User's Manual for the Convective Cloud Module Version 1.0

B. C. Scott

April 1987

**Prepared for
the U.S. Environmental Protection Agency
under a Related Services Agreement
with the U.S. Department of Energy
Contract DE-AC06-76RLO 1830**

**Pacific Northwest Laboratory
Operated for the U.S. Department of Energy
by Battelle Memorial Institute**

EPA NOTICE

This report is based on work supported by the U.S. Environmental Protection Agency under a Related Services Agreement with the U.S. Department of Energy, Contract DE-AC06-76RLO 1830, Interagency Agreement EPA-DW89930059. Although the research described in this report has been funded wholly or in part by the U.S. Environmental Protection Agency (EPA), it has not been subject to EPA review and therefore does not necessarily reflect the views of EPA and no official endorsement should be inferred.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government of any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof, or Battelle Memorial Institute.

PACIFIC NORTHWEST LABORATORY
operated by
BATTELLE MEMORIAL INSTITUTE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC06-76RLO 1830

Printed in the United States of America
Available from
National Technical Information Service
United States Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22161

NTIS Price Codes
Microfiche A01

Printed Copy

Pages	Price Codes
001-025	A02
026-050	A03
051-075	A04
076-100	A05
101-125	A06
126-150	A07
151-175	A08
176-200	A09
201-225	A010
226-250	A011
251-275	A012
276-300	A013

3 3679 00046 5296

USER'S MANUAL FOR THE CONVECTIVE
CLOUD MODULE VERSION 1.0

B. C. Scott

April 1987

Prepared for
the U.S. Environmental Protection Agency
under a Related Services Agreement
with the U.S. Department of Energy
Contract DE-AC06-76RLO 1830

Pacific Northwest Laboratory
Richland, Washington 99352

SUMMARY

This manual describes a convective cloud model and provides guidance for users. The convective cloud model is assumed to provide a time-averaged distribution of data from a population of precipitating, convective clouds in different stages of development. This model was designed to characterize a scavenging environment using meteorological conditions provided by a regional-scale meteorological code. This meteorological code explicitly simulates processes related to cloud microphysics.

The cloud model predicts the vertical profiles of condensed water that correspond to specified surface precipitation rates and cloud top and freezing level heights. The cloud model also predicts profiles of various microphysical constituents, such as fall velocity and accretion rates that exist in conjunction with the condensed water profiles.

Descriptions of routines written to solve mass conservation equations for air, cloud, and precipitation water are presented. The solutions are appropriate for precipitating convective clouds with a horizontal spacing on the order of 10 km. Routines are also provided for generating tables of profiles for many discrete input conditions. Once tables are generated, additional routines may be used to interpolate between the tables and to rapidly determine values at levels other than the discrete input levels.

The model was developed with funding provided by the Environmental Protection Agency and the National Acid Precipitation Assessment Program. The immediate application of the work is to provide a means of defining the cloud environment simulated within the scavenging module of the Regional Acid Deposition Model (RADM) Scavenging Module (RSM). Within the RSM, the convective cloud model provides data on a steady-state scavenging environment in which a time-dependent, aqueous-phase, chemical kinetic system will operate.

CONTENTS

SUMMARY	iii
1.0 INTRODUCTION	1
1.1 INTERRELATIONSHIPS BETWEEN THE CLOUD MODEL, RADM, AND RSM	1
1.2 THE RSM CLOUD MODEL	3
2.0 GOVERNING EQUATIONS	5
2.1 DYNAMICAL EQUATIONS	5
2.1.1 Air Motions	7
2.1.2 Temperature-Height Relationships	12
2.2 MICROPHYSICAL EQUATIONS	12
2.2.1 Condensation Rate	12
2.2.2 Depositional Growth	12
2.2.3 Ice Crystal Concentration	14
2.2.4 Evaporation	15
2.2.5 Precipitation Fall Speeds	15
2.2.6 Transfer of Precipitation from the Updraft Cell to the Rainshaft Cell	17
2.2.7 Pseudodiffusion	18
2.2.8 Scavenging (Accretion) Coefficients	19
2.2.9 Autoconversion	19
3.0 FINITE DIFFERENCES EXPRESSIONS	21
4.0 CLOUD-GENERATION ROUTINES	23
4.1 PROGRAM WETUP	23
4.2 SUBROUTINE MAINUP	26
4.3 SUBROUTINE WATER3	29
4.4 SUBROUTINE WATER4	31

4.5	SUBROUTINE WINDUP	33
4.6	SUBROUTINE SIGMA	34
4.7	SUBROUTINE SIGMAD	34
4.8	SUBROUTINE PRESSF	35
4.9	SUBROUTINE PRESSFD	35
4.10	SUBROUTINE COEF	35
4.11	SUBROUTINE VDEP	35
4.12	SUBROUTINE NUKE	35
4.13	SUBROUTINE SEARCH	36
4.14	SUBROUTINE DEFALT	36
4.15	SUBROUTINE NEUTON	36
4.16	SUBROUTINE ZERO1	37
4.17	SUBROUTINE UNITS	37
4.18	SUBROUTINE FLUX	38
4.19	SUBROUTINE CLOUD_BASE	38
4.20	SUBROUTINE FALOUT	38
4.21	SUBROUTINE FREZLVL	39
5.0	EXECUTION OF THE CLOUD MODEL ROUTINES	41
5.1	DATA INPUT	41
5.2	DATA OUTPUT	45
5.3	EXECUTION SPEED	46
6.0	DIAGNOSTIC MESSAGES FROM THE CLOUD-GENERATION ROUTINES	47
6.1	DIAGNOSTIC MESSAGES FROM WATER3	47
6.2	DIAGNOSTIC MESSAGES FROM WATER4	47
6.3	DIAGNOSTIC MESSAGES FROM WETUP	48
6.4	DIAGNOSTIC MESSAGES FROM MAINUP	49

7.0	INTERPOLATION ROUTINES	51
7.1	PROGRAM TABLEH	51
7.2	SUBROUTINE HEIGHT	52
7.3	PROGRAM BEGIN	53
7.4	SUBROUTINE SELECT	53
7.5	SUBROUTINE INTERP	55
8.0	EXECUTION OF THE INTERPOLATION PROGRAMS	57
8.1	EXECUTION REGARDLESS OF STARTING POINT	57
8.2	MERGING OF THE TABLES	59
8.3	EXECUTION USING MERGED TABLES	59
8.4	EXECUTION SPEED	60
9.0	DIAGNOSTIC MESSAGES FROM THE INTERPOLATION ROUTINES	61
	DIAGNOSTICS FROM INTERP	61
10.0	LIST OF MAJOR ASSUMPTIONS	63
11.0	REFERENCES	67
	APPENDIX A - SAMPLE OUTPUT	A.1
	APPENDIX B - PROGRAM LISTINGS	B.1

1.0 INTRODUCTION

1.1 INTERRELATIONSHIPS BETWEEN THE CLOUD MODEL, RADM, AND RSM

This manual describes in detail one of three major components to Version 1.0 of the Regional Acid Deposition Model (RADM) Scavenging Module (RSM). The RADM is currently being developed at the National Center for Atmospheric Research (NCAR), funded by the Environmental Protection Agency, and coordinated by the National Acid Precipitation Assessment Program Task force.

The RADM is a clear-air, regional air-pollution code that does not explicitly account for cloud-related processes. The model described in this report has been designed to provide data on a steady-state, convective scavenging environment in which a time-dependent, aqueous-phase, chemical kinetic system will operate. Similar models describing a stratiform storm environment and a fair-weather cumulus scavenging environment are also being used in the RSM.^(a) Output from these cloud models will be used by the RSM to quantitatively describe cloud-related processes in the RADM. Berkowitz and Easter (1986) have documented the structure and theory of the RSM.

In the absence of the RSM, the RADM is simply an advanced clear-air, regional air-pollution code that operates in an Eulerian frame of reference over an extensive 80- by 80-km mesh grid. In contrast, the RSM deals with all processes related to clouds and precipitation (Figure 1). The interaction between the two modules is illustrated by the following steps:

1. The RADM computes transport and clear-air chemistry for a 1-h period. The synoptic scale environment in which these processes occur is defined by data provided by the Meso-alpha Scale (Regional) Limited-Area Model developed by Anthes (1985).

(a) B. C. Scott of the Pacific Northwest Laboratory, Richland, Washington, and F. Vukovich of the Research Triangle Institute, Research Triangle Park, North Carolina, are currently preparing user's manuals for these components of the RSM.

