /ARRAY1/,   COMEQV.2
C-----/ARRAY2/ ETC, AND THEY SHOULD BE EQUIVALENCED.                    COMEQV.3
      DIMENSION COM1(32910),COM2(32910),COM3(32190),COM4(34320),        COMEQV.4
```

```
                 .         COM5(23625),COM6(24675),COM7(4388),COM8(3675),        COMEQV.5
                 .         COM9(226),COM10(33600),COM11(1344),COM12(96),         BATELE.2
                 .         COM13(240),COM14(80),COM15(111),COM16(75)             COMEQV.7
                 .        ,COM17(1)                                              COMEQV.8
                  LEVEL 2,COM1,COM2,COM3,COM4,COM5,COM6,COM10                    COMEQV.9
                  EQUIVALENCE (UA(1),COM1(1)),(UB(1),COM2(1)),(U(1),COM3(1)),    COMEQV.10
                 .            (UTEN(1),COM4(1)),(PHI(1),COM5(1)),(QA(1),COM6(1)),COMEQV.11
                 .            (PSA(1),COM7(1)),(TH(1),COM8(1)),(UPRES(1),COM9(1)), COMEQV.12
                 .            (RADTN(1),COM10(1)),(FACTOR(1),COM11(1)),(FC(1),   COMEQV.13
                 .            COM12(1)),(TWT(1),COM13(1)),(SAVE1(1),COM14(1)),   COMEQV.14
                 .            (REFLUX(1),COM15(1)),(DT,COM16(1))                 COMEQV.15
                 .           ,(IDUM,COM17(1))                                    COMEQV.16
                                                                                 OUTPUT.15
                  LOUT=0                                                         OUTPUT.16
                  LPLOT=0                                                        OUTPUT.17
                  LEDIT=0                                                        OUTPUT.18
                  LFILM=0                                                        OUTPUT.19
                  IF(LTAPE.GT.0) GO TO 391                                       OUTPUT.20
C-----OUTPUT ON CYCLES                                                            OUTPUT.21
              101 CONTINUE                                                       OUTPUT.22
                  IF(IEDIT.GT.0) LEDIT=1/(MOD(NCYCL,IEDIT)+1)                    OUTPUT.23
                  IF(IPLOT.GT.0) LPLOT=1/(MOD(NCYCL,IPLOT)+1)                    OUTPUT.24
                  IF(IOUT.GT.0) LOUT=1/(MOD(NCYCL,IOUT)+1)                       OUTPUT.25
                  IF(IFILM.GT.0) LFILM=1/(MOD(NCYCL,IFILM)+1)                    OUTPUT.26
              102 IF(LPLOT+LEDIT+LFILM.LE.0) RETURN                              OUTPUT.27
                                                                                 OUTPUT.28
C-----FORM U,V,T,C CORRESPONDING TO PRESENT SIMULATION TIME                       OUTPUT.29
                  DO 100 I=1,IEND                                                OUTPUT.30
                  II=I                                                           OUTPUT.31
                  IM=I-1                                                         OUTPUT.32
                  IF(I.EQ.1) IM=1                                                OUTPUT.33
                  IF(I.EQ.IEND) II=IE                                            OUTPUT.34
                  DO 100 J=1,JEND                                                OUTPUT.35
                  JJ=J                                                           OUTPUT.36
                  JM=J-1                                                         OUTPUT.37
                  IF(J.EQ.1) JM=1                                                OUTPUT.38
                  IF(J.EQ.JEND) JJ=JE                                            OUTPUT.39
                  PS=0.25*(PSA(II,JJ)+PSA(IM,JJ)+PSA(II,JM)+PSA(IM,JM))          OUTPUT.40
                  DO 100 K=1,KMAX                                                OUTPUT.41
                  PSIG=PSA(II,JJ)*A(K)+PTOP                                      OUTPUT.42
                  UTEN(I,J,K)=UA(I,J,K)/PS                                       OUTPUT.43
                  VTEN(I,J,K)=VA(I,J,K)/PS                                       OUTPUT.44
                  CTEN(II,JJ,K)=CA(II,JJ,K)/PSA(II,JJ)                           OUTPUT.45
                  TTEN(II,JJ,K)=TA(II,JJ,K)/PSA(II,JJ)                           OUTPUT.46
                  TTEN(II,JJ,K)=TTEN(II,JJ,K)*(PSIG/PBOT)**0.286                 OUTPUT.47
              100 CONTINUE                                                       OUTPUT.48
                  CALL LOCTIM                                                    OUTPUT.49
                                                                                 OUTPUT.50
C-----FILM PLOTS REQUESTED                                                        OUTPUT.58
              350 IF(LFILM.LE.0) GO TO 360                                       OUTPUT.59
                                                                                 OUTPUT.60
C-----VECTOR PLOTS ON HORIZONTAL PLANES (K=KBGPL TO KMAX)                          OUTPUT.61
                  CALL VELVEC(UTEN,VTEN,HM,IEND,JEND,IE,JE,KMAX,KBGPL,DX,DY,ICYC )  OUTPUT.62
C-----CONTOUR PLOTS OF T ON HORIZONTAL PLANES                                      OUTPUT.63
                  DO 901 K=KBGPL,KMAX                                            OUTPUT.64
```

```
      KS=K                                                              OUTPUT.65
      DO 910 I=1,IE                                                     OUTPUT.66
      DO 910 J=1,JE                                                     OUTPUT.67
  910 CONVAR(I,J)=TTEN(I,J,K)                                           OUTPUT.68
      CALL CONTUR(CONVAR,IEND,IE,JE,DX,DY,KS,1)                         OUTPUT.69
  901 CONTINUE                                                          OUTPUT.70
C-----CONTOUR PLOTS OF C ON HORIZONTAL PLANES                            OUTPUT.71
      DO 902 K=KBGPL,KMAX                                               OUTPUT.72
      KS=K                                                              OUTPUT.73
      DO 920 I=1,IE                                                     OUTPUT.74
      DO 920 J=1,JE                                                     OUTPUT.75
  920 CONVAR(I,J)=CTEN(I,J,K)                                           OUTPUT.76
      CALL CONTUR(CONVAR,IEND,IE,JE,DX,DY,KS,2)                         OUTPUT.77
  902 CONTINUE                                                          OUTPUT.78
C-----CONTOUR PLOTS OF Q ON HORIZONTAL PLANES                            OUTPUT.79
      DO 903 K=KBGPL,KMAX                                               OUTPUT.80
      KS=K                                                              OUTPUT.81
      DO 930 I=1,IE                                                     OUTPUT.82
      DO 930 J=1,JE                                                     OUTPUT.83
  930 CONVAR(I,J)=QA(I,J,K)                                             OUTPUT.84
      CALL CONTUR(CONVAR,IEND,IE,JE,DX,DY,KS,3)                         OUTPUT.85
  903 CONTINUE                                                          OUTPUT.86
C-----CONTOUR PLOT OF PSA                                                OUTPUT.87
      KS=0                                                              OUTPUT.88
      CALL CONTUR(PSA,IE,IE,JE,DX,DY,KS,4)                              OUTPUT.89
C-----CONTOUR PLOT OF TERRAIN                                            OUTPUT.90
      CALL CONTUR(HM,IE,IE,JE,DX,DY,KS,5)                               OUTPUT.91
                                                                        OUTPUT.92
C-----PRINTER EDIT OF FIELD ARRAYS                                       OUTPUT.93
  360 IF(LEDIT.LE.0) GO TO 370                                          OUTPUT.94
      WRITE(6,800) ICYCL,SIMTIM,NOWDAY,ISUNH,ISUNM,ISUNS                OUTPUT.95
                                                                        OUTPUT.96
C-----U VELOCITY                                                         OUTPUT.97
      WRITE (6,554)                                                     OUTPUT.98
      DO 385 K=KBEG,KMAX                                                OUTPUT.99
      WRITE (6,555) ICYCL,K                                             OUTPUT.100
      WRITE (6,565) (I,I=1,IEND)                                        OUTPUT.101
      DO 385 JJ=1,JEND                                                  OUTPUT.102
      J=JEND+1-JJ                                                       OUTPUT.103
      WRITE (6,561) J,(UTEN(I,J,K),I=1,IEND)                            OUTPUT.104
  385 CONTINUE                                                          OUTPUT.105
                                                                        OUTPUT.106
C-----V VELOCITY                                                         OUTPUT.107
      WRITE (6,554)                                                     OUTPUT.108
      DO 390 K=KBEG,KMAX                                                OUTPUT.109
      WRITE (6,556) ICYCL,K                                             OUTPUT.110
      WRITE (6,565) (I,I=1,IEND)                                        OUTPUT.111
      DO 390 JJ=1,JEND                                                  OUTPUT.112
      J=JEND+1-JJ                                                       OUTPUT.113
      WRITE (6,561) J,(VTEN(I,J,K),I=1,IEND)                            OUTPUT.114
  390 CONTINUE                                                          OUTPUT.115
                                                                        OUTPUT.116
C-----TEMPERATURE                                                        OUTPUT.117
      WRITE (6,554)                                                     OUTPUT.118
      DO 395 K=KBEG,KMAX                                                OUTPUT.119
```

```
      WRITE (6,557) ICYCL,K
      WRITE (6,573) (I,I=1,IF)
      DO 395 JJ=1,JE
      J=JE+1-JJ
      WRITE (6,562) J,(TTEN(I,J,K),I=1,IF)
  395 CONTINUE
      WRITE (6,568) ICYCL
      WRITE (6,573) (I,I=1,IF)
      DO 396 JJ=1,JE
      J=JE+1-JJ
      WRITE (6,562) J,(TG(I,J),I=1,IF)
  396 CONTINUE

C-----MOISTURE
      WRITE (6,554)
      DO 400 K=KBEG,KMAX
      WRITE (6,558) ICYCL,K
      WRITE (6,573) (I,I=1,IF)
      DO 400 JJ=1,JE
      J=JE+1-JJ
      WRITE (6,563) J,(CTEN(I,J,K),I=1,IF)
  400 CONTINUE
      WRITE (6,569) ICYCL
      WRITE (6,573) (I,I=1,IF)
      DO 401 JJ=1,JE
      J=JE+1-JJ
      WRITE (6,563) J,(CG(I,J),I=1,IF)
  401 CONTINUE

C-----TURBULENT KINETIC ENERGY
      WRITE (6,554)
      DO 410 K=KBEG,KMAX
      WRITE (6,564) ICYCL,K
      WRITE (6,573) (I,I=1,IF)
      DO 410 JJ=1,JE
      J=JE+1-JJ
      WRITE (6,574) J,(QA(I,J,K),I=1,IF)
  410 CONTINUE

C-----COMPUTE A.G.L.
  370 IF(LEDIT+LTAPE.LE.0) GO TO 380
      DO 403 J=1,JE
      DO 403 I=1,IE
      TEM=29.26*PSA(I,J)
      K=KMAX
      RPRIM=(1.0+RPDIF*CTEN(I,J,K))
      CTEN(I,J,K)=TEM*RPRIM*TTEN(I,J,K)/(PSA(I,J)*A(K)+PTOP)*DQ(K)
      DO 403 KK=1,KE
      K=KMAX-KK
      RPRIM=(1.0+RPDIF*CTEN(I,J,K))
      CTEN(I,J,K)=CTEN(I,J,K+1)
     +TEM*RPRIM*TTEN(I,J,K)/(PSA(I,J)*A(K)+PTOP)*DQ(K)
  403 CONTINUE

      IF(LEDIT.LE.0) GO TO 380
```

```
      WRITE (6,554)                                          OUTPUT.175
      DO 404 K=KBEG,KMAX                                     OUTPUT.176
      WRITE (6,566) ICYCL,K                                  OUTPUT.177
      WRITE (6,565) (I,I=1,IE)                               OUTPUT.178
      DO 404 JJ=1,JE                                         OUTPUT.179
      J=JE+1-JJ                                              OUTPUT.180
      WRITE (6,567) J,(CTEN(I,J,K),I=1,IE)                   OUTPUT.181
  404 CONTINUE                                               OUTPUT.182
                                                             OUTPUT.183
C-----SOIL MOISTURE                                          OUTPUT.184
      WRITE (6,554)                                          OUTPUT.185
      WRITE (6,570) ICYCL                                    OUTPUT.186
      WRITE (6,565) (I,I=1,IE)                               OUTPUT.187
      DO 406 JJ=1,JE                                         OUTPUT.188
      J=JE+1-JJ                                              OUTPUT.189
      WRITE (6,572) J,(GW(I,J),I=1,IE)                       OUTPUT.190
  406 CONTINUE                                               OUTPUT.191
                                                             OUTPUT.192
C-----PSA                                                    OUTPUT.193
      WRITE (6,554)                                          OUTPUT.194
      WRITE (6,580) ICYCL                                    OUTPUT.195
      WRITE (6,573) (I,I=1,IE)                               OUTPUT.196
      DO 500 JJ=1,JE                                         OUTPUT.197
      J=JE+1-JJ                                              OUTPUT.198
      WRITE (6,581) J,(PSA(I,J),I=1,IE)                      OUTPUT.199
  500 CONTINUE                                               OUTPUT.200
                                                             OUTPUT.201
      DEG=57.29577951                                        NEW.176
      PII=3.1415926536                                       NEW.177
      E10=0.0000000001                                       NEW.178
      ISTATI=0                                               NEW.179
      DO 871 I=1,IE                                          NEW.180
      DO 871 J=1,JE                                          NEW.181
      IF (IJCOLM(I,J).EQ.0) GO TO 871                        NEW.182
      ISTATI=ISTATI+1                                        NEW.183
      WRITE(6,874)                                           NEW.184
      WRITE(6,873) ISTATI,I,J,HM(I,J)                        NEW.185
      WRITE(6,875)                                           NEW.186
      WRITE(6,876)                                           NEW.187
      DO 881 K=1,KMAX                                        NEW.188
      KP=K+1                                                 NEW.189
      IF (K.EQ.KMAX) GO TO 882                               NEW.190
      ZAGLO=0.5*(CTEN(I,J,K)+CTEN(I,J,KP))                   NEW.191
      GO TO 883                                              NEW.192
  882 ZAGLO=0.5*CTEN(I,J,KMAX)                               NEW.193
  883 CONTINUE                                               NEW.194
      SPEEDO=SQRT(UTEN(I,J,K)**2+VTEN(I,J,K)**2)             NEW.195
      C1=0.5                                                 NEW.196
      IF (UTEN(I,J,K).GT.0.0)C1=1.5                          NEW.197
      DIRO=DEG*(C1*PII-ATAN(VTEN(I,J,K)/(UTEN(I,J,K)+E10)))  NEW.198
      POTE=TA(I,J,K)/PSA(I,J)                                NEW.199
      WRITE(6,877) ZAGLO,SPEEDO,DIRO,TTEN(I,J,K),POTE,QA(I,J,K) NEW.200
  881 CONTINUE                                               NEW.201
  871 CONTINUE                                               NEW.202
  873 FORMAT(*  PROFILE AT STATION  *,I3,*  I= *,I3,*  J= *,I3 NEW.203
```