FIGURE 1. Illustration of calling hierarchy between the convective scavenging model (described in this report), the stratiform model, the fair weather cumulus model, the Regional Acid Deposition Model (RADM), the mesoscale meteorology model (MM4), and the RADM Scavenging Module (RSM) (Berkowitz and Easter 1986).

2. The RSM is activated at each grid column. A grid column is the vertical column over a particular RADM surface grid point. In activating the RSM, the RADM/MM4 system provides
 - vertical profiles of temperature, moisture, and updraft velocities
 - precipitation amounts (convective and nonconvective) for the past hour
 - vertical profiles of clear-air pollutant species (gases only)
3. Fields of clouds and precipitation consistent with these profiles are established by the RSM. It is the primary role of the cloud model defined in this report to provide a detailed description of the cloud microphysical environment that would be associated with the

meteorological conditions specified by MM4 for a particular RADM grid column at a fixed time. Using the environmental characteristics so defined, the RSM calculates effects of 1 hour's precipitation scavenging, aqueous chemical transformations, and vertical transport by subgrid (convective) motions.

4. At the end of an hour, pollutants in the RSM exist as gases and aerosols. Pollutants also exist in cloud, rain, and snow media. The RSM combines the vertical profiles of these different forms of each pollutant to produce an equivalent clear-air profile. These profiles are returned to the RADM along with surface wet-deposition amounts for pollutants that have been scavenged.
5. The effects of clouds and precipitation for the hour are represented by the change between the initial profiles passed from the RADM to the RSM and the equivalent clear-air profiles passed back. The RADM combines this change with those changes computed in Step 1, then proceeds to the next 1-hour simulation (Steps 2 through 5).

1.2 THE RSM CLOUD MODEL

The primary purpose of the cloud model is to predict the vertical profiles of condensed water that correspond to specified surface precipitation rates, surface temperatures, and cloud top heights. A secondary purpose is to predict profiles of various microphysical constituents, such as fall velocity and accretion rates, that exist in conjunction with the condensed water profiles.

Precipitating convective cells are simulated in adjacent updraft and rainshaft columns. The horizontal dimensions of each column are taken to be 10 km. In the updraft column, cloud and precipitation water are assumed to move upward or to be suspended by the updraft. When the water concentrations in the updraft portion can no longer be supported by the updraft, excess water is transferred to the rainshaft cell. The precipitation in the rainshaft column then falls out through a saturated downdraft. Both columns are assumed to be in a steady state. Computer-generated data from the calculations are segregated according to the updraft or downdraft portions and placed in mass storage where they can be easily accessed by a lookup routine.

In the cloud model, only two condensed water categories are used: cloud water and precipitation water. Cloud water is assumed to move with the air velocity. Precipitation water moves horizontally with the air but falls vertically with respect to the air. The fall speeds and collection rates of precipitation depend on the classification of precipitation as frozen or liquid, which, in turn, depends upon the local air temperature.

Precipitation embryos are assumed to be ice particles that grow by vapor deposition and accretion as the ice particles fall. Below the freezing level, the ice particles melt, accelerate, continue to accrete cloud droplets, and fall to the surface as millimeter-sized rain drops.

The model equations have been coded in FORTRAN, and the entire model has been successfully executed on a VAX 11/780 computer using a VMS operating system. The cloud model routines execute relatively slowly. To allow information to be rapidly accessed for many discrete input conditions, routines are provided for generating tables of profiles for a variety of initial conditions. Once the tables have been generated and saved, a user can invoke routines designed to interpolate within the tables. Thus, values for conditions different from the discrete input levels can be rapidly obtained. This table lookup and interpolation procedure is about 100 times faster than generating profiles for each discrete set of input data.

This document describes the basic conservation equations for condensed water and the function of each routine in the model. It also provides a list of the assumptions used in the model and the procedures for executing the model and interpolation routines.

2.0 GOVERNING EQUATIONS

This section presents a brief discussion of the dynamical and microphysical concepts used to develop the cloud module. The subroutines where a particular concept is executed are also identified, but specific details of coding and subroutine objectives are deferred to Section 4.0.

2.1 DYNAMICAL EQUATIONS

Within the model domain, the water concentrations are determined by a balance between the liquid water produced in a region, the water passing in or out through the boundaries of the region, and the water removed from the region after capture by falling precipitation. Specifically, by defining M_c as the mass of water per unit volume of space associated with cloud water and M_p as that associated with precipitation water, then the local change of each within a precipitating cloud is given by

$$\frac{\partial M_c}{\partial t} = - \vec{V} \cdot M_c \vec{V} + D_c - C_p - A_p + S_c \quad (2.1)$$

$$\frac{\partial M_p}{\partial t} = - \vec{V} \cdot M_p \vec{V} + \frac{\partial [(V_p - V_d)M_p]}{\partial z} + D_p + C_p + A_p + S_p \quad (2.2)$$

In the above,

$\vec{V}$ = the wind velocity vector

V_p = the fall speed of precipitation (taken as positive)

V_d = the diffusion velocity for precipitation

D = the diffusion of moisture from the air to cloud or precipitation water

C_p = the collection of cloud water by precipitation

A_p = the rate at which cloud water is converted to precipitation water through an autoconversion process

S = the sinks or sources resulting from transfers of water between the adjacent updraft and rainshaft cells.

In the convective cloud model, two cells (an updraft cell and a rainshaft cell) are simulated to exist side by side. Equations (2.1) and (2.2) apply separately to each cell. To form a mass budget equation for the condensed water within each region, several assumptions are made. First, the precipitation and cloud water concentrations are assumed to be horizontally uniform. Next, at the external, lateral, inflow boundaries of each cell, the cloud and precipitation water concentrations are assumed to be zero. Finally, the storm is assumed to be in a quasi-steady state.

Mathematically, these assumptions enable Equations (2.1) and (2.2) to be expressed as

$$- Gx \bar{M}_c + (M_c w)|_z - (M_c w)|_{z+\Delta z} + (\bar{D}_c - \bar{C}_p - \bar{A}_p + \bar{S}_c)\Delta z = 0 \quad (2.3)$$

and

$$\begin{aligned} - Gx \bar{M}_p + [M_p(V_p - V_d - w)]|_{z+\Delta z} - [M_p(V_p - V_d - w)]|_z \\ + (\bar{D}_p + \bar{C}_p + \bar{A}_p + \bar{S}_p)\Delta z = 0 \end{aligned} \quad (2.4)$$

where

$$Gx = \delta_u \bar{u}|_{x+\Delta x} - \delta_u \bar{u}|_x + \delta_v \bar{v}|_{y+\Delta y} - \delta_v \bar{v}|_y \quad (2.5)$$

A single overbar indicates the quantity averaged over the interval z to $z + \Delta z$, and a double overbar indicates the quantity averaged over the volume $\Delta x, \Delta y, \Delta z$.

Equation (2.3) is linear and can be solved for M_c recursively, given vertical profiles of M_p and w . However, Equation (2.4) is nonlinear due to the strong dependence of V_p and C_p on M_p , and therefore must be solved iteratively for M_p .

Equations (2.3) and (2.4), are the basic dynamic equations of the model. They are coupled in the sense that one cannot be solved without knowledge of the values obtained from the other. The solution sequence for these two equations is carried out in subroutines WATER3 and WATER4.

2.1.1 Air Motions

In Equations (2.3) and (2.4), the quantity G_x represents the mass blown out through the sides of a cell by the horizontal winds. The variables u and v are the horizontal wind components, and the values δ_u and δ_v are given by

$$\begin{aligned}\delta_u &= \Delta z / \Delta x, \text{ if airflow at an external x-boundary is out of the storm} \\ &= 0 \text{ otherwise at an external x-boundary,} \\ \delta_v &= \Delta z / \Delta y, \text{ if airflow at an external y-boundary is out of the storm} \\ &= 0 \text{ otherwise at an external y-boundary.}\end{aligned}\tag{2.6}$$

At the internal boundary between the updraft and rainshaft cells, precipitation water that cannot be supported by the updraft velocity in the updraft cell is transferred to the rainshaft cell. This point is discussed in more detail in Section 2.2.6.

Horizontal transfer between the two cells also occurs at the internal boundary. If all of the cloud water being transported from the updraft cell is placed into the appropriate rainshaft cloud-water bins, profiles of cloud water near the top of the rainshaft may be unrealistic. Physically, what occurs is that large amounts of cloud water pass into the rainshaft near the top where the updraft speed is near zero. However, cloud water transported from the updraft cell is likely to have significant fall speeds because it has been suspended in a large updraft. Therefore, half of the transported cloud water is categorized as rainshaft precipitation and half as rainshaft cloud water. This redistribution prevents the occurrence of large cloud water spikes near the cloud top in the rainshaft cell, and it is probably more physically realistic than a straight transfer between cloud-water bins in the adjacent cells would be.

The horizontal average of the horizontal wind fields is necessary for determining the interaction of the cloud cells with each other and with the environment. The average is given by

$$\text{HORZ} = -1/A \iint \vec{V} \cdot \vec{V}_h C dx dy = (C_i F_{io} - F_{oi} C_o) / A \quad (2.7)$$

where A = the area of the updraft or downdraft cell

$\vec{V}_h$ = the horizontal wind vector

C = the concentration of the constituent of interest

F = an outflow or inflow velocity times the dimension of the side
where the outflow or inflow is occurring.

The subscripts i and o mean inner and outer regions, respectively, while subscripts io and oi mean transport from inner to outer or outer to inner regions, respectively. The variable F is determined in subroutine FLUX.

Concentrations obtained from Equations (2.3) and (2.4) are dependent upon the updraft velocity. The updraft velocity in the updraft cell has been selected to provide accumulation zones in the upper portion of the cell. The updraft increases linearly from the ground to some height, $z(\text{max})$, and then decreases linearly from $z(\text{max})$ to the cloud top. The exact form used for calculating updraft velocity, w , is

$$w = \frac{W_{\text{MAX}} z}{RH} \quad (2.8)$$

where z is less than or equal to $z(\text{max})/H$, and

$$w = W_{\text{MAX}} \left[1 - \frac{(z - RH)}{H(1-R)} \right] \quad (2.9)$$

where z is greater than $z(\text{max})/H$. Here,

H = the cloud top height

W_{MAX} = the maximum updraft velocity

$R = z(\text{max})/H$.

The value of R has been set equal to 0.875. W_{MAX} is varied by subroutine WINDUP to adjust the storm efficiency, E , defined as

$$E = \frac{\text{surface precipitation flux}}{\text{inflow vapor flux}} \quad (2.10)$$

For each set of input conditions, the storm efficiency is initially taken to have the value set by the user in program WETUP. If, however, the maximum updraft speed is computed to be less than 2 m/s in the updraft cell, the storm efficiency is decreased in 0.1 increments until the 2 m/s updraft is achieved.

Other changes in storm efficiency occasionally result from microphysical constraints that will be discussed in Section 2.2.5. In these instances, the storm efficiency is increased to reduce the velocity in the updraft-cell.

The model is designed to compute environmental conditions associated with hourly precipitation amounts, not precipitation rates. Therefore, when updraft-cell velocities are quite large and large inflow vapor fluxes are induced, the desired storm efficiency can still be achieved by reducing the storm duration and increasing the surface precipitation rate. The storm duration must, however, be greater than 20 minutes. If Equation (2.10) can be satisfied only with a storm duration of less than 20 minutes, then the storm efficiency is increased in 0.1 increments until the duration exceeds the 20-minute limit. The actual storm duration and surface precipitation rate are determined in subroutine MAINUP.

An inverted S-shaped profile has been selected to describe the vertical velocity in the rainshaft cell. The profile is parabolic with upward motions above $H/2$ and downward motions below $H/2$, where H is cloud top height. This particular profile is a crude approximation to time-averaged vertical-velocity profiles obtained from more sophisticated, time-dependent models. Mathematically, for the rainshaft cell,

$$w = -12W \left[z - \left(2z^2/H \right) \right] / H \quad (2.11)$$

for z less than $H/2$, and

$$w = 12W \left[3z - H - \left(2z^2/H \right) \right] / H \quad (2.12)$$

for z greater than or equal to $H/2$. Here W is the magnitude of the mean updraft profile obtained by vertically averaging from the ground to $H/2$. The user can easily choose different vertical wind profiles by modifying the statements in subroutine WINDUP.

For some input conditions, it is not possible to obtain solutions with rainshaft profiles described by Equations (2.11) and (2.12). Should this happen, subsidence is assumed to occur in the entire rainshaft. The profile is taken to be parabolic from the surface to the cloud top, and:

$$w = -6W \left(z - z^2/H \right) / H \quad (2.13)$$

Also important in determining concentrations of condensed water is the magnitude of the horizontal wind components. All three wind components are coupled through the anelastic continuity equation

$$\vec{\nabla} \cdot \rho_a \vec{V} = 0 \quad (2.14)$$

where ρ is the air density, assumed to be a function of height alone. Integrating Equation (2.14) over a cell volume of $\Delta x, \Delta y, \Delta z$ and letting $\Delta x = \Delta y$, yields a continuity equation

$$\frac{\Delta z}{\Delta x} \bar{\rho} \left[\bar{u}|_{x+\Delta x} - \bar{u}|_x + \bar{v}|_{y+\Delta y} - \bar{v}|_y \right] + (\rho w)|_{z+\Delta z} - (\rho w)|_z = 0 \quad (2.15)$$

where the overbar implies a vertical average over the thickness of the cell.

The solution of Equation (2.15) is obtained in subroutine WINDUP by using the updraft profile together with several assumptions about the vertical variation of the inflow u and v components. First, it is assumed that

$$\frac{\partial v}{\partial y} = \frac{\partial u}{\partial x} \quad (2.16)$$

Next, the u and v wind components are specified at two boundaries for each cell. For the updraft cell, when z is less than $z(\text{max})$,

$$\begin{aligned}\bar{u}(x + \Delta x) &= 0 \\ \bar{v}(y) &= \bar{u}(x)/2\end{aligned}\tag{2.17}$$

When z is greater than $z(\text{max})$ in the updraft cell,

$$\begin{aligned}\bar{u}(x) &= 0 \\ \bar{v}(y) &= -\bar{u}(x + \Delta x)/2\end{aligned}\tag{2.18}$$

For the rainshaft cell, $u(x + \Delta x)$ is taken to equal the negative of the value of the $u(x + \Delta x)$ of the updraft cell, and

$$\bar{v}(y) = \bar{u}(x)/2\tag{2.19}$$

With the above restrictions for the two u and v components, the remaining u and v components are required by Equation (2.15) to equal

$$\begin{aligned}\bar{u}(x) &= \bar{u}(x + \Delta x) + \frac{\left[(\rho_w)_{z+\Delta z} - (\rho_w)_z \right] (\Delta x / \Delta z)}{\rho_{z+\Delta z} + \rho_z} \\ \bar{v}(y + \Delta y) &= \bar{v}(y) + [\bar{u}(x + \Delta x) - \bar{u}(x)]\end{aligned}\tag{2.20}$$

The sign convention is that for airflow into a cell, $u(x)$ and $v(y)$ are positive, while $u(x + \Delta x)$ and $v(y + \Delta y)$ are negative.

2.1.2 Temperature-Height Relationships

The cloud top temperature is taken to be the same for both the rainshaft and the updraft cells, and the temperature lapse rate is assumed to be moist adiabatic within the cloud and dry adiabatic below the cloud base. The cloud base height is taken to be the surface for the rainshaft cell and the lifting condensation level for the updraft cell.

2.2 MICROPHYSICAL EQUATIONS

The microphysical concepts and formulations used in the model are presented in this subsection. As was the case for the dynamical equations, the subroutines where a particular microphysical concept is executed are named here. However, specific details of coding and subroutine objectives are deferred to Section 4.0.

2.2.1 Condensation Rate

The total condensation rate, $\bar{D}_T$, is computed from

$$\bar{D}_T = -w p_a \frac{\partial r_s}{\partial z} \quad (2.21)$$

Both air density, p_a , and the saturation mixing ratio, r_s , are determined from thermodynamic variables computed along a moist adiabat in subroutines SIGMA and SIGMAD. The total condensation rate, $\bar{D}_T$, is computed for both water and ice and the results are stored in the arrays SORCE (for water) and SORCEI (for ice).