```
     *             .*   LOCAL ALTITUDE = *,F6.1)                              NEW.204
 874 FORMAT(///)                                                              NEW.205
 875 FORMAT(/)                                                                NEW.206
 876 FORMAT (60H      AGL        SPEED      ANGLE       TEMP      POT. T.     NEW.207
     * TURB /)                                                                NEW.208
 877 FORMAT(1X,6F10.3)                                                        NEW.209
C-----COMPUTE W AND STORE IN TTEN (DEFINED AT SAME POINTS AS U AND V)         OUTPUT.202
 380 IF(LFILM.LE.0) RETURN                                                    OUTPUT.203
     DO 354 I=1,IEND                                                          OUTPUT.204
     DO 354 J=1,JEND                                                          OUTPUT.205
     DO 354 K=1,KMAX                                                          OUTPUT.206
     TTEN(I,J,K)=0.0                                                          OUTPUT.207
 354 CONTINUE                                                                 OUTPUT.208
     DO 351 J=JBEG,JE                                                         OUTPUT.209
     JP1=J+1                                                                  OUTPUT.210
     JM1=J-1                                                                  OUTPUT.211
     IF(J.NE.1) GO TO 353                                                     OUTPUT.212
     JP1=1                                                                    OUTPUT.213
     JM1=1                                                                    OUTPUT.214
 353 DO 351 I=2,IE                                                            OUTPUT.215
     IP1=I+1                                                                  OUTPUT.216
     IM1=I-1                                                                  OUTPUT.217
     PIAV=PUA(I,J)                                                            OUTPUT.218
     PTEN(I,J)=0.0                                                            OUTPUT.219
     DO 352 K=1,KMAX                                                          OUTPUT.220
     SCR(K)=((UA(IP1,J,K)-UA(IM1,J,K))*RDXH                                   OUTPUT.221
    *      +(VA(I,JP1,K)-VA(I,JM1,K))*RDYH)*DQ(K)                             OUTPUT.222
     PTEN(I,J)=PTEN(I,J)-SCR(K)                                               OUTPUT.223
 352 CONTINUE                                                                 OUTPUT.224
     SDOTR(J,1)=0.0                                                           OUTPUT.225
     DO 355 K=2,KMAXP                                                         OUTPUT.226
     SDOTR(J,K)=SDOTR(J,K-1)-(PTEN(I,J)*DQ(K-1)+SCR(K-1))/PIAV                OUTPUT.227
 355 CONTINUE                                                                 OUTPUT.228
     DO 351 K=1,KMAX                                                          OUTPUT.229
     KP=K+1                                                                   OUTPUT.230
     OMEG=A(K)*PIAV*((SDOTR(J,K)+SDOTR(J,KP))*0.5/A(K)-                       OUTPUT.231
    *(SDOTR(J,KP)-SDOTR(J,K))*DQI(K)-(U(IP1,J,K)-U(IM1,J,K))*RDXH-            OUTPUT.232
    *(V(I,JP1,K)-V(I,JM1,K))*RDYH)                                            OUTPUT.233
     RPRIM=1.0+RPDIF*0.25*(C(I,J,K)+C(IM1,J,K)+C(I,JM1,K)+C(IM1,JM1,K))OUTPUT.234
     RHOG=(PIAV*A(K)+PTOP)/(RPRIM*7.315*(T(I,J,K)+T(IM1,J,K)                  OUTPUT.235
    *+T(I,JM1,K)+T(IM1,JM1,K)))                                               OUTPUT.236
     TTEN(I,J,K)=-OMEG/RHOG                                                   OUTPUT.237
 351 CONTINUE                                                                 OUTPUT.238
                                                                              OUTPUT.239
C-----CONOUR PLOTS OF TERRAIN CONFORMAL HORIZONTAL WINDSPEED                  NEW.210
     DO 931 K=KBGPL,KMAX                                                      OUTPUT.241
     KS=K                                                                     OUTPUT.242
     DO 940 I=1,IEND                                                          OUTPUT.243
     DO 940 J=1,JEND                                                          OUTPUT.244
 940 CONVAR(I,J)=SQRT(UTEN(I,J,K)**2+VTEN(I,J,K)**2)                          NEW.211
     CALL CONTUR(CONVAR,IEND,IEND,JEND,DX,DY,KS,6)                            OUTPUT.246
 931 CONTINUE                                                                 OUTPUT.247
     DO 933 K=KBGPL,KMAX                                                      OUTPUT.248
     KS=K                                                                     OUTPUT.249
     DO 932 I=1,IEND                                                          OUTPUT.250
```

```
          DO 932 J=1,JEND                                                   OUTPUT.251
      932 CONVAR(I,J)=TTEN(I,J,K)                                           NEW.212
          CALL CONTUR(CONVAR,IEND,IEND,JEND,DX,DY,KS,7)                     OUTPUT.253
      933 CONTINUE                                                          OUTPUT.254
          RETURN                                                            OUTPUT.255
                                                                            OUTPUT.256
C-----WRITE DUMP TAPE TO BE USED FOR RESTART                                OUTPUT.257
      391 CONTINUE                                                          OUTPUT.258
          LTAPE=0                                                           OUTPUT.259
          REWIND 8                                                          OUTPUT.260
          WRITE (8) COM1                                                    OUTPUT.261
          WRITE (8) COM2                                                    OUTPUT.262
          WRITE (8) COM3                                                    OUTPUT.263
          WRITE (8) COM4                                                    OUTPUT.264
          WRITE (8) COM5                                                    OUTPUT.265
          WRITE (8) COM6                                                    OUTPUT.266
          WRITE (8) COM7                                                    OUTPUT.267
          WRITE (8) COM8                                                    OUTPUT.268
          WRITE (8) COM9                                                    OUTPUT.269
          WRITE (8) COM10                                                   OUTPUT.270
          WRITE (8) COM11                                                   OUTPUT.271
          WRITE (8) COM12                                                   OUTPUT.272
          WRITE (8) COM13                                                   OUTPUT.273
          WRITE (8) COM14                                                   OUTPUT.274
          WRITE (8) COM15                                                   OUTPUT.275
          WRITE (8) COM16                                                   OUTPUT.276
          WRITE (8) COM17                                                   OUTPUT.277
          WRITE (6,801)                                                     OUTPUT.278
          RETURN                                                            OUTPUT.279
                                                                            OUTPUT.280
C-----I/O FORMATS                                                           OUTPUT.281
      554 FORMAT(//)                                                        OUTPUT.282
      800 FORMAT(10H1COMPLETED,I5,35H CYCLES. ELAPSED SIMULATION TIME IS,   OUTPUT.283
         ,1PE12.4,4H SEC,5X,4HDAY=,I3,5X,5HTIME=,I3,1H.,I2,1H.,I2)          OUTPUT.284
      801 FORMAT(//41H DUMP TAPE WRITTEN FOR RESTART CAPABILITY//)          OUTPUT.285
      565 FORMAT(5H I=   ,20I6/(5X,20I6))                                   OUTPUT.286
      555 FORMAT(1X,60(1H-)/12H U VELOCITY  ,20X,11HTIME CYCLE=,I4,10H  K LEVOUTPUT.287
         .EL=,I3/1X,60(1H-))                                                OUTPUT.288
      556 FORMAT(1X,60(1H-)/12H V VELOCITY ,20X,11HTIME CYCLE=,I4,10H  K LEVOUTPUT.289
         .EL=,I3/1X,60(1H-))                                                OUTPUT.290
      557 FORMAT(1X,60(1H-)/12H TEMPERATURE,20X,11HTIME CYCLE=,I4,10H  K LEVOUTPUT.291
         .EL=,I3/1X,60(1H-))                                                OUTPUT.292
      564 FORMAT(1X,73(1H-)/25H TURBULENT KINETIC ENERGY,20X,11HTIME CYCLE=,OUTPUT.293
         .I4,10H  K LEVEL=,I3/1X,73(1H-))                                   OUTPUT.294
      561 FORMAT(3H J=,I2,20F6.2/(5X,20F6.2))                               OUTPUT.295
      562 FORMAT(3H J=,I2,15F8.3/(5X,15F8.3))                               OUTPUT.296
      563 FORMAT(3H J=,I2,15F8.6/(5X,15F8.6))                               OUTPUT.297
      567 FORMAT(3H J=,I2,20F6.0/(5X,20F6.0))                               OUTPUT.298
      574 FORMAT(3H J=,I2,15F8.4/(5X,15F8.4))                               OUTPUT.299
      568 FORMAT(1X,47(1H-)/19H GROUND TEMPERATURE,13X,11HTIME CYCLE=,I4,/  OUTPUT.300
         .1X,47(1H-))                                                       OUTPUT.301
      558 FORMAT(1X,60(1H-)/12H MOISTURE    ,20X,11HTIME CYCLE=,I4,10H  K LEVOUTPUT.302
         .EL=,I3/1X,60(1H-))                                                OUTPUT.303
      569 FORMAT(1X,47(1H-)/19H GROUND MOISTURE    ,13X,11HTIME CYCLE=,I4/  OUTPUT.304
         .1X,47(1H-))                                                       OUTPUT.305
```

```
      566 FORMAT(1X,60(1H-)/12H A.G.L.       ,20X,11HTIME CYCLE=,I4,10H  K-LEVOUTPUT.306
         .EL=,I3/1X,60(1H-))                                                OUTPUT.307
      570 FORMAT(1X,47(1H-)/19H  SP. HMD. OF SOIL ,13X,11HTIME CYCLE=,I4,/  OUTPUT.308
         ,1X,47(1H-))                                                       OUTPUT.309
      571 FORMAT(1X,49(1H-)/29H TERRAIN ELEVATION ABOVE S.L.,5X,11HTIME CYCLOUTPUT.310
         .E=,I4/1X,49(1H-))                                                 OUTPUT.311
      572 FORMAT(3H J=,I2,20F6.3/(5X,20F6.3))                               OUTPUT.312
      573 FORMAT(5H I=   ,15I8/(5X,15I8))                                   OUTPUT.313
      580 FORMAT(1X,47(1H-)/4H PSA,28X,11HTIME CYCLE=,I4/1X,47(1H-))        OUTPUT.314
      581 FORMAT(3H J=,I2,15F8.3/(5X,15F8.3))                               OUTPUT.315
                                                                            OUTPUT.316
          END                                                               OUTPUT.317
```

```
      SUBROUTINE PRESUR                                                 PRESUR.2
                                                                        PRESUR.3
C-----HERE THE PRESSURE AND GEOPOTENTIAL GRADIENTS ARE CALCULATED AND   PRESUR.4
C     ADDED TO THE TENDENCY TERMS                                       PRESUR.5
                                                                        PRESUR.6
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATFLE.1
      COMMON/RADIAT/RADTN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/EC(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),PI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),OEFF(16),            COMMON.24
     .              OEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,TRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,FLEF,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .    PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO             COMMON.41
                                                                        COMMON.42
C-----CALCULATE GEOPOTENTIAL HEIGHT                                     PRESUR.10
      DO 250 I=1,IE                                                     PRESUR.11
      DO 250 J=1,JE                                                     PRESUR.12
      RPRIM=287.0*(1.0+RPDIF*C(I,J,KMAX))                               PRESUR.13
      DUM1=PSA(I,J)+PTOP                                                PRESUR.14
      DUM2=PSA(I,J)*A(KMAX)+PTOP                                        PRESUR.15
      PHI(I,J,KMAX)=9.81*HM(I,J)+RPRIM*T(I,J,KMAX)*                     PRESUR.16
     .ALOG(DUM1/DUM2)                                                   NEW.213
      DO 250 KK=2,KMAX                                                  PRESUR.18
```

```
      K=KMAXP-KK                                                        PRESUR.19
      RPRIM=287.*(1.0+0.5*RPDIF*(C(I,J,K)+C(I,J,K+1)))                  NEW.214
      DUM1=PSA(I,J)*A(K+1)+PTOP                                         PRESUR.21
      DUM2=PSA(I,J)*A(K)+PTOP                                           PRESUR.22
      PHI(I,J,K)=PHI(I,J,K+1)+RPRIM*((T(I,J,K)*DUM1-T(I,J,K+1)*DUM2)/   NEW.215
     *           (DUM1-DUM2)*ALOG(DUM1/DUM2)+T(I,J,K+1)-T(I,J,K))       NEW.216
  250 CONTINUE                                                          PRESUR.25
      DO 300 I=2,IE                                                     PRESUR.26
      IM1=I-1                                                           PRESUR.27
      IM2=I-2                                                           PRESUR.28
      IP1=I+1                                                           PRESUR.29
      IF(I.EQ.2) IM2=1                                                  PRESUR.30
      IF(I.EQ.IE) IP1=IE                                                PRESUR.31
      DO 300 J=JBEG,JE                                                  PRESUR.32
      JM1=J-1                                                           PRESUR.33
      JM2=J-2                                                           PRESUR.34
      JP1=J+1                                                           PRESUR.35
      IF(J.EQ.1) JM1=JM2=JP1=1                                          PRESUR.36
      IF(J.EQ.2) JM2=1                                                  PRESUR.37
      IF(J.EQ.JE) JP1=JE                                                PRESUR.38
      DPIDX=(PSA(I,J)+PSA(I,JM1)-PSA(IM1,J)-PSA(IM1,JM1))*RDXH          PRESUR.39
      DPIDY=(PSA(I,J)+PSA(IM1,J)-PSA(I,JM1)-PSA(IM1,JM1))*RDYH          PRESUR.40
      DO 300 K=1,KMAX                                                   PRESUR.41
      DPHIDX=(PHI(I,J,K)+PHI(I,JM1,K)-PHI(IM1,J,K)-PHI(IM1,JM1,K))*RDXH NEW.217
      DPHIDY=(PHI(I,J,K)+PHI(IM1,J,K)-PHI(I,JM1,K)-PHI(IM1,JM1,K))*RDYH NEW.218
      CSUM=C(I,J,K)+C(IM1,J,K)+C(I,JM1,K)+C(IM1,JM1,K)                  PRESUR.46
      TSUM=T(I,J,K)+T(IM1,J,K)+T(I,JM1,K)+T(IM1,JM1,K)                  PRESUR.47
      FACT=71.75+14.53125*CSUM                                          PRESUR.48
      PSIG=PUA(I,J)*A(K)+PTOP                                           PRESUR.49
      FACT=FACT*TSUM/PSIG*A(K)                                          PRESUR.50
      UTEN(I,J,K)=UTEN(I,J,K)-PUA(I,J)*(DPHIDX+FACT*DPIDX)              NEW.219
      VTEN(I,J,K)=VTEN(I,J,K)-PUA(I,J)*(DPHIDY+FACT*DPIDY)              NEW.220
  300 CONTINUE                                                          PRESUR.58
                                                                        PRESUR.59
C-----DEBUG PRINT                                                       PRESUR.60
      IF(LOUT.LE.0) RETURN                                              PRESUR.61
      DO 500 J=1,JE                                                     PRESUR.62
      DO 500 I=1,IE                                                     PRESUR.63
      IF(IJCOLM(I,J).LE.0) GO TO 500                                    PRESUR.64
      WRITE(6,610) I,J                                                  PRESUR.65
      WRITE(6,615)                                                      PRESUR.66
      DO 450 K=1,KMAX                                                   PRESUR.67
      WRITE(6,620) K,UTEN(I,J,K),VTEN(I,J,K)                            PRESUR.68
  450 CONTINUE                                                          PRESUR.69
      WRITE(6,630) PSA(I,J)                                             PRESUR.70
  500 CONTINUE                                                          PRESUR.71
  600 FORMAT(1P5E25.15)                                                 PRESUR.72
  610 FORMAT(22H0PRESUR OUTPUT, COLUMN,I2,1H,,I2)                       PRESUR.73
  615 FORMAT(6H     K,9X,4HUTEN,9X,4HVTEN)                              PRESUR.74
  620 FORMAT(I5,1P2E13.5)                                               PRESUR.75
  630 FORMAT(5H PSA=,1PE13.5)                                           PRESUR.76
      RETURN                                                            PRESUR.77
      END                                                               PRESUR.78
```