2.2.2 Depositional Growth

Depositional growth of precipitation is considered only if the updraft velocity is positive and only for temperatures less than 0°C. Condensation onto liquid precipitation is neglected. The depositional growth rate for ice, $\bar{D}_p$, is computed by considering the growth, by vapor deposition, of individual ice crystals in a snowflake (Scott 1982)

$$\frac{dx}{dt} = 31.5gS_i x^{0.38} \quad (2.22)$$

where

x = the mass of an ice crystal

g = a thermodynamic ice growth function from Fletcher (1969)

S_i = the cloud supersaturation with respect to ice

Integrating Equation (2.22) over the time interval Δt and assuming that the number of ice crystals per unit volume, N , remains constant during Δt , yields

$$M_d = \left(M_{do}^{0.62} + 19.53 g \bar{S}_i N^{0.62} \Delta t \right)^{1.61} \quad (2.23)$$

where

$$M_d = Nx$$

M_{do} = the flake mass concentration at the beginning of the interval Δt and the overbars represent average values during Δt . Note that where precipitation enters a cell, its mass is given by M_{do} .

The depositional growth rate, $\bar{\bar{D}}_p$, can then be approximated from

$$\bar{\bar{D}}_p = \frac{M_d - M_{do}}{\Delta t} \approx \frac{\delta M_d}{\delta t} \quad (2.24)$$

or

$$\bar{\bar{D}}_p = \left\{ \left[M_{do}^{0.62} + \frac{19.53 g \bar{S}_i N^{0.62} \Delta z}{(\bar{V}_p - \bar{w})} \right]^{1.61} - M_{do} \right\} \frac{\bar{V}_p - \bar{w}}{\Delta z} \quad (2.25)$$

where Δt has been replaced by $\Delta z/(\bar{V}_p - \bar{w})$, and all the quantities with overbars are evaluated in the interval Δz .

The function gS_i is curve-fit to give

$$\begin{aligned}
gS_i &= 0.98 \times 10^{-8} (-T)^{0.5} & \text{if } T > -10^\circ\text{C} \\
&= 3.2 \times 10^{-8} & \text{if } -10^\circ\text{C} > T > -20^\circ\text{C} \\
&= (5.4 + 0.11T) \times 10^{-8} & \text{if } -20^\circ\text{C} > T > -40^\circ\text{C} \\
&= 1 \times 10^{-8} & \text{if } -40^\circ\text{C} > T
\end{aligned} \tag{2.26}$$

where T is the temperature.

The depositional growth rate determined from Equation (2.25) can often exceed the permissible growth rate computed in Equation (2.21), especially when ice crystal concentrations are large. Should such excessively large growth rates occur, the depositional growth rate is limited to the total condensational growth rate, $\bar{D}_T$ (computed with respect to ice). In all instances, the depositional growth rate is subtracted from the total condensation rate. If any excess results, the remaining condensed water contributes to condensed cloud water $\bar{D}_C$; i.e.,

$$\begin{aligned}
\bar{D}_C &= \bar{D}_T - \bar{D}_p & \text{for } \bar{D}_p < \bar{D}_T \\
\bar{D}_C &= 0 & \text{for } \bar{D}_p > \bar{D}_T
\end{aligned} \tag{2.27}$$

If $\bar{D}_p$ should exceed $\bar{D}_T$, then $\bar{D}_p$ is set equal to $\bar{D}_T$.

2.2.3 Ice Crystal Concentration

The ice crystal concentration, N , is determined from

$$N = 400 (-T/16)^4 \quad \text{m}^{-3} \tag{2.28}$$

where T is the temperature at height z . In the rainshaft cell, N is computed using the cloud top temperature. For the updraft cell, N is computed using the temperature at each height. To simulate the aggregation of ice crystals and to

permit the onset of riming as soon as possible (i.e., to limit cloud glaciation), the total ice crystal concentration is not allowed to exceed 30 crystals/liter.

2.2.4 Evaporation

In the updraft cell, the rising air is always saturated, and no precipitation falls below cloud base. Evaporation is not considered. In the rainshaft cell, the cloud base is at the ground and "below-cloud" evaporation does not occur. However, evaporation occurs continuously in the saturated downdraft. Because no cloud water is available in the saturated downdraft, this evaporation serves to reduce only the precipitation concentration.

2.2.5 Precipitation Fall Speeds

To a large extent, the properties of the updraft cell near the top of the cloud determine how the precipitation fall speeds are computed. The vertical velocity profile in the updraft cell is designed to produce float or accumulation levels in the upper portion of the cell. To do so, there must be a level where the median value of the precipitation fall speeds is equal to the updraft speed. The concentration of the accumulated water must be large (i.e., greater than 1 g/m^3) so that substantial portions can be dumped into the rainshaft cell to produce the observed surface rain rate. In all situations represented by the convective cloud model, the accumulation zone occurs above the freezing level. For that reason, the precipitation fall speeds of ice are those considered at the accumulation zone.

The ice phase precipitation in the convective cloud model is assumed to be graupel. Its fall speed is taken to be a fraction, RAYICE, of the rain fall speed, where

$$\text{RAYICE} = V(\text{ice})/V(\text{rain}), \quad (2.29)$$

and

$$V(\text{ice}) = \text{RAYICE} [V(\text{rain})] \quad (2.30)$$

The lower limit for RAYICE is taken to be 0.4, while the upper limit is taken to be 0.8.

The rain fall speeds, in units of m/s, are given by

$$V(\text{rain}) = 385.267 \left(M_p / N_w \right)^{0.25} \quad M_p < 0.11 \text{ g/m}^3 \quad (2.31)$$

$$V(\text{rain}) = 38.8496 \left(M_p / N_w \right)^{0.125} \quad M_p > 0.11 \text{ g/m}^3 \quad (2.32)$$

The quantity RAYICE can be used to place bounds on the minimum allowable updraft velocity in the updraft cell. To illustrate how, invert Equation (2.32) and solve for M_p :

$$M_p = N_w [V(\text{ice})/\text{RAYICE}/38.8496]^8 \quad (2.33)$$

If RAYICE is assumed to be near its lower limit of 0.4, then according to Equation (2.33), to achieve ice concentrations on the order of 1 g/m^3 , the updraft speed (or fall speed) must exceed 2 m/s. Fall speeds near 2 m/s are reasonable values for graupel particles.

The variable RAYICE is also used to help ensure that M_p does not exceed some predetermined maximum value. At the float level where $V(\text{ice}) = W$, Equation (2.33) can be solved for RAYICE to give

$$\text{RAYICE} = W/38.8496 \left[N_w / M_p(\text{max}) \right]^{0.125} \quad (2.34)$$

where $M_p(\text{max})$ is the maximum allowable concentration for M_p . For each set of input data, $M_p(\text{max})$ is initially set equal to 4.0 g/m^3 .

Equation (2.34) is used to compute RAYICE [and thus, $V(\text{ice})$] in the updraft cell. It ensures that, at the float level, the ice concentration will be less than or equal to $M_p(\text{max})$.

When the updraft velocity becomes large and Equation (2.34) indicates RAYICE will exceed 0.8, one of two actions is taken: either the value of W is reduced or the value of $M_p(\max)$ is increased. Reducing W is equivalent to increasing the storm efficiency for a fixed precipitation rate. To keep RAYICE below 0.8, the storm efficiency is first increased by increments of 0.1. If a storm efficiency value of 0.9 does not reduce RAYICE below the required 0.8 value, then the value of $M_p(\max)$ is increased by increments of 2.0 g/m^3 until RAYICE is reduced below 0.8. Both of these possibilities are considered in subroutine MAINUP and the appropriate action is taken there.

The vertical average of RAYICE in the updraft cell is used in Equation (2.30) to compute $V(\text{ice})$ in the rainshaft cell.

2.2.6 Transfer of Precipitation from the Updraft Cell to the Rainshaft Cell

Once the precipitation mass exceeds 0.2 g/m^3 at the top of a bin in the updraft cell, some fraction is considered to be too heavy to support and is dumped into the rainshaft cell. The mass that is removed contains all those precipitation particles whose fall speeds exceed the updraft speed. Because the particle size distribution is based on a Marshall-Palmer distribution, the mass falling out is given by

$$\text{mass falling out} = M(\text{initial}) - \int_0^{D_c} \lambda N_0 \exp(-\lambda D) dD \quad (2.35)$$

where $M(\text{initial})$ is the mass concentration initially predicted by Equation (2.4) to be present at the top of the bin and D_c is the diameter of the ice particle with fall speed equal to the updraft velocity. Simplifying Equation (2.35) to

$$M(\text{fall}) = M_i - M_o, \quad (2.36)$$

and rewriting it in terms of fluxes gives

$$\text{flux-out} = M_i (w - V_i) - M_o (w - V_o). \quad (2.37)$$

Here, V_i and V_o are median volume fall speeds, computed using the ice mass-fall speed relations given in Section 2.2.5. Note that as Equation (2.37) is written, the flux-out is a negative quantity, because the fall speed of the falling precipitation is greater than the updraft velocity. This is particularly evident when Equation (2.4) predicts a float level with w equaling V_i , and V_o less than w .