```
      SUBROUTINE RADFLD                                                 RADFLD.2
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATELE.1
      COMMON/RADIAT/RADTN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQT(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),PI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,ELEE,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTIM,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,INPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO         COMMON.41
                                                                        COMMON.42
      DATA IONCE,IONCE1/0,0/                                            NEW.221
      IF(ISTART.GT.0) IONCE=1                                           NEW.222
C                                                                       NEW.223
C-----CHECK FOR SUN-RISE TIME                                           NEW.224
      IF(IONCE1.EQ.1) GO TO 99                                          NEW.225
      COSZEN1=COS((GMDYBG+SIMTIM*1.1574074074E-5-1.0)*.017208006)       NEW.226
      IF(COSZEN.LT.0.0.AND.COSZEN1.GE.0.0) GO TO 98                     NEW.227
      GO TO 99                                                          NEW.228
   98 IONCE1=1                                                          NEW.229
      CALL LOCTIM                                                       NEW.230
      WRITE (6,900) ICYCL,SIMTIM,NOWDAY,ISUNH,ISUNM,ISUNS               NEW.231
  900 FORMAT(/19H SUN RISE AT CYCLE=,I5,26H  ELAPSED SIMULATION TIME=,  NEW.232
     .1PE12.4,4H SEC,5X,4HDAY=,I3,5X,5HTIME=,I3,1H.,I2,1H.,I2/)         NEW.233
```

```
         GO TO 100                                                  NEW.234
      99 CONTINUE                                                   NEW.235
                                                                    RADFLD.5
C-----CHECK TO SEE IF TIME IS RIGHT FOR NEW RADIATION CALCULATION   RADFLD.6
         IF(ICYCL.LE.IRAD) GO TO 100                                RADFLD.7
         IF(MOD(ICYCL,IRAD).NE.0) RETURN                            RADFLD.8
                                                                    RADFLD.9
C*************************************                              RADFLD.10
C*    NEW RADIATIVE FLUXES           *                              RADFLD.11
C*************************************                              RADFLD.12
C-----INSOLATION                                                    RADFLD.13
     100 CALL SOLAR                                                 RADFLD.14
                                                                    RADFLD.15
         DO 500 J=1,JF                                              RADFLD.16
         DO 500 I=1,IE                                              RADFLD.17
         IS=I                                                       RADFLD.18
         JS=J                                                       RADFLD.19
                                                                    RADFLD.20
C-----PRESSURE PROFILE                                              RADFLD.21
         DO 200 K=1,KMAXP                                           RADFLD.22
         P(K)=Q(K)*PSA(I,J)+PTOP                                    RADFLD.23
     200 CONTINUE                                                   RADFLD.24
                                                                    RADFLD.25
C-----EFFECTIVE WATER VAPOR (CGS)                                   RADFLD.26
         QEFF(1)=QEFFG                                              RADFLD.27
         TEMP1=PTOP**1.6                                            RADFLD.28
         DO 300 K=2,KMAXP                                           RADFLD.29
         TEMP2=P(K)**1.6                                            RADFLD.30
         QEFF(K)=QEFF(K-1)+0.402*C(I,J,K-1)*(TEMP2-TEMP1)           RADFLD.31
         TEMP1=TEMP2                                                RADFLD.32
     300 CONTINUE                                                   RADFLD.33
                                                                    RADFLD.34
C-----TERRESTRIAL RADIATION                                         RADFLD.35
         CALL RADTER                                                RADFLD.36
                                                                    RADFLD.37
C-----SOLAR RADIATION                                               RADFLD.38
         CALL RADSOL                                                RADFLD.39
                                                                    RADFLD.40
C-----STORE IN A 3D ARRAY                                           RADFLD.41
         IF(ICYCL.LE.IRAD) GO TO 450                                RADFLD.42
     350 DO 400 K=1,KMAXP                                           RADFLD.43
         RADTO(I,J,K)=RADTN(I,J,K)                                  RADFLD.44
         RADSO(I,J,K)=RADSN(I,J,K)                                  RADFLD.45
     400 CONTINUE                                                   RADFLD.46
         IF(I+J.EQ.2) RTOLD=RTNEW                                   RADFLD.47
         IF(IONCE.EQ.0) GO TO 500                                   RADFLD.48
     450 DO 480 K=1,KMAXP                                           RADFLD.49
         RADTN(I,J,K)=FLUXL2(K)                                     RADFLD.50
         RADSN(I,J,K)=RFLUX (K)                                     RADFLD.51
     480 CONTINUE                                                   RADFLD.52
         RTNEW=ICYCL                                                RADFLD.53
         IF(IONCE.EQ.0) GO TO 350                                   RADFLD.54
     500 CONTINUE                                                   RADFLD.55
         IONCE=1                                                    RADFLD.56
         RETURN                                                     RADFLD.57
```

```
      END                                                               RADFLD.58
```

109

```
      SUBROUTINE RADFLX                                                  RADFLX.2
C*********************************************************************RADFLX.3
C*                                                                   *RADFLX.4
C*       "RADFLX" PERFORMS A LINEAR EXTRAPOLATION UPON RADIATIVE      *RADFLX.5
C*    FLUXES DEFINED AT TWO EARLIER TIMES IN ORDER TO OBATIN AN      *RADFLX.6
C*    ESTIMATE OF THE FLUX AT THE CURRENT TIME.                      *RADFLX.7
C*                                                                   *RADFLX.8
C*********************************************************************RADFLX.9
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15)  COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15)  COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)      COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),        COMMON.5
     .              TTEN(26,22,15)                                       COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)      COMMON.7
      COMMON/ARRAY6/OA(25,21,16),OB(25,21,16),HDIFH(25,21,15)            COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),         COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)      COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),          COMMON.11
     .              GW(25,21),HRATE(25,21),TJCOLM(25,21)                 COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),   COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)                BATELE.1
      COMMON/RADIAT/RADIN(25,21,16),RADIO(25,21,16),RADSN(25,21,16),     COMMON.15
     .              RADSO(25,21,16)                                      COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),            COMMON.17
     .              SDOTL(21,16)                                         COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)            COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SGP(16),DIFM(16),DIFH(16),PI(16),      COMMON.21
     .              DVELDZ(16)                                           COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)    COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),             COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,   COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,      COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,     COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,      COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,           COMMON.29
     .              HHHH,ELEE,RRRR,SSSS,NPLT,NPLTT,SUREM                 COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,    COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,         COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,    COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IF,IEND,IEM1,IS,KS,CX,    COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS,  COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,      COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,     COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4        COMMON.38
      COMMON/ADARAY/IDUM                                                 COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,   COMMON.40
     .        PHIPIX,PHIPIY,OA,OB,HDIFH,RADIN,RADIO,RADSN,RADSO          COMMON.41
                                                                         COMMON.42
                                                                         RADFLX.11
      DELDEF=RTNEW-RTOLD                                                 RADFLX.12
      DELNOW=ICYCL-RTOLD                                                 RADFLX.13
      FACT=1.0                                                           RADFLX.14
      IF(DELDEF.GT.0.0) FACT=DELNOW/DELDEF                               RADFLX.15
                                                                         RADFLX.16
```

```
C-----OBTAIN CURRENT RADIATIVE FLUXES VIA EXTRAPOLATION
      DO 200 K=1,KMAXP
      RFLUX (K)=RADSO(IS,JS,K)+FACT*(RADSN(IS,JS,K)-RADSO(IS,JS,K))
      IF(RFLUX(K).LT.0.0) RFLUX(K)=0.0
      FLUXL2(K)=RADTO(IS,JS,K)+FACT*(RADTN(IS,JS,K)-RADTO(IS,JS,K))
  200 CONTINUE

C-----DEBUG PRINT
      IF(LOUT.LE.0) RETURN
      IF(IJCOLM(IS,JS).LE.0) RETURN
      WRITE(6,700) IS,JS
      WRITE(6,710)
      DO 600 K=1,KMAXP
      WRITE (6,720) K,RFLUX(K),FLUXL2(K)
  600 CONTINUE
      WRITE(6,730) DELDFF,RINEW,RTOLD,DELNOW

C-----I/O FORMATS
  700 FORMAT(23H0RADFLX OUTPUT, COLUMN=,I3,1H,,I2)
  710 FORMAT(5H    K,5X,5HRFLUX,8X,6HFLUXL2)
  720 FORMAT(I5,1P4E13.5)
  730 FORMAT(8H DELDFF=,1PE13.5,5X,7HRINEW =,1PE13.5,5X,7HRTOLD =,
     .1PE13.5,5X,7HDELNOW=,1PE13.5)
      RETURN
      END
```

```
      SUBROUTINE RADSOL                                                 RADSOL.2
C***********************************************************************RADSOL.3
C*                                                                     *RADSOL.4
C*       "RADSOL" CALCULATES THE RADIANT FLUX WITHIN THE ATMOSPHERE    *RADSOL.5
C*    DUE TO SOLAR INSOLATION. THE PRESCRIPTION IS THAT OF KATAYAMA    *RADSOL.6
C*    AS WAS INCLUDED IN UCLA"S 3 LEVEL GCM.                           *RADSOL.7
C*                                                                     *RADSOL.8
C***********************************************************************RADSOL.9
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),TJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TPDAT(8)               BATELE.1
      COMMON/RADIAT/RADIN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),RI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(15)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),OEFF(16),            COMMON.24
     .              OEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,R,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,ELEE,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIFQ,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADIN,RADTO,RADSN,RADSO         COMMON.41
                                                                        COMMON.42
                                                                        RADSOL.11
C-----CHECK TO SEE IF SUN IS ABOVE HORIZON                              RADSOL.12
      IF(COSZEN.GT.0.0) GO TO 200                                       RADSOL.13
      DO 100 K=1,KMAXP                                                  RADSOL.14
      RFLUX(K)=0.0                                                      RADSOL.15
  100 CONTINUE                                                          RADSOL.16
```

```
      GO TO 550

C-----RAYLEIGH SCATTERING ALBEDO (NORMALIZED TO THE SCATTERED FLUX)
C-----( PRESCRIPTION OF LACIS AND HANSON J.A.S. 31, PP.118 )
  200 DUMZZ=COSZEN/(PSA(IS)+PTOP)*100.
      ALCSKY=0.430/(1.0+6.43*DUMZZ)

C-----SURFACE ALBEDO
      ALSURF=0.12+0.025*VEG/(0.02+COSZEN/TRANS)
      IF(HHH.LT.12.0) GO TO 210
      ALSURF=ALSURF+((0.05+0.02*VEG)/(0.02+COSZEN/TRANS**0.25)-ALSURF)
     .*(1.0-COSZEN/(COEFF1+COEFF2))
  210 ALSURF=AMIN1(1.0,ALSURF)

C-----FORM SCATTERED FLUX
C-----OZONE ABSORPTION (NORMALIZED TO SCATTERED FLUX)
C-----OZONE VERTICAL COLUMN ASSIGNED VALUE OF 0.4
      SECZ=35.0/SQRT(1.0+1224.*COSZEN*COSZEN)
      OZON=0.4
      OZ=OZON*SECZ
      AOZ=(0.0325/(1.0+(0.042+0.000323*OZ)*OZ)
     *+1.661/(1.0+139.6*OZ)**0.805+0.1011/(1.0+(103.6*OZ)**3.0))*OZ
      FLUXS=SS*(1.0-AOZ-ALCSKY)/(1.0-0.0685*ALSURF)*(1.0-ALSURF)
      FAC2=FACTOM*RDX
      TEMP1=1.0
      SAA=SA*TEMP1

C-----CALCULATE THE FLUX
      TEPP=QEFF(KMAXP)*SECZ
      DO 300 K=1,KMAXP
      TEMP=QEFF(K)*SECZ
      TEMP1=TEPP+1.667*(QEFF(KMAXP)-QEFF(K))
      RFLUX(K)=SAA*(AMAX1(0.0,1.0-0.279*TEMP**0.303)-ALSURF*AMAX1(
     .0.0,1.0-0.279*TEMP1**0.303))+FLUXS
  300 CONTINUE

C-----DEBUG PRINT
  550 IF(LOUT.LE.0) RETURN
      IF(IJCOLM(IS,JS).LE.0) RETURN
      WRITE(6,700) IS,JS
      WRITE(6,720)
      DO 600 K=1,KMAXP
      WRITE(6,710) K,QEFF(K),RFLUX(K)
  600 CONTINUE
      WRITE(6,730) SA,FAC2,SAA,ALCSKY,ALSURF,COSZEN,HHH,SECZ,VEG,
     .TRANS,COEFF1,COEFF2,SS,PSA(IS,JS),FLUXS

C-----I/O FORMATS
  700 FORMAT(/,23H RADSOL OUTPUT, COLUMN=,I3,1H,,I2)
  710 FORMAT(I5,1P3E12.4)
  720 FORMAT(5H    K,9X,4HQEFF,8X,5HRFLUX)
  730 FORMAT(8H SA     =,1PE13.5,5X,7HFAC2  =,1PE13.5,5X,7HSAG    =,
     .1PE13.5,5X,7HALCSKY=,1PE13.5,5X,/,8H ALSURF=,1PE13.5,5X,
     .7HCOSZEN=,1PE13.5,5X,7HHHHH   =,1PE13.5,5X,7HSECZ   =,1PE13.5,5X,
     ./,8H VEG    =,1PE13.5,5X,7HTRANS =,1PE13.5,5X,7HCOEFF1=,1PE13.5,5X,
```

```
     .7HCOEFF2=,1PE13.5,5X,/,8H SS      =,1PE13.5,5X,7HPSA     =,1PE13.5,    RADSOL.72
     .5X,7HFLUXS =,1PE13.5)                                                  RADSOL.73
      RETURN                                                                 RADSOL.74
      END                                                                    RADSOL.75
```

```
      SUBROUTINE RADTER                                                 RADTER.2
C***********************************************************************RADTER.3
C*                                                                     *RADTER.4
C*       "RADTER" DETERMINES THE CONTRIBUTION TO ATMOSPHERIC HEATING   *RADTER.5
C*    FROM LONG WAVE RADIATION. THE PRESCRIPTION UTILIZED IS THAT OF   *RADTER.6
C*    KATAYAMA AS INCLUDED IN THE 3 LEVEL UCLA GCM (SEE PAGE II-4).    *RADTER.7
C*    THE EQUATIONS OF RADIATIVE TRANSFER ARE SOLVED SUBJECT TO THE    *RADTER.8
C*    BOUNDARY CONDITIONS THAT (1) DOWNWARD  IR FLUX IS ZERO AT THE TOP*RADTER.9
C*    OF THE MESH AND (2) THE UPWARD FLUX AT THE EARTH"S SURFACE IS    *RADTER.10
C*    BLACK BODY AT THE SURFACE TEMPERATURE.                           *RADTER.11
C*        THE SOLUTION UTILIZES A BULK TRANSMISSION FUNCTION DEFINED   *RADTER.12
C*    AS THE PRODUCT OF H2O AND CO2 TRANSMISSION FUNCTIONS.            *RADTER.13
C*    ABSORPTION BY OZONE IS NEGLECTED. THE TRANSMISSION FUNCTION      *RADTER.14
C*    IS ASSUMED LINEAR ACROSS A CELL SO THAT TRAPEZOIDAL INTEGRATION  *RADTER.15
C*    CAN BE DONE. THE CORRECTION TERM USED BY KATAYAMA TO ACCOUNT     *RADTER.16
C*    FOR THE NON-LINEARITY IN TAU FOR NEARBY ZONES IS NEGLECTED.      *RADTER.17
C*                                                                     *RADTER.18
C***********************************************************************RADTER.19
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .             TTEN(26,22,15)                                       COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .             PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)      COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),GG(25,21),         COMMON.11
     .             GW(25,21),HRATE(25,21),IJCOLM(25,21)                 COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .             PPRES(30),RELHM(30),UADAT(8),TRDAT(8)                RATELE.1
      COMMON/RADIAT/RADIN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .             RADSO(25,21,16)                                      COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .             SDOTL(21,16)                                         COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),AQ(16),AQI(16),AKI(16),DZ(16),COMMON.20
     .             ALT(16),P(16),SCP(16),DIFM(16),DIFH(16),RI(16),      COMMON.21
     .             DVELDZ(16)                                           COMMON.22
      COMMON/DERHGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .             QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,   COMMON.25
     .             SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,      COMMON.26
     .             DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,     COMMON.27
     .             SOLCON,SOLARN,IRAD,PTOLD,PTNEW,PRECIP,WW,XSAND,      COMMON.28
     .             XCLAY,XORG,XSOIL,GDEPTH,IDEEP,TWATER,GGGG,           COMMON.29
     .             HHHH,FLEE,RRRR,SSSS,NPLT,NPLTT,SUREM                 COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .             ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,         COMMON.32
     .             CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,    COMMON.33
     .             CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,    COMMON.34
     .             RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS,  COMMON.35
     .             GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,      COMMON.36
     .             RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,     COMMON.37
     .             KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4        COMMON.38
```

```
          COMMON/ADARAY/IDUM                                              COMMON.39
          LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI, COMMON.40
     .          PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO         COMMON.41
                                                                          COMMON.42
          TRCO(D)=AMIN1(1.0,1.0-2.1E-5*D)                                 RADTER.21
          TINF = 220.                                                     RADTER.22
                                                                          RADTER.23
C****************************************                                 RADTER.24
C*    BLACK BODY EMISSION               *                                 RADTER.25
C****************************************                                 RADTER.26
C-----DEFINE THE BLACK BODY EMISSION AT ALL SIGMA LEVELS                  RADTER.27
          TDERIV=(T(IS,JS,KMAX)-T(IS,JS,KF))*AKI(KMAX)                    RADTER.28
          TSURF=T(IS,JS,KMAX)+TDERIV*(1.0-A(KMAX))                        RADTER.29
          T4SIG(KMAXP)=5.6692E-8*TSURF**4                                 RADTER.30
          T4SIG(    1)=5.6692E-8*T(IS,JS,1)**4                            RADTER.31
          DO 350 K=2,KMAX                                                 RADTER.32
          TSIGMA =T(IS,JS,K)*TWT(K,1)+T(IS,JS,K-1)*TWT(K,2)               RADTER.33
          T4SIG(K)=5.6692E-8*TSIGMA**4                                    RADTER.34
      350 CONTINUE                                                        RADTER.35
                                                                          RADTER.36
C-----BLACK BODY OF STRATOSPHERE                                          RADTER.37
          T4INF=5.6692E-8*TINF**4                                         RADTER.38
                                                                          RADTER.39
C-----BLACK BODY OF SURFACE - EMISSIVITY GIVEN BY "SUREM"                 RADTER.40
          T4GND=5.6692E-8*TG(IS,JS)**4*SUREM                              RADTER.41
                                                                          RADTER.42
C****************************************                                 RADTER.43
C*    NET UPWARD FLUX                   *                                 RADTER.44
C****************************************                                 RADTER.45
C-----A,B,C TERMS OF KATAYAMA WHERE C IS MODIFIED TO INCLUDE SURFACE      RADTER.46
C-----AND STRATOSPHERIC EFFECTS                                           RADTER.47
          FACGND=T4GND+T4GND-T4SIG(KMAXP)-T4SIG(KMAX)                     RADTER.48
          FACINF=T4SIG(2)-T4SIG(1)                                        RADTER.49
          FLUXL2(KMAXP)=0.0                                               RADTER.50
          FLUXL2(    1)=0.0                                               RADTER.51
                                                                          RADTER.52
C----------FLUX AT INTERIOR POINTS LESS SURF AND STRAT EFFECTS            RADTER.53
          DO 500 K=2,KMAX                                                 RADTER.54
          FLUXL2(K)=0.0                                                   RADTER.55
          IF(K.EQ.2) GO TO 450                                            RADTER.56
          KM1=K-1                                                         RADTER.57
          DO 400 L=2,KM1                                                  RADTER.58
          TAU=TRA1(QEFF(K)-QEFF(L))*TRCO(P(K)-P(L))                       RADTER.59
          FLUXL2(K)=FLUXL2(K)+TAU*(T4SIG(L+1)-T4SIG(L-1))                 RADTER.60
          FLUXL2(L)=FLUXL2(L)+TAU*(T4SIG(K+1)-T4SIG(K-1))                 RADTER.61
      400 CONTINUE                                                        RADTER.62
      450 FLUXL2(K)=FLUXL2(K)+0.5*(T4SIG(K+1)-T4SIG(K-1))                 RADTER.63
                                                                          RADTER.64
C----------ADD SURFACE EFFECT TO INTERIOR ZONES                           RADTER.65
          TAU=TRA1(QEFF(KMAXP)-QEFF(K))*TRCO(P(KMAXP)-P(K))               RADTER.66
          FLUXL2(K)=FLUXL2(K)+TAU*FACGND                                  RADTER.67
                                                                          RADTER.68
C----------FLUX AT SURFACE DUE TO INTERIOR ZONES                          RADTER.69
          FLUXL2(KMAXP)=FLUXL2(KMAXP)+TAU*(T4SIG(K+1)-T4SIG(K-1))         RADTER.70
                                                                          RADTER.71
```

```
      C----------ADD STRATOSPHERIC EFFECT TO INTERIOR ZONES          RADTER.72
            TAU=TRA1(QEFF(K)-QEFF(1))*TRCO(P(K)-P(1))                RADTER.73
            FLUXL2(K)=FLUXL2(K)+TAU*FACINF                           RADTER.74
                                                                     RADTER.75
      C----------FLUX AT TOP OF GRID DUE TO INTERIOR ZONES           RADTER.76
            FLUXL2(1)=FLUXL2(1)+TAU*(T4SIG(K+1)-T4SIG(K-1))          RADTER.77
        500 CONTINUE                                                 RADTER.78
                                                                     RADTER.79
      C----------DETERMINE THE REMAINING BOUNDARY TERMS              RADTER.80
            FLUXL2(KMAXP)=FLUXL2(KMAXP)+0.5*FACGND                   RADTER.81
            FLUXL2(    1)=FLUXL2(    1)+0.5*FACINF                   RADTER.82
            TAU=TRA1(QEFF(KMAXP)-QEFF(1))*TRCO(P(KMAXP)-P(1))        RADTER.83
            FLUXL2(KMAXP)=FLUXL2(KMAXP)+TAU*FACINF                   RADTER.84
            FLUXL2(    1)=FLUXL2(    1)+TAU*FACGND                   RADTER.85
                                                                     RADTER.86
      C----------CALCULATE "A" AND "B" TERMS                         RADTER.87
            DO 600 K=1,KMAXP                                         RADTER.88
            SAVE1(K)=TRA1(QEFF(K)-QEFF(1))                           RADTER.89
            SAVE2(K)=TRCO(P(K)-PTOP)                                 RADTER.90
            SAVE3(K)=TRA2(QEFF(K)-QEFF(1))                           RADTER.91
            TAU=SAVE1(K)*SAVE2(K)                                    RADTER.92
            FLUXL2(K)=FLUXL2(K)+TAU*(2.0*T4SIG(1)-266.0)             RADTER.93
            TAU=SAVE3(K)*SAVE2(K)                                    RADTER.94
            FLUXL2(K)=FLUXL2(K)+TAU*133.0                            RADTER.95
        600 CONTINUE                                                 RADTER.96
            FLUXL2(1)=FLUXL2(2)                                      RADTER.97
                                                                     RADTER.98
      C-----DEBUG PRINT                                              RADTER.99
            IF(LOUT.LE.0) RETURN                                     RADTER.100
            IF(IJCOLM(IS,JS).LE.0) RETURN                            RADTER.101
        620 WRITE(6,700) IS,JS                                       RADTER.102
            WRITE(6,705)                                             RADTER.103
            DO 650 K=1,KMAXP                                         RADTER.104
            WRITE(6,710) K,FLUXL2(K),QEFF(K),P(K),T4SIG(K),SAVE1(K),SAVE2(K),  RADTER.105
           .SAVE3(K)                                                 RADTER.106
        650 CONTINUE                                                 RADTER.107
        700 FORMAT(/,23H RADTER OUTPUT, COLUMN=,I3,1H,,I2)           RADTER.108
        705 FORMAT(5H    K,11X,6HFLUXL2,13X,4HQEFF,16X,1HP,12X,5HT4SIG,  RADTER.109
           .13X,4HTRA1,13X,4HTRCO,13X,4HTRA2)                        RADTER.110
        710 FORMAT(I5,1P7E17.9)                                      RADTER.111
            RETURN                                                   RADTER.112
            END                                                      RADTER.113
```

```
      SUBROUTINE SOLAR                                                  SOLAR.2
C***********************************************************************SOLAR.3
C*                                                                     *SOLAR.4
C*         "SOLAR" DETERMINES THE SOLAR INSOLATION FOR THE CURRENT     *SOLAR.5
C*    SIMULATION TIME.                                                 *SOLAR.6
C*                                                                     *SOLAR.7
C***********************************************************************SOLAR.8
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TROAT(8)               BATELE.1
      COMMON/RADIAT/RADTN(25,21,16),PADTO(25,21,16),PADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/FG(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),RI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,ELEE,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO         COMMON.41
                                                                        COMMON.42
                                                                        SOLAR.10
C-----FORM SOLAR CONSTANT FOR CURRENT EARTH-SUN DISTANCE                SOLAR.11
C-----AZMUTH IS DEFINED TO BE ZERO AT MIDNIGHT GMT ON JAN 1.            SOLAR.12
C     (DAYNOW=NO. OF DAYS (INCLUDING FRACTIONS) MEASURED FROM MIDNIGHT  SOLAR.13
C     GMT ON JAN 1. )                                                   SOLAR.14
      DAYNOW=GMDYBG+SIMTIM*1.1574074074E-5                              SOLAR.15
      AZMUTH=(DAYNOW-1.0)*0.01720806                                    SOLAR.16
```

```
      COSAZM=COS(AZMUTH)
      SINAZM=SIN(AZMUTH)
      SOLCON=1353.2/(1.0-0.016733*COSAZM)**2

C-----DETERMINE CURRENT DECLINATION OF THE SUN
      ANGSUN=4.885783-AZMUTH
      SIGMA=ANGSUN+0.007133*SIN(ANGSUN)+0.03268*COS(ANGSUN)
     1-0.000318*SIN(ANGSUN+ANGSUN)+0.000145*COS(ANGSUN+ANGSUN)
      SINDEC=0.3978*SIN(SIGMA)
      COSDEC=SQRT(1.0-SINDEC*SINDEC)

C-----DETERMINE THE CURRENT HOUR ANGLE
      B=3.141592+0.032351*SINAZM-0.0011228*COSAZM+0.040267
     1*SIN(AZMUTH+AZMUTH)+0.0159129*COS(AZMUTH+AZMUTH)
      HRANGL=(GMTMHR*3600.0+SIMTIM)*7.272205E-5 -B-RLONG
      HHH=AMOD(HRANGL*3.8197186+36.0,24.0)

C-----FORM SOLAR ZENITH ANGLE
      SINLAT=SIN(RLAT)
      COSLAT=COS(RLAT)
      COEFF1=SINDEC*SINLAT
      COEFF2=COSDEC*COSLAT
      COSZEN=COEFF1+COEFF2*COS(HRANGL)
      SINZEN=SQRT(1.0-COSZEN*COSZEN)
      TANZEN=0.0
      IF(ABS(COSZEN).GT.0.0) TANZEN=SINZEN/COSZEN