The flux-out computed by Equation (2.37) is added to the rainshaft cell at the same height and represents the precipitation transferred from the updraft cell to the rainshaft cell. The flux of precipitation continuing upward to the next bin is given by

$$\text{flux-up} = M_o (w - V_o) \quad (2.38)$$

Calculation of the upward flux and the fallout flux is done in subroutines FALOUT and WATER4. The formulas used to compute D_c and M_o are given in Section 4.20.

2.2.7 Pseudodiffusion

For rain and snow, a "pseudodiffusivity" is used to set the parameters for the vertical dispersion of precipitation that results from the variation of settling velocity with particle size. Because the model does not directly consider diffusivities, this diffusion is expressed in terms of a diffusion velocity

$$V_d = -D_s \frac{\partial}{\partial z} \left[\ln \left(\frac{M_p}{p_a} \right) \right] \quad (2.39)$$

where D_s is the diffusivity expressed as

$$D_s = \Delta z \left(\overline{V_p^2} - \bar{V}_p^2 \right) / \bar{V}_p \quad (2.40)$$

Here, $\overline{V_p^2}$ is the mass-weighted average of the square of the settling velocity and $\overline{V_p}$ is the mass-weighted average of the settling velocity. The diffusion velocity is computed in subroutine VDEP.

2.2.8 Scavenging (Accretion) Coefficients

The collection of cloud water by precipitation, C_p , is defined in terms of a scavenging coefficient, Ψ , as

$$C_p = M_c(j)\Psi \quad (2.41)$$

The scavenging coefficients are functions of the precipitation water concentration and are based on Scott (1982). For rain

$$\Psi = \left(7.008 \times 10^{-4}\right) N_w^{0.125} E_w \bar{M}_p^{0.875} \quad (2.42)$$

Graupel is assumed to be a spherical ice particle with bulk density of 0.9. Its scavenging coefficient is

$$\Psi = \left(6.3072 \times 10^{-4}\right) \text{RAYICE } N_w^{0.125} E_s \bar{M}_p^{0.875} \quad (2.43)$$

where units are per second when M_p is in g/m^3 . Collection efficiencies for rain, E_w , and snow, E_s , and Marshall-Palmer intercepts for rain, N_w , are input in program WETUP.

2.2.9 Autoconversion

The autoconversion rate in $\text{g/m}^3/\text{s}$ is from Berry and Reinhardt (1974) and is given by

$$A_p = 5.5 \times 10^9 M_c^2 \chi_c^{5/3} \left[1 - \left(\frac{2.35 \times 10^{-9}}{\chi_c} \right)^{4/3} \right] \left[1 - \left(\frac{2.04 \times 10^{-9}}{\chi_c} \right)^{1/3} \right] \quad (2.44)$$

where χ_c is the average cloud drop mass;

$$X_c = M_c / N_d \quad (2.45)$$

and N_d is the concentration of cloud droplets per meter. The cloud droplet concentration is held constant in the current version at $500/\text{cm}^3$. Changes in the cloud droplet concentration can be achieved by adjusting the variable DROPS in program WETUP.

Autoconversion is allowed to proceed within the cloud regardless of cloud temperature. However, if the cloud temperature is colder than 0°C , large cloud droplets are assumed to freeze immediately after being converted into raindrops. Thus, there is no intermediate raindrop category above the freezing level.

3.0 FINITE DIFFERENCES EXPRESSIONS

To facilitate the identification of the more complex equations in the computer code, their finite difference forms are presented here. The subroutines where the finite difference expressions appear are also identified.

To express the cloud water equation, (2.3), in the finite difference form used in subroutines WATER3 and WATER4, set C_p equal to $\Psi \bar{M}_c$, set $\bar{M}_c$ equal to $[M_c(j+1) + M_c(j)]/2$, and solve for $M_c(j+1)$:

$$M_c(j+1) = \frac{M_c(j) [w(j) - 0.5 (\Psi \Delta z + Gx)] + \left(D_c - A_p + S_c \right) \Delta z}{w(j+1) + 0.5 (\Psi \Delta z + Gx)} \quad (3.1)$$

Using similar substitutions, the precipitation water concentration, Equation (2.4), is written in finite difference form as

$$\begin{aligned} & - \left[M_p (V_p - V_d - w) \right]_j + \left[M_p (V_p - V_d - w) \right]_{j+1} \\ & + \left(D_p + C_p + A_p + S_p \right) \Delta z - \bar{M}_p Gx = 0 \end{aligned} \quad (3.2)$$

where

$$\bar{M}_p = [M_p(j+1) + M_p(j)]/2 \quad (3.3)$$

Equation (3.2) is also solved in routines WATER3 and WATER4.

In finite difference form, the continuity equation, (2.15), becomes

$$u(x)/\Delta x = \frac{\rho(j+1)w(j+1) - \rho(j)w(j)}{\bar{\rho}(j)\Delta z} + \frac{u(x+\Delta x) - u(x)}{\Delta x} + \frac{v(y+\Delta y) - v(y)}{\Delta y} \quad (3.4)$$

where

$$\bar{p}(j) = [p(j+1) + p(j)]/2, \quad (3.5)$$

and Δx has been set equal to Δy .

Equation (3.4) is solved in subroutine WINDUP by applying several assumptions about the vertical variation of the inflow u and v components to the appropriate rainshaft or updraft cell. These assumptions have been discussed in Section 2.2.1.

The diffusion velocity is based on a pseudodiffusivity value, V_d , resulting from a variation of settling velocity with particle size as in Equation (2.39). V_d is expressed as

$$V_d = - \frac{D_s}{\Delta z} \ln \left[\left(\frac{M_p}{p_a} \right)_{z+\Delta z/2} / \left(\frac{M_p}{p_a} \right)_{z-\Delta z/2} \right] \quad (3.6)$$

where D_s is given by Equation (2.40). After substitution, D_s becomes

$$D_s = \left[\Delta z / V_p(j) \right] 96.39 \left[M_p(j) / N_w \right]^{0.25} \quad (3.7)$$

for rain and

$$D_s = \left[\Delta z / V_p(j) \right] 78.1 \left[M_p(j) / N_w \right]^{0.25}$$

for graupel.

The diffusion velocity is computed in routine VDEP.

4.0 CLOUD-GENERATION ROUTINES

Table 4.1 shows the relationships among the cloud-generation routines. The main program, WETUP, calls the routines in the Level 1 column. The Level 1 routines call the Level 2 routines, and the Level 2 routines call the Level 3 routines. Routines WETUP, MAINUP, WATER3, WATER4, and WINDUP do the bulk of the operations. Several of these routines write diagnostic messages indicating some unacceptable condition has occurred. The actual messages and their interpretations are presented in Section 6.0.

4.1 PROGRAM WETUP

WETUP is the main program for the cloud model. Various input conditions, constants, and flags are set in this program. The program initiates calls to other routines that produce profiles of temperature and moisture, and it removes default cloud profiles to mass storage devices.

The cloud model is currently configured to generate tables of profiles for various input precipitation rates, surface temperatures, and cloud top and cloud base heights. The starting values for computations, the increments or step sizes for successive calculations, and the number of calculations are determined by the user near the beginning of WETUP. Specific instructions for executing WETUP are provided in Section 5.0.

The following values set in WETUP are constant:

- surface pressure = 980 mb
- surface height = 0 m (height above surface)
- surface humidity = 30%
- collection efficiency for water = 0.90
- collection efficiency for ice = 0.90
- Marshall-Palmer intercept for water, $N_w = 1 \times 10^7 \text{ m}^{-4}$
- Marshall-Palmer intercept for ice, $N_s = 5 \times 10^7 \text{ m}^{-4}$
- cloud drop concentration = $500/\text{cm}^3$
- width of each cell = 10 km
- storm efficiency = 0.6.

TABLE 4.1. Cloud Generation Routines

<u>Main Program</u>	<u>Level 1</u>	<u>Level 2</u>	<u>Level 3</u>
WETUP			
	MAINUP		
		WATER3	
			COEF
			VDEP
			NEUTON
			ZERO1
		WATER4	
			COEF
			VDEP
			NUKE
			SEARCH
			NEUTON
			ZERO1
			FALOUT
		WINDUP	
			COEF
			NEUTON
			FLUX
		DEFAULT	
		NEUTON	
		UNITS	
	SIGMA		
		PRESSF	
	SIGMAD		
		PRESSFD	
	DEFAULT		
	CLOUD_BASE		
	FREZLVL		

For each cloud top height and surface temperature combination, subroutines SIGMAD and SIGMA are called to determine the temperature-height profiles. Each cell is assigned the same cloud top temperature, and the temperature lapse rate is assumed to be moist adiabatic within the cloud and dry adiabatic below the cloud base. The cloud base height is at the surface for the rainshaft cell and at the lifting condensation level for the updraft cell. A call to subroutine CLOUD_BASE determines the cloud base for the updraft cell, and calls to subroutine FREZLV determine the freezing-level heights for both cells.