C-----FORM CURRENT NORMAL INSOLATION
      SOLARN=SOLCON*COSZEN

C-----MISCELLANEOUS
      SA=0.349*SOLARN
      SS=SOLARN-SA
      SINAZ=COSDEC/SINZEN*SIN(HRANGL)
      COSAZ=(SINLAT*COSZEN-SINDEC)/(COSLAT*SINZEN)
      FACTOM=TANZEN*(SINAZ*COSANG+COSAZ*SINANG)

C-----DEBUG PRINT
      IF(LOUT.LE.0) RETURN
      WRITE(6,650)
      WRITE(6,600) DAYNOW,AZMUTH,SOLCON,SIGMA,B,RLONG,HRANGL,HHH,RLAT,
     .COEFF1,COEFF2,COSZEN,TANZEN,SOLARN,SINAZ,COSAZ,FACTOM
C-----I/O FORMATS

  600 FORMAT(8H DAYNOW=,1PE13.5,5X,7HAZMUTH=,1PE13.5,5X,7HSOLCON=,
     .1PE13.5,5X,7HSIGMA =,1PE13.5,5X,/,8H B      =,1PE13.5,5X,7HRLONG =,
     .1PE13.5,5X,7HHRANGL=,1PE13.5,5X,7HHHH    =,1PE13.5,5X,7HRLAT  =,
     .1PE13.5,/,8H COEFF1=,1PE13.5,5X,7HCOEFF2=,1PE13.5,5X,7HCOSZEN=,
     .1PE13.5,5X,7HTANZEN=,1PE13.5,5X,7HSOLARN=,1PE13.5,5X,/,8H SINAZ =,
     .1PE13.5,5X,7HCOSAZ =,1PE13.5,5X,7HFACTOM=,1PE13.5,5X)
  650 FORMAT(/,13H SOLAR OUTPUT)
      RETURN
      END
```

```
          SUBROUTINE TENDH                                                TENDH.2
                                                                          NEW.236
C-----HORIZONTAL ADVECTION TERMS IN MOMENTUM AND CONSERVATION EQUATIONS NEW.237
C-----ARE CALCULATED IN THIS SUBROUTINE                                   NEW.238
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATELF.1
      COMMON/RADIAT/RADIN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/FG(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),RI(16),     COMMON.21
     .              OVELAZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),OEFF(16),            COMMON.24
     .              OEFFG,VEG,TRANS,COSANG,SINANG,PLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTO4,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RIOLD,RINEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,FLEF,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADIN,RADTO,RADSN,RADSO         COMMON.41
                                                                        COMMON.42
                                                                        TENDH.4
C****************************************                              TENDH.5
C*    HORIZONTAL ADVECTION OF SCALARS   *                              TENDH.6
C****************************************                              TENDH.7
C-----X DIRECTION                                                       TENDH.8
      DO 200 K=1,KMAX                                                   TENDH.9
      DO 200 J=1,JE                                                     TENDH.10
      KS=K                                                              TENDH.11
      JS=J                                                              TENDH.12
      ES=1                                                              TENDH.13
```

120

```
      JP1=J+1                                                           TENDH.14
      IF(JE.EQ.1) JP1=1                                                 TENDH.15
      CBB=CA(1,J,K)                                                     TENDH.16
      TBB=TA(1,J,K)                                                     TENDH.17
      UAVG=U(1,JP1,K)+U(1,J,K)                                          TENDH.18
      IF(UAVG.LE.0.0) GO TO 160                                         TENDH.19
      PSIG=PSA(1,J)*A(K)+PTOP                                           TENDH.20
      CBB=ENTERP(PSIG,3)*PSA(1,J)                                       TENDH.21
      TBB=ENTERP(PSIG,4)*PSA(1,J)                                       TENDH.22
      TBB=TBB*(PBOT/PSIG)**0.286                                        TENDH.23
  160 FLX1=2.0*UAVG*CBB                                                 TENDH.24
      FLX3=2.0*UAVG*TBB                                                 TENDH.25
      DO 170 I=1,IEM1                                                   TENDH.26
      IP1=I+1                                                           TENDH.27
      PIUAVG=UA(IP1,JP1,K)+UA(IP1,J,K)                                  NEW.239
      FLX2=PIUAVG*(C(IP1,J,K)+C(I,J,K))                                 NEW.240
      FLX4=PIUAVG*(TA(IP1,J,K)/PSA(IP1,J)+TA(I,J,K)/PSA(I,J))           NEW.241
      CTEN(I,J,K)=CTEN(I,J,K)-(FLX2-FLX1)*PDX4                          TENDH.31
      TTEN(I,J,K)=TTEN(I,J,K)-(FLX4-FLX3)*PDX4                          TENDH.32
      FLX1=FLX2                                                         TENDH.33
      FLX3=FLX4                                                         TENDH.34
  170 CONTINUE                                                          TENDH.35
      CBB=CA(IE,J,K)                                                    TENDH.36
      TBB=TA(IE,J,K)                                                    TENDH.37
      UAVG=U(IEND,JP1,K)+U(IEND,J,K)                                    TENDH.38
      IF(UAVG.GE.0.0) GO TO 180                                         TENDH.39
      PSIG=PSA(IE,J)*A(K)+PTOP                                          TENDH.40
      IS=IE                                                             TENDH.41
      CBB=ENTERP(PSIG,3)*PSA(IE,J)                                      TENDH.42
      TBB=ENTERP(PSIG,4)*PSA(IE,J)                                      TENDH.43
      TBB=TBB*(PBOT/PSIG)**0.286                                        TENDH.44
  180 FLX2=2.0*UAVG*CBB                                                 TENDH.45
      FLX4=2.0*UAVG*TBB                                                 TENDH.46
      CTEN(IE,J,K)=CTEN(IE,J,K)-(FLX2-FLX1)*PDX4                        TENDH.47
      TTEN(IE,J,K)=TTEN(IE,J,K)-(FLX4-FLX3)*PDX4                        TENDH.48
  200 CONTINUE                                                          TENDH.49
                                                                        TENDH.50
C-----Y DIRECTION                                                       TENDH.51
      IF(JE.EQ.1) GO TO 301                                             TENDH.52
      DO 300 K=1,KMAX                                                   TENDH.53
      DO 300 I=1,IE                                                     TENDH.54
      KS=K                                                              TENDH.55
      IS=I                                                              TENDH.56
      JS=1                                                              TENDH.57
      IP1=I+1                                                           TENDH.58
      CBB=CA(I,1,K)                                                     TENDH.59
      TBB=TA(I,1,K)                                                     TENDH.60
      VAVG=V(IP1,1,K)+V(I,1,K)                                          TENDH.61
      IF(VAVG.LE.0.0) GO TO 210                                         TENDH.62
      PSIG=PSA(I,1)*A(K)+PTOP                                           TENDH.63
      CBB=ENTERP(PSIG,3)*PSA(I,1)                                       TENDH.64
      TBB=ENTERP(PSIG,4)*PSA(I,1)                                       TENDH.65
      TBB=TBB*(PBOT/PSIG)**0.286                                        TENDH.66
  210 FLX1=2.0*VAVG*CBB                                                 TENDH.67
      FLX3=2.0*VAVG*TBB                                                 TENDH.68
```

```
      DO 270 J=1,JEM1                                                 TENDH.69
      JP1=J+1                                                         TENDH.70
      PIVAVG=VA(IP1,JP1,K)+VA(I,JP1,K)                                NEW.242
      FLX2=PIVAVG*(C(I,JP1,K)+C(I,J,K))                               NEW.243
      FLX4=PIVAVG*(TA(I,JP1,K)/PSA(I,JP1)+TA(I,J,K)/PSA(I,J))         NEW.244
      CTEN(I,J,K)=CTEN(I,J,K)-(FLX2-FLX1)*RDY4                        TENDH.74
      TTEN(I,J,K)=TTEN(I,J,K)-(FLX4-FLX3)*RDY4                        TENDH.75
      FLX1=FLX2                                                       TENDH.76
      FLX3=FLX4                                                       TENDH.77
  270 CONTINUE                                                        TENDH.78
      TBB=TA(I,JE,K)                                                  TENDH.79
      CBB=CA(I,JE,K)                                                  TENDH.80
      VAVG=V(IP1,JEMD,K)+V(I,JEMD,K)                                  TENDH.81
      IF(VAVG.GE.0.0) GO TO 280                                       TENDH.82
      PSIG=PSA(I,JE)*A(K)+PTOP                                        TENDH.83
      JS=JE                                                           TENDH.84
      CBB=FNTERP(PSIG,3)*PSA(I,JE)                                    TENDH.85
      TBB=FNTERP(PSIG,4)*PSA(I,JE)                                    TENDH.86
      TBB=TBB*(PBOT/PSIG)**0.286                                      TENDH.87
  280 FLX2=2.0*VAVG*CBB                                               TENDH.88
      FLX4=2.0*VAVG*TBB                                               TENDH.89
      CTEN(I,JE,K)=CTEN(I,JE,K)-(FLX2-FLX1)*RDY4                      TENDH.90
      TTEN(I,JE,K)=TTEN(I,JE,K)-(FLX4-FLX3)*RDY4                      TENDH.91
  300 CONTINUE                                                        TENDH.92
  301 CONTINUE                                                        TENDH.93
                                                                      TENDH.94
C*************************************                                TENDH.95
C*    MOMENTUM ADVECTION             *                                TENDH.96
C*************************************                                TENDH.97
C-----X DIRECTION                                                     TENDH.98
      DO 400 K=1,KMAX                                                 TENDH.99
      DO 400 J=JBEG,JE                                                TENDH.100
      JM1=J-1                                                         TENDH.101
      JP1=J+1                                                         TENDH.102
      IF(J.NE.1) GO TO 401                                            TENDH.103
      JP1=1                                                           TENDH.104
      JM1=1                                                           TENDH.105
  401 UAF=UA(2,JP1,K)+UA(1,JP1,K)+2.0*(UA(2,J,K)+UA(1,J,K))          TENDH.106
     .   +UA(2,JM1,K)+UA(1,JM1,K)                                     TENDH.107
      FLX1=UAF*(U(2,J,K)-U(1,J,K))                                    TENDH.108
      FLX3=UAF*(V(2,J,K)-V(1,J,K))                                    TENDH.109
      DO 400 I=2,IE                                                   TENDH.110
      IP1=I+1                                                         TENDH.111
      UAF=UA(IP1,JP1,K)+UA(I,JP1,K)+2.0*(UA(IP1,J,K)+UA(I,J,K))      TENDH.112
     .  +UA(IP1,JM1,K)+UA(I,JM1,K)                                    TENDH.113
      FLX2=UAF*(U(IP1,J,K)+U(I,J,K))                                  TENDH.114
      FLX4=UAF*(V(IP1,J,K)+V(I,J,K))                                  TENDH.115
      UTEN(I,J,K)=UTEN(I,J,K)-(FLX2-FLX1)*RDX16                       TENDH.116
      VTEN(I,J,K)=VTEN(I,J,K)-(FLX4-FLX3)*RDX16                       TENDH.117
      FLX1=FLX2                                                       TENDH.118
      FLX3=FLX4                                                       TENDH.119
  400 CONTINUE                                                        TENDH.120
                                                                      TENDH.121
C-----Y DIRECTION                                                     TENDH.122
      IF(JE.EQ.1) GO TO 501                                           TENDH.123
```

```
      DO 500 K=1,KMAX
      DO 500 I=2,IE
      IP1=I+1
      IM1=I-1
      VAF=VA(IP1,2,K)+VA(IP1,1,K)+2.0*(VA(I,2,K)+VA(I,1,K))
     +   +VA(IM1,2,K)+VA(IM1,1,K)
      FLX1=VAF*(U(I,2,K)+U(I,1,K))
      FLX3=VAF*(V(I,2,K)+V(I,1,K))
      DO 500 J=2,JE
      JP1=J+1
      VAF=VA(IP1,JP1,K)+VA(IP1,J,K)+2.0*(VA(I,JP1,K)+VA(I,J,K))
     +   +VA(IM1,JP1,K)+VA(IM1,J,K)
      FLX2=VAF*(U(I,JP1,K)+U(I,J,K))
      FLX4=VAF*(V(I,JP1,K)+V(I,J,K))
      UTEN(I,J,K)=UTEN(I,J,K)-(FLX2-FLX1)*RDY16
      VTEN(I,J,K)=VTEN(I,J,K)-(FLX4-FLX3)*RDY16
      FLX1=FLX2
      FLX3=FLX4
  500 CONTINUE
  501 CONTINUE

C-----DEBUG PRINT
      IF(LOUT.LE.0) RETURN
      DO 550 J=1,JE
      DO 550 I=1,IE
      IF(IJCOLM(I,J).LE.0) GO TO 550
      WRITE(6,600) I,J
      WRITE(6,610)
      DO 525 K=1,KMAX
      WRITE(6,620) K,UTEN(I,J,K),VTEN(I,J,K),
     .TTEN(I,J,K),CTEN(I,J,K)
  525 CONTINUE
  550 CONTINUE
  600 FORMAT(22H TENDH OUTPUT, COLUMN=,I2,1H,,I2)
  610 FORMAT(6H     K,3X,4HUTEN,8X,4HVTEN,8X,4HTTEN,8X,4HCTEN)
  620 FORMAT(I5,1P4E12.4)
      RETURN
      END
```

```
      SUBROUTINE TENDV                                                  TENDV.2
                                                                        NEW.245
C-----VERTICAL   ADVECTION TERMS IN MOMENTUM AND CONSERVATION EQUATIONS NEW.246
C-----ARE CALCULATED IN THIS SUBROUTINE                                 NEW.247
                                                                        NEW.248
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATELE.1
      COMMON/RADIAT/RADIN(25,21,16),RADIO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FU(16),FV(16),ET(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),PI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/REFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),           COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,RLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,EEEE,RRRR,SSSS,NPLT,NPLTT,SURFM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IF,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,TNPRS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .       PHIPIX,PHIPIY,QA,QB,HDIFH,RADIN,RADIO,RADSN,RADSO          COMMON.41
                                                                        COMMON.42
      PTEN(IS,JS)=0.0                                                   TENDV.4
C-----CALCULATE DIVERGENCE, "PI" TENDENCY                              TENDV.5
      PSAI=1.0/PSA(IS,JS)                                               TENDV.6
      PTEN(IS,JS)=0.0                                                   TENDV.7
      DO 200 K=1,KMAX                                                   TENDV.8
      S=(UA(IS+1,JS+1,K)+UA(IS+1,JS,K)-UA(IS,JS+1,K)-UA(IS,JS,K))*RDXH  TENDV.9
     . +(VA(IS+1,JS+1,K)+VA(IS,JS+1,K)-VA(IS+1,JS,K)-VA(IS,JS,K))*RDYH  TENDV.10
      SCR(K)=S*DQ(K)                                                    TENDV.11
      PTEN(IS,JS)=PTEN(IS,JS)-SCR(K)                                    TENDV.12
```

```
      200 CONTINUE                                                          TENDV.13
                                                                            TENDV.14
C-----CALCULATE SIGMA DOT                                                   TENDV.15
          IF(IS.EQ.1) GO TO 300                                             TENDV.16
          DO 250 K=1,KMAXP                                                  TENDV.17
          SDOTL(JS,K)=SDOTR(JS,K)                                           TENDV.18
      250 CONTINUE                                                          TENDV.19
      300 SDOTR(JS,1)=0.0                                                   TENDV.20
          DO 350 K=2,KMAXP                                                  TENDV.21
          SDOTR(JS,K)=SDOTR(JS,K-1)-(PTEN(IS,JS)*DQ(K-1)+SGR(K-1))*PSAI     TENDV.22
      350 CONTINUE                                                          TENDV.23
                                                                            TENDV.24
C-----VERTICAL ADVECTION OF SCALARS                                         TENDV.25
          FTM=0.0                                                           TENDV.26
          FCM=0.0                                                           TENDV.27
          DO 400 K=1,KMAX                                                   TENDV.28
          KP=K+1                                                            TENDV.29
          IF(K.LT.KMAX) GO TO 380                                           TENDV.30
          FCP=0.0                                                           TENDV.31
          FTP=0.0                                                           TENDV.32
          GO TO 390                                                         TENDV.33
      380 FTP=SDOTR(JS,KP)*(TWT(KP,2)*TA(IS,JS,K)+TWT(KP,1)*TA(IS,JS,KP))   TENDV.34
          FCP=SDOTR(JS,KP)*(TWT(KP,2)*CA(IS,JS,K)+TWT(KP,1)*CA(IS,JS,KP))   TENDV.35
      390 CONTINUE                                                          TENDV.36
          TTEN(IS,JS,K)=TTEN(IS,JS,K)-(FTP-FTM)*DQI(K)                      TENDV.37
          CTEN(IS,JS,K)=CTEN(IS,JS,K)-(FCP-FCM)*DQI(K)                      TENDV.38
          FTM=FTP                                                           TENDV.39
          FCM=FCP                                                           TENDV.40
      400 CONTINUE                                                          TENDV.41
                                                                            TENDV.42
          JSM1=JS-1                                                         TENDV.43
          IF(JE.EQ.1) JSM1=1                                                TENDV.44
          IF(JE.EQ.1) GO TO 101                                             TENDV.45
          IF(IS.EQ.1.OR.JS.EQ.1) GO TO 498                                  TENDV.46
      101 IF(IS.EQ.1) GO TO 498                                             TENDV.47
C-----VERTICAL ADVECTION OF MOMENTUM                                        TENDV.48
          IF(IS.EQ.1) GO TO 498                                             TENDV.49
          FUM=0.0                                                           TENDV.50
          FVM=0.0                                                           TENDV.51
          DO 495 K=1,KMAX                                                   TENDV.52
          KP=K+1                                                            TENDV.53
          IF(JS.GT.1) GO TO 480                                             TENDV.54
          SDOTAV=0.5*(SDOTR(1,KP)+SDOTL(1,KP))                              TENDV.55
          GO TO 485                                                         TENDV.56
      480 SDOTAV=0.25*(SDOTR(JS,KP)+SDOTR(JSM1,KP)                          TENDV.57
         *          +SDOTL(JS,KP)+SDOTL(JSM1,KP))                           TENDV.58
      485 IF(K.LT.KMAX) GO TO 490                                           TENDV.59
          FUP=0.0                                                           TENDV.60
          FVP=0.0                                                           TENDV.61
          SAVE2(K)=SDOTAV                                                   TENDV.62
          GO TO 492                                                         TENDV.63
      490 FUP=SDOTAV*(TWT(K+1,2)*UA(IS,JS,K)+TWT(K+1,1)*UA(IS,JS,K+1))      TENDV.64
          FVP=SDOTAV*(TWT(K+1,2)*VA(IS,JS,K)+TWT(K+1,1)*VA(IS,JS,K+1))      TENDV.65
          SAVE2(K)=SDOTAV                                                   TENDV.66
      492 UTEN(IS,JS,K)=UTEN(IS,JS,K)-(FUP-FUM)*DQI(K)                      TENDV.67
```

```
      VTEN(IS,JS,K)=VTEN(IS,JS,K)-(FVP-FVM)*DDI(K)                      TENDV.68
      FUM=FUP                                                           TENDV.69
      FVM=FVP                                                           TENDV.70
  495 CONTINUE                                                          TENDV.71
                                                                        TENDV.72
C-----CORIOLIS TERMS                                                    TENDV.73
C     BOUNDARY-LAYER PROFILE IS NOT USED FOR INTERPOLATING UGEO,VGEO    TENDV.74
      PSAVE=PCUT(IS,JS)                                                 TENDV.75
      PCUT(IS,JS)=PUA(IS,JS)+PTOP                                       TENDV.76
      DO 102 K=1,KMAX                                                   TENDV.77
      PSIG=PUA(IS,JS)*A(K)+PTOP                                         TENDV.78
      KS=K                                                              TENDV.79
      UGEO=ENTERP(PSIG,1)                                               TENDV.80
      VGEO=ENTERP(PSIG,2)                                               TENDV.81
      UTEN(IS,JS,K)=UTEN(IS,JS,K)+F*(VA(IS,JS,K)-PUA(IS,JS)*VGEO)       TENDV.82
      VTEN(IS,JS,K)=VTEN(IS,JS,K)-F*(UA(IS,JS,K)-PUA(IS,JS)*UGEO)       TENDV.83
  102 CONTINUE                                                          TENDV.84
      PCUT(IS,JS)=PSAVE                                                 TENDV.85
                                                                        TENDV.86
C-----DEBUG PRINT                                                       TENDV.87
  498 IF(LOUT.LE.0) RETURN                                              TENDV.88
      IF(IJCOLM(IS,JS).LE.0) RETURN                                     TENDV.89
      WRITE(6,620) IS,JS                                                TENDV.90
      WRITE(6,625)                                                      TENDV.91
      DO 500 K=1,KMAX                                                   TENDV.92
      WRITE(6,600) K,SCR(K),SDOTR(JS,K),SAVE2(K),SAVE3(K),UTEN(IS,JS,K),TENDV.93
     .VTEN(IS,JS,K),TTEN(IS,JS,K),CTEN(IS,JS,K),SDOTL(JS,K),            TENDV.94
     .SDOTR(JS-1,K)                                                     TENDV.95
  500 CONTINUE                                                          TENDV.96
      WRITE(6,610) PTEN(IS,JS)                                          TENDV.97
  600 FORMAT(I5,1P10E12.4)                                              TENDV.98
  610 FORMAT(6H PTEN=,1PE12.4)                                          TENDV.99
  620 FORMAT(22H0TENDV OUTPUT, COLUMN=,I2,1H,,I2)                       TENDV.100
  625 FORMAT(2H0K,9X,3HSGR,6X,6HSDOTR ,6X,6HSDOTAV,8X,4HOMEG,           TENDV.101
     .8X,4HUTEN,8X,4HVTEN,8X,4HTTEN,8X,4HCTEN,8X,5HSDOTL,6X,6HSDOTR-)   TENDV.102
      RETURN                                                            TENDV.103
      END                                                               TENDV.104
```

```
      FUNCTION TRA1(USTAR)                                              TRA1.2
C***********************************************************************TRA1.3
C*                                                                     *TRA1.4
C*       FORMS WATER-VAPOR-TRANSMISSION-FUNCTION-AVERAGED-OVER         *TRA1.5
C*    TEMPERATURE. USES OPTION E OF KATAYAMA ON PAGE II-18. ROUTINE    *TRA1.6
C*    ACTUALLY FORMS TAU/2.                                            *TRA1.7
C*    NOTE THE COEFFICIENTS ARE IN CGS UNITS.                          *TRA1.8
C*                                                                     *TRA1.9
C***********************************************************************TRA1.10
                                                                        TRA1.11
      IF(USTAR.LT.0.0001) GO TO 300                                     TRA1.12
      Z=ALOG10(USTAR)                                                   TRA1.13
                                                                        TRA1.14
C-----FOR "USTAR" GREATER THAN OR EQUAL TO 1.0                          TRA1.15
      IF(USTAR.LT.1.0) GO TO 200                                        TRA1.16
      TRA1=AMAX1(0.0,0.1865+(0.0175*Z-0.137)*Z)                         TRA1.17
      RETURN                                                            TRA1.18
                                                                        TRA1.19
C-----FOR "USTAR" GREATER THAN OR EQUAL TO 0.0001 AND LESS THAN 1.0     TRA1.20
  200 TRA1=0.1865-(0.12975+0.01375*Z)*Z                                 TRA1.21
      RETURN                                                            TRA1.22
                                                                        TRA1.23
C-----FOR "USTAR" LESS THAN 0.0001                                      TRA1.24
  300 TRA1=1.0/(2.0+597.4*USTAR)                                        TRA1.25
      RETURN                                                            TRA1.26
      END                                                               TRA1.27
```

```
      FUNCTION TRA2(USTAR)                                              TRA2.2
C***********************************************************************TRA2.3
C*                                                                     *TRA2.4
C*       FORMS WATER VAPOR TRANSMISSION FUNCTION TAU TILDE USING       *TRA2.5
C*    B" OPTION OF KATAYAMA ON PAGE II-18.                             *TRA2.6
C*    NOTE THE COEFFICIENTS ARE IN CGS UNITS.                          *TRA2.7
C*                                                                     *TRA2.8
C***********************************************************************TRA2.9
                                                                        TRA2.10
      IF(USTAR.LT.0.0001) GO TO 300                                     TRA2.11
      IF(USTAR.GE.10.0) GO TO 500                                       TRA2.12
      Z=ALOG10(USTAR)                                                   TRA2.13
      IF(USTAR.LT.0.1) GO TO 200                                        TRA2.14
                                                                        TRA2.15
C-----FOR "USTAR" GREATER THAN OR EQUAL TO 0.1 AND LESS THAN 10.0       TRA2.16
      IF(USTAR.LT.0.1) GO TO 200                                        TRA2.17
      TRA2=AMAX1(0.0,0.254+(0.0205*Z-0.1985)*Z)                         TRA2.18
      RETURN                                                            TRA2.19
                                                                        TRA2.20
C-----FOR "USTAR" GREATER THAN OR EQUAL TO 0.0001 AND LESS THAN 0.1     TRA2.21
  200 TRA2=0.216-(0.2827+0.0258*Z)*Z                                    TRA2.22
      RETURN                                                            TRA2.23
                                                                        TRA2.24
C-----FOR "USTAR" LESS THAN 0.0001                                      TRA2.25
  300 TRA2=1.0/(1.0+2.95*USTAR**0.408)                                  TRA2.26
      RETURN                                                            TRA2.27
                                                                        TRA2.28
C-----ARGUMENT OUT OF RANGE OF FORMULAE APPLICABILITY                   TRA2.29
  500 WRITE(6,510) USTAR                                                TRA2.30
  510 FORMAT(80H1PROGRAM TERMINATING IN "TRA2", ARGUMENT OUT OF RANGE OFTRA2.31
     1 FORMULAE APPLICABILITY.,5X,6HUSTAR=,1PE12.5)                     TRA2.32
      STOP                                                              TRA2.33
      END                                                               TRA2.34
```

```
      SUBROUTINE TRIDG(K,A,B,C,D,X,NI,NF)                    TRA2.35
                                                             TRA2.36
      DIMENSION A(NF),B(NF),C(NF),D(NF),X(NF)                TRA2.37
      DIMENSION E(101),F(101), XA(101)                       TRA2.38
                                                             TRA2.39
      IF((NF-NI).LT.1) STOP"TRIDG"                           TRA2.40
      IF(K.EQ.2) GO TO 3                                     TRA2.41
    1 NIP=NI+1                                               TRA2.42
      E(NI)=C(NI)/B(NI)                                      TRA2.43
      DO 2 I=NIP,NF                                          TRA2.44
      XA(I)=1.0/(B(I)-A(I)*E(I-1))                           TRA2.45
    2 E(I)=C(I)*XA(I)                                        TRA2.46
    3 F(NI)=D(NI)/B(NI)                                      TRA2.47
      DO 4 I=NIP,NF                                          TRA2.48
    4 F(I)=(D(I)-A(I)*F(I-1))*XA(I)                          TRA2.49
      X(NF)=F(NF)                                            TRA2.50
      DO 5 II=NIP,NF                                         TRA2.51
      I=NF+NI-II                                             TRA2.52
    5 X(I)=F(I)-E(I)*X(I+1)                                  TRA2.53
                                                             TRA2.54
      RETURN                                                 TRA2.55
      END                                                    TRA2.56
```

```
      SUBROUTINE UPDATE                                                 UPDATE.2
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15) COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15) COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)     COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),       COMMON.5
     .              TTEN(26,22,15)                                      COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)     COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)           COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),        COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)     COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CS(25,21),         COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),  COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)               BATELE.1
      COMMON/RADIAT/RADTN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),    COMMON.15
     .              RADSO(25,21,16)                                     COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),           COMMON.17
     .              SDOTL(21,16)                                        COMMON.18
      COMMON/MATRIX/FC(16),FT(16),FH(16),FV(16),FT(16),EU(16)           COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16),COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),PI(16),     COMMON.21
     .              DVELDZ(16)                                          COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)   COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),            COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,PLAT,RLONG,DLONG,SS,  COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,     COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,    COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,     COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,TDEEP,TWATER,GGGG,          COMMON.29
     .              HHHH,ELEF,RRRR,SSSS,NPLT,NPLTT,SUREM                COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,   COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,        COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,   COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IF,IEND,IEM1,IS,KS,DX,   COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS, COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,     COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,RINV,INPPS,GAM1,    COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4       COMMON.38
      COMMON/ADARAY/IDUM                                                COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,  COMMON.40
     .    PHIPIX,PHIPIY,QA,QB,HDIFH,RADTN,RADTO,RADSN,RADSO             COMMON.41
                                                                        COMMON.42
                                                                        UPDATE.4
C-----UPDATE ALL PRIMARY VARIABLES                                      UPDATE.5
                                                                        UPDATE.6
C---------A FREQUENCY FILTER FOR TIME INTEGRATION IS USED (ASSELIN M.W.R.UPDATE.7
C---------P.487). ALPH=0. CORRESPONDS TO NO FILTER. FOR BEST RESULTS TIMEUPDATE.8
C---------DEPENDANT VALUES FOR ALPH SHOULD BE USED                      UPDATE.9
                                                                        UPDATE.10
C-----SCALARS                                                           UPDATE.11
      PSAVE=PSA(IS,JS)                                                  UPDATE.12
      PSA(IS,JS)=PSB(IS,JS)+DT*PTEN(IS,JS)                              UPDATE.13
      PSB(IS,JS)=PSAVE+0.5*ALPH*(PSB(IS,JS)-2.0*PSAVE+PSA(IS,JS))       UPDATE.14
      DO 100 K=1,KMAX                                                   UPDATE.15
      TSAVE=TA(IS,JS,K)                                                 UPDATE.16
```