Cloud top temperatures warmer than 0°C are not permitted in the model. If WETUP discovers such an occurrence, a diagnostic message is written and subroutine DEFAULT is called to substitute data from the last successful run for the current input conditions.

If cloud top height and surface temperature are held constant and only surface precipitation amount is being varied, then it may be more efficient to vary storm duration than to compute new cloud parameters. For example, if

$$R(1) T(1) = A(1) \quad (4.1)$$

where

$R(1)$ = previously computed surface precipitation rate

$T(1)$ = previously computed storm duration

$A(1)$ = previously computed hourly precipitation amount,

then for a new hourly precipitation amount, $A(2)$,

$$R(1) T = A(2) \quad (4.2)$$

If WETUP computes T to be less than one hour, then the previously computed values of cloud parameters can be used to define a storm producing $A(2)$ precipitation in one hour. WETUP calls DEFAULT to substitute the previously computed values and writes a diagnostic message indicating the operation has occurred.

If the default mode is triggered by a call to DEFAULT, WETUP writes 25 variables to mass storage. To store more variables, the WRITE(9) and WRITE(10) statements must be rewritten in program WETUP and subroutine MAINUP.

4.2 SUBROUTINE MAINUP

Subroutine MAINUP adjusts the vertical velocities and storm efficiency to give the desired surface precipitation rate. Much of the program output is produced here.

The logic of the routine flows as follows:

- a) calculations start in MAINUP with a storm duration of 60 minutes
- b) for the given storm efficiency, E, MAINUP obtains the required updraft profile by calling subroutine WINDUP
 - if the maximum updraft speed from WINDUP is greater than or equal to 2 m/s, MAINUP goes to (c).
 - if the maximum updraft speed is less than 2 m/s, MAINUP decreases the storm duration by 5 minutes (implicitly increasing the precipitation rate), and
 - if the storm duration is greater than 20 minutes, MAINUP returns to (b).
 - if the storm duration is less than 20 minutes, MAINUP reduces E by 0.1 and returns to (a).
- c) MAINUP calls WATER4 to compute precipitation concentrations in the updraft cell
 - if the maximum precipitation concentration is above the maximum allowable level, MAINUP increases E (implicitly reducing the updraft speed). If E already exceeds 0.9, MAINUP keeps E at 0.9 and increases the maximum allowable precipitation concentration. It then returns to (a).
 - if the maximum precipitation concentration is below the maximum allowable level, MAINUP goes to (d).

- d) if cloud top cloud water concentration is less than 2 g/m^3 , MAINUP goes to (e). If it is not, MAINUP decreases storm duration to increase winds. MAINUP then goes to (b).
- e) MAINUP calls WATER3 to compute the precipitation concentration in the rainshaft
 - a solution exists if the updraft profiles in the rainshaft can be adjusted to provide the precipitation rate determined from

$$\text{precip rate} = \frac{\text{hourly precipitation amount}}{\text{storm duration}} \quad (4.3)$$

To adjust the mean updraft velocity in the rainshaft, W , MAINUP compares the precipitation rate computed with the current guess of W to the desired precipitation rate. The difference between the two rates, DIFF1, is then passed to NEUTON and a new value for W is returned. MAINUP goes to (f) when convergence is achieved.

- if WATER3 is unable to provide a solution, MAINUP increases E and returns to (a).
- f) MAINUP provides hardcopy output on device IOUT and output cell parameters on devices 9 (for the updraft cell) and 10 (for the rainshaft cell).

Subroutine MAINUP will substitute the last set of successfully computed cloud parameters for the current input conditions in five situations. In these situations, subroutine DEFAULT is called to make the substitutions. The default situations are

- 1) when inflow or updraft velocity requirements cannot be satisfied with a storm efficiency less than 0.9
- 2) when storm updraft velocity requirements cannot be satisfied with a storm efficiency greater than 0.1
- 3) when precipitation concentrations greater than 12 g/m^3 in the updraft are needed to provide the desired precipitation rate

- 4) when a solution has not been achieved after 30 tries
- 5) when an unreasonably high amount of precipitation is dumped into the rainshaft at low updraft speeds.

Once proper vertical velocity profiles have been determined, MAINUP prints out all information requested before returning to WETUP for new input conditions.

MAINUP is designed to write data onto storage devices as if it were being stored in an array $R(NVM, NP, NH, NT)$, where NVM is the number of vertical grids, NP is the number of precipitation increments, NH is the number of height increments, and NT is the number of temperature increments. The number of vertical grids (NVM) is established by setting the variable L in common block 'SCAV.FOR'. The rainshaft cell data are written on device 10, while the updraft cell data are written on device 9. Device 9 also contains information (added in WETUP) about the storm duration and actual precipitation rate. Device 11 is used for interim storage.

Currently, MAINUP writes 25 variables to mass storage. To store more variables, the WRITE(9) and WRITE(10) statements must be modified in the program. The same changes in format must also be made in WETUP.

The variables placed in mass storage and their units are

- cloud water for the updraft and rainshaft cells (g/g)
- precipitation water for the updraft and rainshaft cells (g/g)
- updraft speed for the updraft and rainshaft cells (cm/s)
- scavenging coefficient for the updraft and rainshaft cells (1/s)
- deposition or evaporation rate for the updraft and rainshaft cells (1/s)
- precipitation rate for the updraft and rainshaft cells (mm/h)
- precipitation fall speed for the updraft and rainshaft cells (cm/s)
- autoconversion rate for the updraft and rainshaft cells (1/s)
- precipitation water dumped from updraft to rainshaft ($\text{g/cm}^2/\text{s}$)

- flux from the environment to the rainshaft cell (cm^2/s)
- flux from the rainshaft cell to the environment (cm^2/s)
- flux from the rainshaft cell to the updraft cell (cm^2/s)
- flux from the environment to the updraft cell (cm^2/s)
- flux from the updraft cell to the environment (cm^2/s)
- flux from the updraft cell to the rainshaft cell (cm^2/s)
- actual storm precipitation rate (mm/h)
- actual storm duration (min)

Units can be changed by adjusting the conversion factors in subroutine UNITS (called at the end of MAINUP).

4.3 SUBROUTINE WATER3

Subroutine WATER3 provides vertical profiles of condensed water in the rainshaft cell. To do so, the updraft velocity is adjusted iteratively to provide the precipitation rate defined by Equation (4.3).

Subroutine WATER3 passes the computed precipitation rate to MAINUP, where it is compared with the desired precipitation rate. MAINUP then passes a new guess for updraft velocity to WINDUP, where new wind components are computed for use in WATER3. WATER3 then recomputes profiles of condensed water in the rainshaft cell and passes the new precipitation rate back to MAINUP. The process is repeated until the computed surface precipitation rate agrees with the desired value.

If the call to WATER3 by MAINUP carries new environmental conditions ($\text{IFLG} = 0$), then initial guesses of precipitation concentration in WATER3 are obtained by assuming that the precipitation rate, J , varies linearly from the surface to cloud top. For this interpolation, the precipitation rates are converted to concentrations using the following equation (Scott 1982):

$$M_p = 123.8 N_w^{0.11} J^{0.889}$$

for rain and

$$M_p = 0.117 N_s^{0.0687} J^{0.936} \quad (4.4)$$

for snow.

If MAINUP calls WATER3 with unchanged environmental conditions but with new values for updraft velocity ($IFLG > 0$), the previous values of precipitation and cloud water are used for initial guesses of the true solution.

Early in the subroutine, condensation rates are computed using Equation (2.21). The results are stored in arrays SORCE and SORCEI. Next, a portion of the depositional growth term is computed, and the results are stored in array SDEP. A call to subroutine COEF determines the horizontal winds to be used in the balance equations.

Cloud water concentrations are computed from Equation (3.1) by integrating from the ground to the cloud top and by using the previous guess of precipitation water, M_p , for the loss terms represented by the accretion, autoconversion, and deposition coefficients.