```
              CSAVE=CA(IS,JS,K)                                                  UPDATE.17
              TA(IS,JS,K)=TB(IS,JS,K)+DT*TTEN(IS,JS,K)                           UPDATE.18
              CA(IS,JS,K)=CB(IS,JS,K)+DT*CTEN(IS,JS,K)                           UPDATE.19
              IF(CA(IS,JS,K).LT.0.0) CA(IS,JS,K)=0.0                             UPDATE.20
              TB(IS,JS,K)=TSAVE+0.5*ALPH*(TB(IS,JS,K)-2.0*TSAVE+TA(IS,JS,K))     UPDATE.21
              CB(IS,JS,K)=CSAVE+0.5*ALPH*(CB(IS,JS,K)-2.0*CSAVE+CA(IS,JS,K))     UPDATE.22
          100 CONTINUE                                                           UPDATE.23
                                                                                 UPDATE.24
        C-----VECTORS                                                            UPDATE.25
              IF(JE.EQ.1) GO TO 201                                              UPDATE.26
              IF(IS.EQ.1.OR.JS.EQ.1) GO TO 300                                   UPDATE.27
          201 IF(IS.EQ.1) GO TO 300                                              UPDATE.28
              DO 200 K=1,KMAX                                                    UPDATE.29
              USAVE=UA(IS,JS,K)                                                  UPDATE.30
              VSAVE=VA(IS,JS,K)                                                  UPDATE.31
              UA(IS,JS,K)=UB(IS,JS,K)+DT*UTEN(IS,JS,K)                           UPDATE.32
              VA(IS,JS,K)=VB(IS,JS,K)+DT*VTEN(IS,JS,K)                           UPDATE.33
              UB(IS,JS,K)=USAVE+0.5*ALPH*(UB(IS,JS,K)-2.0*USAVE+UA(IS,JS,K))     UPDATE.34
              VB(IS,JS,K)=VSAVE+0.5*ALPH*(VB(IS,JS,K)-2.0*VSAVE+VA(IS,JS,K))     UPDATE.35
          200 CONTINUE                                                           UPDATE.36
          300 CONTINUE                                                           UPDATE.37
                                                                                 UPDATE.38
        C-----DEBUG PRINT                                                        UPDATE.39
              IF(LOUT.LE.0) RETURN                                               UPDATE.40
              IF(IJCOLM(IS,JS).LE.0) RETURN                                      UPDATE.41
              WRITE(6,700) IS,JS                                                 UPDATE.42
              WRITE(6,730)                                                       UPDATE.43
              DO 610 K=1,KMAX                                                    UPDATE.44
              WRITE (6,710) K,UA(IS,JS,K),UB(IS,JS,K),VA(IS,JS,K),VB(IS,JS,K),   UPDATE.45
             .                 TA(IS,JS,K),TB(IS,JS,K),CA(IS,JS,K),CB(IS,JS,K)   UPDATE.46
          610 CONTINUE                                                           UPDATE.47
              WRITE (6,740) PSA(IS,JS),PSB(IS,JS)                                UPDATE.48
          700 FORMAT(23H0UPDATE OUTPUT, COLUMN=,I2,1H,,I2)                       UPDATE.49
          710 FORMAT(I5,1P9E12.4)                                                UPDATE.50
          730 FORMAT(5H    K,10X,2HUA,10X,2HUB,10X,2HVA,10X,2HVB,10X,2HTA,10X,   UPDATE.51
             .2HTB,10X,2HCA,10X,2HCB)                                            UPDATE.52
          740 FORMAT(5H PSA=,1PE12.4,5X,5H PSB=,1PE12.4)                         UPDATE.53
              RETURN                                                             UPDATE.54
              END                                                                UPDATE.55
```

```
      SUBROUTINE UVBOUN                                                  UVBOUN.2
C                                                                        UVBOUN.3
C-----THIS SETS THE LATERAL BOUNDARY CONDITIONS FOR THE VECTORS           UVBOUN.4
C                                                                        UVBOUN.5
      COMMON/ARRAY1/UA(26,22,15),VA(26,22,15),CA(25,21,15),TA(25,21,15)  COMMON.2
      COMMON/ARRAY2/UB(26,22,15),VB(26,22,15),CB(25,21,15),TB(25,21,15)  COMMON.3
      COMMON/ARRAY3/U(26,22,15),V(26,22,15),C(25,21,15),T(25,21,15)      COMMON.4
      COMMON/ARRAY4/UTEN(26,22,15),VTEN(26,22,15),CTEN(26,22,15),        COMMON.5
     .              TTEN(26,22,15)                                       COMMON.6
      COMMON/ARRAY5/PHI(25,21,15),PHIPIX(25,21,15),PHIPIY(25,21,15)      COMMON.7
      COMMON/ARRAY6/QA(25,21,16),QB(25,21,16),HDIFH(25,21,15)            COMMON.8
      COMMON/ARRAY7/PSA(25,21),PSB(25,21),PTEN(25,21),HM(25,21),         COMMON.9
     .              PUA(26,22),PUB(26,22),PCUT(26,22),CONVAR(26,22)      COMMON.10
      COMMON/ARRAY8/TH(25,21),TG(25,21),STEMP(25,21),CG(25,21),          COMMON.11
     .              GW(25,21),HRATE(25,21),IJCOLM(25,21)                 COMMON.12
      COMMON/ARRAY9/UPRES(30),VPRES(30),CPRES(30),TPRES(30),THETA(30),   COMMON.13
     .              PPRES(30),RELHM(30),UADAT(8),TRDAT(8)                BATFLE.1
      COMMON/RADIAT/RADIN(25,21,16),RADTO(25,21,16),RADSN(25,21,16),     COMMON.15
     .              RADSO(25,21,16)                                      COMMON.16
      COMMON/SCRACH/FACTOR(21,16),FACTOL(21,16),SDOTR(21,16),            COMMON.17
     .              SDOTL(21,16)                                         COMMON.18
      COMMON/MATRIX/FC(16),FI(16),FU(16),FV(16),FT(16),EU(16)            COMMON.19
      COMMON/MISCLN/TWT(16,2),Q(16),A(16),DQ(16),DQI(16),AKI(16),DZ(16), COMMON.20
     .              ALT(16),P(16),SCR(16),DIFM(16),DIFH(16),RI(16),      COMMON.21
     .              DVELDZ(16)                                           COMMON.22
      COMMON/DEBUGS/SAVE1(16),SAVE2(16),SAVE3(16),SAVE4(16),SAVE5(16)    COMMON.23
      COMMON/RADATS/RFLUX(16),FLUXL2(16),T4SIG(16),QEFF(16),             COMMON.24
     .              QEFFG,VEG,TRANS,COSANG,SINANG,PLAT,PLONG,DLONG,SS,   COMMON.25
     .              SA,HHH,FACTOM,COEFF1,COEFF2,COSZEN,DAYBEG,DLAT,      COMMON.26
     .              DAYNOW,AZMUTH,SIGMA,B,HRANGL,TANZEN,SINAZ,COSAZ,     COMMON.27
     .              SOLCON,SOLARN,IRAD,RTOLD,RTNEW,PRECIP,WW,XSAND,      COMMON.28
     .              XCLAY,XORG,XSOIL,GDEPTH,IDEEP,TWATER,GGGG,           COMMON.29
     .              HHHH,ELEF,RRRR,SSSS,NPLT,NPLTT,SURFM                 COMMON.30
      COMMON/SCALAR/DT,DELTAT,ICYCL,SIMTM1,SIMTIM,NCYCLS,IEDIT,IPLOT,    COMMON.31
     .              ITAPE,KMAX,KMAXP,KE,KEM,PTOP,PBOT,UTOP,VTOP,         COMMON.32
     .              CTOP,TTOP,ZBOT,ZNAUT,XLN,IOUT,IFILM,LOUT,NPRES,F,    COMMON.33
     .              CPDIF,RPDIF,ZBL,RHOFAC,GAM,IE,IEND,IEM1,IS,KS,DX,    COMMON.34
     .              RDX,RDXH,RDX4,RDX16,RDXSQ,NOWDAY,ISUNH,ISUNM,ISUNS,  COMMON.35
     .              GMTMHR,GMDYBG,DIF0,DIF1,DY,RDY,RDYH,RDY4,RDY16,      COMMON.36
     .              RDYSQ,JE,JEND,JEM1,JS,JBEG,ALPH,PINV,INPPS,GAM1,     COMMON.37
     .              KBEG,KBGPL,ISTART,NCYCL,LTAPE,CX1,CX2,CX3,CX4        COMMON.38
      COMMON/ADARAY/IDUM                                                 COMMON.39
      LEVEL 2,UA,VA,CA,TA,UB,VB,CB,TB,U,V,C,T,UTEN,VTEN,CTEN,TTEN,PHI,   COMMON.40
     .        PHIPIX,PHIPIY,QA,QB,HDIFH,RADIN,RADTO,RADSN,RADSO          COMMON.41
C                                                                        COMMON.42
C                                                                        UVBOUN.7
C-----COMPUTE PSA CENTERED AT U,V POINTS                                 UVBOUN.8
C     ALSO UPDATE PCUT TO ACCOMODATE CHANGES IN PSA                      UVBOUN.9
      DO 101 I=1,IEND                                                    UVBOUN.10
      II=I                                                               UVBOUN.11
      IM=I-1                                                             UVBOUN.12
      IF(I.EQ.1) IM=1                                                    UVBOUN.13
      IF(I.EQ.IEND) II=IE                                                UVBOUN.14
      DO 101 J=1,JEND                                                    UVBOUN.15
      JJ=J                                                               UVBOUN.16
```

```
            JM=J-1                                                          UVBOUN.17
            IF(J.EQ.1) JM=1                                                 UVBOUN.18
            IF(J.EQ.JEND) JJ=JE                                             UVBOUN.19
            PUB(I,J)=0.25*(PSB(II,JJ)+PSB(IM,JJ)+PSB(II,JM)+PSB(IM,JM))     UVBOUN.20
            PUA(I,J)=0.25*(PSA(II,JJ)+PSA(IM,JJ)+PSA(II,JM)+PSA(IM,JM))     UVBOUN.21
            PCUT(I,J)=PCUT(I,J)+PUA(I,J)-PUB(I,J)                           UVBOUN.22
        101 CONTINUE                                                        UVBOUN.23
                                                                            UVBOUN.24
      C-----WEST AND EAST SIDES LESS CORNERS                                UVBOUN.25
            DO 300 K=1,KMAX                                                 UVBOUN.26
            KS=K                                                            UVBOUN.27
            DO 300 J=JBEG,JE                                                UVBOUN.28
            JS=J                                                            UVBOUN.29
            JM1=J-1                                                         UVBOUN.30
            IF(JE.EQ.1) JM1=1                                               UVBOUN.31
                                                                            UVBOUN.32
            IS=1                                                            UVBOUN.33
            USAVE=UA(1,J,K)                                                 UVBOUN.34
            VSAVE=VA(1,J,K)                                                 UVBOUN.35
            PAV=PUA(1,J)                                                    UVBOUN.36
            PSIG=PAV*A(K)+PTOP                                              UVBOUN.37
            VEL=ENTERP(PSIG,1)                                              UVBOUN.38
            IF(VEL.LE.0.0) GO TO 160                                        UVBOUN.39
            UA(1,J,K)=VEL*PAV                                               UVBOUN.40
            VA(1,J,K)=PAV*VA(2,J,K)/PUA(2,J)                                NEW.249
            GO TO 170                                                       UVBOUN.42
        160 PAV1=PUA(2,J)                                                   UVBOUN.43
            UA(1,J,K)=PAV*UA(2,J,K)/PAV1                                    UVBOUN.44
            VA(1,J,K)=PAV*VA(2,J,K)/PAV1                                    UVBOUN.45
        170 UB(1,J,K)=USAVE+0.5*ALPH*(UB(1,J,K)-2.0*USAVE+UA(1,J,K))        UVBOUN.46
            VB(1,J,K)=VSAVE+0.5*ALPH*(VB(1,J,K)-2.0*VSAVE+VA(1,J,K))        UVBOUN.47
                                                                            UVBOUN.48
            IS=IEND                                                         UVBOUN.49
            USAVE=UA(IEND,J,K)                                              UVBOUN.50
            VSAVE=VA(IEND,J,K)                                              UVBOUN.51
            PAV=PUA(IEND,J)                                                 UVBOUN.52
            PSIG=PAV*A(K)+PTOP                                              UVBOUN.53
            VEL=ENTERP(PSIG,1)                                              UVBOUN.54
            IF(VEL.GE.0.0) GO TO 180                                        UVBOUN.55
            UA(IEND,J,K)=VEL*PAV                                            UVBOUN.56
            VA(IEND,J,K)=PAV*VA(IE,J,K)/PUA(IE,J)                           NEW.250
            GO TO 190                                                       UVBOUN.58
        180 PAV1=PUA(IE,J)                                                  UVBOUN.59
            UA(IEND,J,K)=PAV*UA(IE,J,K)/PAV1                                UVBOUN.60
            VA(IEND,J,K)=PAV*VA(IE,J,K)/PAV1                                UVBOUN.61
        190 UB(IEND,J,K)=USAVE+0.5*ALPH*(UB(IEND,J,K)-2.0*USAVE+UA(IEND,J,K)) UVBOUN.62
            VB(IEND,J,K)=VSAVE+0.5*ALPH*(VB(IEND,J,K)-2.0*VSAVE+VA(IEND,J,K)) UVBOUN.63
        300 CONTINUE                                                        UVBOUN.64
                                                                            UVBOUN.65
      C-----SOUTH AND NORTH SIDES LESS CORNERS                              UVBOUN.66
            IF(JE.EQ.1) GO TO 401                                           UVBOUN.67
            DO 400 K=1,KMAX                                                 UVBOUN.68
            KS=K                                                            UVBOUN.69
            DO 400 I=2,IE                                                   UVBOUN.70
            IS=I                                                            UVBOUN.71
```

```
      IM1=I-1                                                          UVBOUN.72
                                                                       UVBOUN.73
      JS=1                                                             UVBOUN.74
      USAVE=UA(I,1,K)                                                  UVBOUN.75
      VSAVE=VA(I,1,K)                                                  UVBOUN.76
      PAV=PUA(I,1)                                                     UVBOUN.77
      PSIG=PAV*A(K)+PTOP                                               UVBOUN.78
      VEL=ENTERP(PSIG,2)                                               UVBOUN.79
      IF(VEL.LE.0.0) GO TO 260                                         UVBOUN.80
      UA(I,1,K)=PAV*UA(I,2,K)/PUA(I,2)                                 NEW.251
      VA(I,1,K)=VEL*PAV                                                UVBOUN.82
      GO TO 270                                                        UVBOUN.83
  260 PAV1=PUA(I,2)                                                    UVBOUN.84
      UA(I,1,K)=PAV*UA(I,2,K)/PAV1                                     UVBOUN.85
      VA(I,1,K)=PAV*VA(I,2,K)/PAV1                                     UVBOUN.86
  270 UB(I,1,K)=USAVE+0.5*ALPH*(UB(I,1,K)-2.0*USAVE+UA(I,1,K))         UVBOUN.87
      VB(I,1,K)=VSAVE+0.5*ALPH*(VB(I,1,K)-2.0*VSAVE+VA(I,1,K))         UVBOUN.88
                                                                       UVBOUN.89
      JS=JEND                                                          UVBOUN.90
      USAVE=UA(I,JEND,K)                                               UVBOUN.91
      VSAVE=VA(I,JEND,K)                                               UVBOUN.92
      PAV=PUA(I,JEND)                                                  UVBOUN.93
      PSIG=PAV*A(K)+PTOP                                               UVBOUN.94
      VEL=ENTERP(PSIG,2)                                               UVBOUN.95
      IF(VEL.GE.0.0) GO TO 280                                         UVBOUN.96
      UA(I,JEND,K)=PAV*UA(I,JE,K)/PUA(I,JE)                            NEW.252
      VA(I,JEND,K)=VEL*PAV                                             UVBOUN.98
      GO TO 290                                                        UVBOUN.99
  280 PAV1=PUA(I,JE)                                                   UVBOUN.100
      UA(I,JEND,K)=PAV*UA(I,JE,K)/PAV1                                 UVBOUN.101
      VA(I,JEND,K)=PAV*VA(I,JE,K)/PAV1                                 UVBOUN.102
  290 UB(I,JEND,K)=USAVE+0.5*ALPH*(UB(I,JEND,K)-2.0*USAVE+UA(I,JEND,K)) UVBOUN.103
      VB(I,JEND,K)=VSAVE+0.5*ALPH*(VB(I,JEND,K)-2.0*VSAVE+VA(I,JEND,K)) UVBOUN.104
  400 CONTINUE                                                         UVBOUN.105
  401 CONTINUE                                                         UVBOUN.106
                                                                       UVBOUN.107
C-----CORNERS                                                          UVBOUN.108
      DO 500 K=1,KMAX                                                  UVBOUN.109
      KS=K                                                             UVBOUN.110
      IS=1                                                             UVBOUN.111
      JS=1                                                             UVBOUN.112
      JSP1=JS+1                                                        UVBOUN.113
      IF(JE.EQ.1) JSP1=1                                               UVBOUN.114
      USAVE=UA(IS,JS,K)                                                UVBOUN.115
      VSAVE=VA(IS,JS,K)                                                UVBOUN.116
      UA(IS,JS,K)=PUA(IS,JS)*UA(IS+1,JS,K)/PUA(IS+1,JS)                NEW.253
      VA(IS,JS,K)=PUA(IS,JS)*VA(IS,JSP1,K)/PUA(IS,JSP1)                NEW.254
      UB(IS,JS,K)=USAVE+0.5*ALPH*(UB(IS,JS,K)-2.0*USAVE+UA(IS,JS,K))   UVBOUN.124
      VB(IS,JS,K)=VSAVE+0.5*ALPH*(VB(IS,JS,K)-2.0*VSAVE+VA(IS,JS,K))   UVBOUN.125
      IS=IEND                                                          UVBOUN.126
      JS=1                                                             UVBOUN.127
      JSP1=JS+1                                                        UVBOUN.128
      IF(JE.EQ.1) JSP1=1                                               UVBOUN.129
      USAVE=UA(IS,JS,K)                                                UVBOUN.130
      VSAVE=VA(IS,JS,K)                                                UVBOUN.131
```

```
      UA(IS,JS,K)=PUA(IS,JS)*UA(IS-1,JS,K)/PUA(IS-1,JS)               NEW.255
      VA(IS,JS,K)=PUA(IS,JS)*VA(IS,JSP1,K)/PUA(IS,JSP1)               NEW.256
      UB(IS,JS,K)=USAVE+0.5*ALPH*(UB(IS,JS,K)-2.0*USAVE+UA(IS,JS,K))  UVBOUN.139
      VB(IS,JS,K)=VSAVE+0.5*ALPH*(VB(IS,JS,K)-2.0*VSAVE+VA(IS,JS,K))  UVBOUN.140
      IF(JE.EQ.1) GO TO 500                                           UVBOUN.141
      IS=1                                                            UVBOUN.142
      JS=JEND                                                         UVBOUN.143
      USAVE=UA(IS,JS,K)                                               UVBOUN.144
      VSAVE=VA(IS,JS,K)                                               UVBOUN.145
      UA(IS,JS,K)=PUA(IS,JS)*UA(IS+1,JS,K)/PUA(IS+1,JS)               NEW.257
      VA(IS,JS,K)=PUA(IS,JS)*VA(IS,JS-1,K)/PUA(IS,JS-1)               NEW.258
      UB(IS,JS,K)=USAVE+0.5*ALPH*(UB(IS,JS,K)-2.0*USAVE+UA(IS,JS,K))  UVBOUN.153
      VB(IS,JS,K)=VSAVE+0.5*ALPH*(VB(IS,JS,K)-2.0*VSAVE+VA(IS,JS,K))  UVBOUN.154
      IS=IEND                                                         UVBOUN.155
      JS=JEND                                                         UVBOUN.156
      USAVE=UA(IS,JS,K)                                               UVBOUN.157
      VSAVE=VA(IS,JS,K)                                               UVBOUN.158
      UA(IS,JS,K)=PUA(IS,JS)*UA(IS-1,JS,K)/PUA(IS-1,JS)               NEW.259
      VA(IS,JS,K)=PUA(IS,JS)*VA(IS,JS-1,K)/PUA(IS,JS-1)               NEW.260
      UB(IS,JS,K)=USAVE+0.5*ALPH*(UB(IS,JS,K)-2.0*USAVE+UA(IS,JS,K))  UVBOUN.166
      VB(IS,JS,K)=VSAVE+0.5*ALPH*(VB(IS,JS,K)-2.0*VSAVE+VA(IS,JS,K))  UVBOUN.167
  500 CONTINUE                                                        UVBOUN.168
                                                                      UVBOUN.169
      RETURN                                                          UVBOUN.170
      END                                                             UVBOUN.171
```

```
      SUBROUTINE VELVEC(U,V,H,IMAX,JMAX,IM1,JM1,KMX,KPL,DX,DY,ICYCL)         VELVEC.2
                                                                             VELVEC.3
      COMMON/IGSZZZ/Z(200)                                                   COMMPL.2
      COMMON/TRANSF/ITRX,ITRY,TRN(26,22)                                     COMMPL.3
      DIMENSION U(IMAX,JMAX,KMX),V(IMAX,JMAX,KMX),H(IM1,JM1)                 VELVEC.5
      LEVEL 2,U,V                                                            VELVEC.6
      DIMENSION CCL(40)                                                      VELVEC.7
      DATA IONCE/0/                                                          VELVEC.8
                                                                             VELVEC.9
                                                                             VELVEC.10
C-----SET UP TO FIT ON 8.5X11 PAPER                                          VELVEC.11
      IF(IONCE.EQ.1) GO TO 100                                               VELVEC.12
      XDIM=0.9                                                               VELVEC.13
      YDIM=0.65                                                              VELVEC.14
      TOT=1.0                                                                VELVEC.15
      ASPECT=XDIM/YDIM                                                       VELVEC.16
      XX=(IMAX-1)*DX                                                         VELVEC.17
      YY=(JMAX-1)*DY                                                         VELVEC.18
      XYA=XX/YY                                                              VELVEC.19
      IF(XYA.LT.ASPECT) GO TO 101                                            VELVEC.20
      XIN=XDIM                                                               VELVEC.21
      YIN=XIN/XYA                                                            VELVEC.22
      GO TO 102                                                              VELVEC.23
  101 YIN=YDIM                                                               VELVEC.24
      XIN=YIN*XYA                                                            VELVEC.25
  102 XORIG=(TOT-XIN)/2.0                                                    VELVEC.26
      YORIG=(TOT-YIN)/2.0                                                    VELVEC.27
      DELX=XIN/FLOAT(IMAX-1)                                                 VELVEC.28
      DELY=YIN/FLOAT(JMAX-1)                                                 VELVEC.29
      AA=0.2                                                                 VELVEC.30
      COSA=1.0/COS(AA)                                                       VELVEC.31
C-----COMPUTE TERRAIN HEIGHTS CENTERED AT U,V POINTS                         VELVEC.32
      DO 105 I=1,IMAX                                                        VELVEC.33
      II=I                                                                   VELVEC.34
      IM=I-1                                                                 VELVEC.35
      IF(I.EQ.1) IM=1                                                        VELVEC.36
      IF(I.EQ.IMAX) II=IMAX-1                                                VELVEC.37
      DO 105 J=1,JMAX                                                        VELVEC.38
      JJ=J                                                                   VELVEC.39
      JM=J-1                                                                 VELVEC.40
      IF(J.EQ.1) JM=1                                                        VELVEC.41
      IF(J.EQ.JMAX) JJ=JMAX-1                                                VELVEC.42
      TRN(I,J)=0.25*(H(II,JJ)+H(IM,JJ)+H(II,JM)+H(IM,JM))                    VELVEC.43
  105 CONTINUE                                                               VELVEC.44
      IONCE=1                                                                VELVEC.45
  100 CONTINUE                                                               VELVEC.46
                                                                             VELVEC.47
C-----OBJECT AND SUBJECT SPACE                                               VELVEC.48
      CALL SETSMG(Z,19,1.0)                                                  VELVEC.49
      CALL SETSMG(Z,20,1.0)                                                  VELVEC.50
      CALL OBJCTG(Z,0.0,0.0,1.0,1.0)                                         VELVEC.51
      CALL SUBJEG(Z,0.0,0.0,1.0,1.0)                                         VELVEC.52
                                                                             VELVEC.53
      DO 300 K=KPL,KMX                                                       VELVEC.54
                                                                             VELVEC.55
```

```
C-----ESTABLISH VECTOR SCALE
      UMAG=0.0
      UMAX=U(1,1,K)
      SCALE=XIN/FLOAT(IMAX+3)
      DO 150 J=1,JMAX
      DO 150 I=1,IMAX
      UMAX=AMAX1(ABS(U(I,J,K)),UMAX)
      UMAX=AMAX1(ABS(V(I,J,K)),UMAX)
      UMAG=AMAX1(UMAG,SQRT(U(I,J,K)**2+V(I,J,K)**2))
  150 CONTINUE
      IF(UMAX.LE.0.0) UMAX=1.0
      SCALE=SCALE/UMAX
      WRITE (6,660) K,UMAG,ICYCL
  660 FORMAT(3H K=,I3,5X,5HUMAG=,1PE12.4,5X,6HICYCL=,I5)

C-----DRAW BOUNDARIES
      CALL SETSMG(Z,30,2.0)
      CALL BOXG(Z,0,0,XORIG,YORIG,XORIG+XIN,YORIG,XORIG,YORIG+YIN,
     .          XORIG+XIN,YORIG+YIN)
      CALL SETSMG(Z,30,1.0)

      YNOW=-DELY+YORIG
      DO 200 J=1,JMAX
      YNOW=YNOW+DELY
      XNOW=-DELX+XORIG
      DO 200 I=1,IMAX
      XNOW=XNOW+DELX

C-----MOVE TO NEW DATA POINT
      CALL LINESG(Z,0,XNOW,YNOW)

C-----PLOT THE VECTOR
      XX=U(I,J,K)*SCALE
      YY=V(I,J,K)*SCALE
      XEND=XNOW+XX
      YEND=YNOW+YY
      CALL LINESG(Z,1,XEND,YEND)

C-----ADD THE ARROW
      R=0.8*SQRT(XX*XX+YY*YY)*COSA
      BB=0.5*3.1415926
      IF(XX.NE.0.0) BB=ATAN(ABS(YY/XX))
      XP=R*COS(AA+BB)
      YP=R*SIN(AA+BB)
      IF(U(I,J,K).LT.0.0) XP=-XP
      IF(V(I,J,K).LT.0.0) YP=-YP
      XEND=XNOW+XP
      YEND=YNOW+YP
      CALL LINESG(Z,1,XEND,YEND)
  200 CONTINUE

C-----SUPERIMPOSE TERRAIN CONTOURS ON THE VECTOR PLOTS
      NCL=9
      QMAX=-1.E+8
      QMIN= 1.E+8
```

```
      DO 920 I=1,IMAX                                                   VELVEC.111
      DO 920 J=1,JMAX                                                   VELVEC.112
      QMAX=AMAX1(QMAX,TRN(I,J))                                         VELVEC.113
      QMIN=AMIN1(QMIN,TRN(I,J))                                         VELVEC.114
  920 CONTINUE                                                          VELVEC.115
      DQQ=(QMAX-QMIN)/FLOAT(NCL+1)                                      VELVEC.116
      DO 930 I=1,NCL                                                    VELVEC.117
      CCL(I)=QMIN+I*DQQ                                                 VELVEC.118
  930 CONTINUE                                                          VELVEC.119
      CALL SET(XORIG,XORIG+XIN,YORIG,YORIG+YIN,1.0,FLOAT(IMAX),1.0,     VELVEC.120
     .         FLOAT(JMAX),1)                                           VELVEC.121
      CALL SETSMG(Z,30,0.3)                                             VELVEC.122
      ITRX=0                                                            VELVEC.123
      ITRY=0                                                            VELVEC.124
      CALL CONREC(TRN,IMAX,IMAX,JMAX,CCL,NCL,1,0,1736B,0)               VELVEC.125
      CALL SETSMG(Z,30,1.0)                                             VELVEC.126
      CALL OBJCTG(Z,0.0,0.C,1.0,1.0)                                    VELVEC.127
      CALL SUBJEG(Z,0.0,0.C,1.0,1.0)                                    VELVEC.128
                                                                        VELVEC.129
C-----EJECT PAGE                                                        VELVEC.130
      CALL PAGEG(Z,0,1,1)                                               VELVEC.131
  300 CONTINUE                                                          VELVEC.132
                                                                        VELVEC.133
      RETURN                                                            VELVEC.134
      END                                                               VELVEC.135
```


*U.S. GOVERNMENT PRINTING OFFICE: 1980-740-145/986