Precipitation water concentrations are computed from Equation (3.2) by integrating from the cloud top to the surface. Subroutine NEUTON is used to determine $M_p(j)$ at each height so that Equation (3.2) is balanced. Thus, within this integration from cloud top to the ground, a separate iteration determines M_p at each height.

If convergence is not achieved within this iteration after 11 tries, subroutine ZERO1 is called to search the interval of M_p values from 0.0001 to 30 g/m³ for a solution. If ZERO1 determines that a solution exists, the iteration in the precipitation water loop of WATER3 (DO 260) begins again with its initial guess near the solution. If a solution does not exist, a diagnostic message is written (see Section 6), and control returns to MAINUP, where a different value for storm efficiency is tried.

When updated values of precipitation water have been determined, cloud water concentrations are recomputed. These updated values of cloud water are used to recompute M_p until negligible changes in M_c and M_p occur for each loop.

When the convergence criteria have been satisfied, WATER3 computes a surface precipitation rate and passes the result, XJ1, out to MAINUP for comparison with the desired value.

During these iterations between cloud and precipitation water in the rainshaft cell, the sign of the precipitation rate is checked. If the precipitation rate becomes negative, indicating precipitation moving upward, an unrealistic split in the precipitation motions is assumed to have occurred. A diagnostic message is written, and control returns to MAINUP, which starts the process anew in WATER3, with a downdraft throughout the entire rainshaft cell.

Precipitation concentrations can change drastically near the freezing level because of the acceleration of melting snow. Therefore special care must be taken in computing concentrations around the freezing level. The important term to consider is the first term in Equation (2.4), $G_x(i)\bar{M}_p(i)$, where the overbar represents the average value of M_p in bin i . By referring to Figure 2, the average of M_p can be represented as

$$\bar{M}_p = \left\{ \left[M_f + M_p(i) \right] (H_f - z) + \left[M_p(i+1) + M_f \right] (z + \Delta z - H_f) \right\} / 2\Delta z \quad (4.5)$$

The value of M_f is estimated by extrapolating from the next higher bin by

$$M_f = M_p(i+1) + \left[M_p(i+2) - M_p(i+1) \right] (H_f - z_{i+1}) / \Delta z \quad (4.6)$$

In all other bins, $\bar{M}_p$ is computed as a linear average of the values at the top and bottom of the bin.

4.4 SUBROUTINE WATER4

Subroutine WATER4 provides vertical profiles of condensed water in the updraft cell. The updraft for the cell is required to satisfy the storm efficiency requirements described by Equation (2.10). The shape of the updraft profile is designed to produce accumulation or float zones of precipitation near the cell top. The updraft magnitude is provided by subroutine WINDUP.

FIGURE 2. Precipitation Water Concentration in the Vicinity of the Freezing Level

WATER4 uses the same logic as WATER3 to compute the deposition and condensation rates. Therefore, the beginning portions of the two subroutines are similar; however, WATER4 contains an additional section to determine the value of the variable RAYICE (discussed above in Section 2.2.5).

In the first pass through the subroutine, precipitation concentrations are estimated at every grid above cloud base by assuming the concentration is equal to the float value for that particular level. The float concentration is determined from Equation (2.33) with $V(\text{ice})$ equal to the updraft speed:

$$M_p = N_w (w/\text{RAYICE}/38.8496)^8 \quad (4.7)$$

At cloud base, the precipitation concentration is set equal to 10^{-5} g/m^3 if the cloud base is above the freezing level; otherwise, it is set equal to 0.

Cloud and precipitation water concentrations are computed simultaneously by integrating from the ground to the cloud top, using Equations (3.1) and (3.2). For the cloud water calculations, previous guesses of M_p are used to determine the loss terms represented by the accretion, autoconversion, and deposition coefficients. Thus, there is an iteration here in which the cloud water concentration is calculated first and then used in computing the

precipitation water. The new value for precipitation water is then used to update the cloud water estimate, and so on, until negligible changes occur for each iteration.

As in WATER3, within the integration from the ground to cloud top, there is a separate iteration to determine M_p at each height. For each guess of M_c , subroutine ZERO1 is called to search the interval of M_p values from 0.0001 g/m^3 to the float concentration for a solution to Equation (3.2). If there are two solutions to Equation (3.2), the correct solution is assumed to be in the vicinity of the upper zero.

When a solution to Equation (3.2) exists, the iteration in the precipitation water loop of WATER4 (DO 260) begins again with its initial guess of M_p near the solution. Subroutine NEUTON is used to compute the exact M_p for the given value of M_c . This exact value is used to compute a new guess for M_c .

If WATER4 fails to find a solution to Equation (3.2), it is assumed that the steady-state criterion of the model has been violated and that an accumulation zone exists where precipitation mass is being dumped from the updraft cell. The amount of precipitation (flux) removed is equal to the excess flux when M_p equals the float concentration in Equation (3.2). The concentration of M_p is set equal to the float concentration and is used to compute a new value for M_c . The precipitation removed is placed in the array POUT.

When the convergence criteria have been satisfied at a given level, the bin-top precipitation concentration is split into a portion that continues upward in the updraft and a portion that falls out into the rainshaft. The precipitation removed is added to the array POUT. This splitting process has been described in Section 2.2.6. It is accomplished after calls to subroutine FALOUT.

4.5 SUBROUTINE WINDUP

The subroutine WINDUP computes vertical velocities and horizontal winds. If the flag variable NOPRE passed to WINDUP by MAINUP equals 1, the calculations are for the updraft cell. If NOPRE equals 0, the calculations are for the rainshaft cell.

When calculating values for the updraft cell, the subroutine adjusts the maximum updraft speed to produce the desired storm efficiency. The adjustment is done iteratively by calling subroutine NEUTON. The inflow vapor flux is calculated by using the values of vapor density (VAP) returned after a call to subroutine COEF. The vertical velocity is determined from Equations (2.8) and (2.9), and the horizontal winds are computed from Equations (2.17) and (2.18).

When calculating values for the rainshaft cell, the subroutine uses the updraft velocity magnitude supplied by MAINUP to calculate vertical velocities and horizontal winds according to Equations (2.11), (2.12), (2.13), and (2.19).

The wind values used to compute the horizontal exchange between the two cells and the environment are stored with a call to FLUX at the end of the routine.

4.6 SUBROUTINE SIGMA

Subroutine SIGMA provides the pressure in dynes/cm² and temperature in degrees K at each height grid in the updraft cell. The temperature lapse rate is assumed to be moist adiabatic within the cloud and dry adiabatic below the cloud base. The cloud base height is taken to be the lifting condensation level. The thermodynamic variables are determined in subroutines SAT, SSAT, RHOAIR, and RLAPSE. These four routines have been described by Easter and Hales (1984). All heights used in the calculations are taken to represent height above the surface.

4.7 SUBROUTINE SIGMAD

The SIGMAD subroutine provides the pressure in dynes/cm² and temperature in degrees K at each height grid in the rainshaft cell. The cloud top temperature is taken to be the same as that for the updraft cell. The cloud base is taken to be at the surface, and the temperature lapse rate is assumed to be moist adiabatic from the surface to the cloud top. As in SIGMA, the thermodynamic variables are determined in SAT, SSAT, RHOAIR, and RLAPSE, described by Easter and Hales (1984). All heights used in the calculations represent height above the surface.

4.8 SUBROUTINE PRESSF

Subroutine PRESSF calculates pressure by integrating the hydrostatic equation from the surface to cloud top. Air density is determined from the computed pressure using the gas law.

4.9 SUBROUTINE PRESSFD

Subroutine PRESSFD calculates pressure by integrating the hydrostatic equation from the cloud top to the surface. Air density is determined from the computed pressure using the gas law.

4.10 SUBROUTINE COEF

Subroutine COEF has two different functions depending upon which routine calls it. If the call originates in WINDUP, so the flag V equals -1, the routine returns the average saturation vapor density in the interval z to $z+\Delta z$. If the call originates in WATER3 or WATER4, V equals -2, and the routine computes wind components for Equations (2.5), (3.1), or (3.2).

4.11 SUBROUTINE VDEP

Subroutine VDEP computes the diffusion velocity for rain and graupel when called by WATER3 and WATER4. Equation (3.7) is the finite difference form used by VDEP.

4.12 SUBROUTINE NUKE

Subroutine NUKE computes depositional growth rates. The ice crystal concentration is computed from Equation (2.28) using the average temperature in a bin. The maximum depositional growth rate, DMDT, is computed using Equation (2.25). Subroutine WATER4 checks that the depositional growth rate computed from Equation (2.25) does not exceed the permissible growth rate determined by Equation (2.21).

4.13 SUBROUTINE SEARCH

Subroutine SEARCH computes the average precipitation concentration and the accretion coefficients near the freezing level. Accretion coefficients are determined from Equations (2.42) and (2.43).

Because integration in WATER4 is upward, the procedures for computing precipitation concentrations near the freezing level are slightly different from those used in WATER3 (see Section 4.3). As illustrated by Figure 2, the average of M_p can be represented as

$$\bar{M}_p = \left\{ [M_f + M_p(i-1)] (H_f - z + \Delta z) + [M_p(i) + M_f] (z - H_f) \right\} / 2\Delta z \quad (4.8)$$

The value of M_f is estimated by extrapolating from the lower bins:

$$M_f = M_p(i-2) + [M_p(i-1) - M_p(i-2)] (H_f - z + 2\Delta z) / \Delta z \quad (4.9)$$

4.14 SUBROUTINE DEFAULT

Subroutine DEFAULT is called by MAINUP and WETUP whenever calculated profiles are to be saved. If acceptable values are produced for the current input conditions, DEFAULT saves all of the output variables for a future default set. If the current input conditions result in unrealistic profiles (or if convergence in some iteration scheme is not achieved), DEFAULT restores the previous acceptable values to all output profiles.

4.15 SUBROUTINE NEUTON

Subroutine NEUTON uses Newton's method to iterate on a variable, A, such that the function $F(A)$ equals 0. Here, the function F describes the difference between the computed and the desired values. F is equal to

- the surface precipitation rate differences when A equals updraft velocity in the rainshaft cell
- the mass flux balance differences from Equation (3.2) when A equals precipitation concentration

- the storm efficiency differences when A equals the updraft speed in the updraft cell.

The value of A is adjusted such that

$$A = A_0 - F(A_0) / \left(\frac{dF}{dA} \right) \quad (4.10)$$

In all calls to NEUTON, DIFF1 represents the function F, and convergence is achieved when A has been adjusted so that F equals 0.

4.16 SUBROUTINE ZERO1

Subroutine ZERO1 steps through the values of concentration ranging from R0 to RMAX g/m³ (on a logarithmic scale) searching for a solution to Equation (3.2). For a particular value of concentration, AXIS(K), the left-hand side of Equation (3.2) has a value of VALUE(K). If two successive values of VALUE have different signs, then a zero has been crossed near AXIS(K-1), and the solution flag is set (IFLG=2). If no solution to Equation (3.2) is found, calls to ZERO1 will continue, with larger guesses of concentration, until the value of RMAX is exceeded.

4.17 SUBROUTINE UNITS

Subroutine UNITS converts units from those used internally to those desired by the user. The units used internally are

- lengths in m
- times in s
- mass in g
- cloud water concentration in g/m³
- precipitation water concentration in g/m³
- updraft velocity in m/s
- accretion rate per s
- deposition-evaporation rate in g/m³/s
- precipitation rate in mm/h

- precipitation fall speed in m/s
- autoconversion rate in $\text{g/m}^3/\text{s}$.

The algorithm currently in subroutine UNITS transforms the internal units to those described in Section 4.2.

4.18 SUBROUTINE FLUX

The horizontal average of the horizontal wind fields must be known to determine how the cloud cells interact with each other and with the environment. The quantities F in Equation (2.7), representing the flow out of or into a cell area, are determined in subroutine FLUX. All horizontal winds have been scaled by the factor DZ/DX in the subroutine WINDUP. In FLUX, F is computed by removing the scaling of WINDUP and multiplying by the constant SCALE to obtain the desired cm^2/s units.

4.19 SUBROUTINE CLOUD_BASE

The subroutine CLOUD_BASE determines cloud base height by computing the dry adiabatic lift of an air parcel from the surface to a level where the saturation mixing ratio of the parcel exceeds the surface mixing ratio value. Then an interpolation over pressure is done to find the pressure at which the saturation mixing level value equals the surface mixing ratio value.

4.20 SUBROUTINE FALOUT

The subroutine FALOUT computes the mass of precipitation, M_o , whose diameter is greater than the float diameter. That is, it computes the integral in Equation (2.35). The float diameter is defined as the diameter, D_c , that produces a fall speed, V_p , equal to the updraft speed, w . The float diameter is determined from a fall speed-diameter relationship for graupel (rain):

$$V_p = 130 D_c^{0.5} \quad (4.11)$$

Evaluating the integral in Equation (2.35) gives

$$M_0 = 6k \left\{ \left[1 - \exp(-Dp) \right] - \exp(-Dp) \left(Dp^3/6 + Dp^2/2 + Dp \right) / \lambda^4 \right\} \quad (4.12)$$

where

$$Dp = \lambda Dc$$

$$k = 0.9 \times 10^6 \pi N_s / 6$$

$$\lambda = (0.9 \times 10^6 \pi N_w)^{0.25} / M_p^{0.25}$$

4.21 SUBROUTINE FREZLV

The subroutine FREZLV searches the temperature array to determine the bin height where the temperature first falls below 0°C. A linear interpolation over height is then used within the selected bin to compute the actual freezing level height. Output is freezing level height above the ground, in meters.

5.0 EXECUTION OF THE CLOUD MODEL ROUTINES

The procedures for executing the cloud model routines are presented below. Only a limited amount of information is input interactively. The majority of the data input is done by entering or changing lines of code in the main program, WETUP. If the user wishes to change output format, coding must be changed in WETUP and subroutine MAINUP. These changes must be made before program execution.

5.1 DATA INPUT

The cloud model routines have been tested and found to give satisfactory solutions over the following input-data ranges:

- surface precipitation amount equaling 3 to 15 mm in 1 hour
- freezing level height equaling 1 to 4.5 km
- cloud top height equaling 6 to 12 km.

The farther data deviate from these limits, the less the likelihood of successful execution.

Data are entered near the beginning of WETUP. The program varies values of freezing level height, cloud top height, and precipitation rate. The user must enter the starting values and the ranges for these variations. The first series of input statements set by the user designates the number of initial starting conditions. The data are entered by modifying the following lines of code in WETUP:

```
C ... NF=# OF FREEZING LEVEL INTERVALS BEING CONSIDERED
C ... NH=# OF CLOUD HEIGHT INTERVALS BEING CONSIDERED
C ... NP=# OF PRECIPITATION RATE INTERVALS BEING CONSIDERED
C
      NF=2 (for example)
      NH=3 (for example)
      NP=4 (for example)
```

If only one set of input data is to be considered at a time, meaning that tables of data are not being generated, set NF, NH, and NP equal to 1.

The next input are values specifying cloud top (H0), freezing level height (HFO), and hourly surface precipitation amount in the rainshaft cell (XJ0). All heights are input as height above the surface. The relevant portion of code follows. Note that these are the only data input from the console.

```
C ***** INPUT HEIGHT AS HEIGHT ABOVE GROUND LEVEL *****
C
C ... HFO=INITIAL FREEZING LEVEL HEIGHT TO CONSIDER (METERS)
C ... H0=INITIAL CLOUD TOP HEIGHT TO CONSIDER (METERS)
C ... XJ0=INITIAL SURFACE HOURLY PRECIP AMOUNT TO CONSIDER
C                                     (MM IN 1 HR)
C
      WRITE(6,907)
907  FORMAT(1H ,' CLOUD TOP HEIGHT (M), FREEZING LEVEL HT (M)')
      ACCEPT *,H0,HFO
      WRITE(6,904)
904  FORMAT(1H ,' RAINSHAFT PRECIP AMOUNT (MM IN 1 HR)')
      ACCEPT *,XJ0
```

The next data supplied to the code in WETUP are the increments used when varying cloud top, precipitation rate, and freezing level height. If tables of data are not being generated, these variables are not used and can be omitted. The following lines of code in WETUP must be modified to suit the user's own needs.

```
C
C ... DH=INCREMENT TO USE WHEN VARYING CLOUD TOP AND BASE HEIGHT (M)
C ... DJ=INCREMENT TO USE WHEN VARYING SURFACE PRECIP AMOUNT
C                                     (MM IN 1 HR)
C ... DHF=INCREMENT TO USE WHEN VARYING FREEZING LEVEL HT (M)
C
      DH = -2000.      (for example)
      DJ = 4.0        (for example)
      DHF = 1750.     (for example)
```

Note that the variable DH must be input as a negative number.

The next series of statements found in WETUP sets values for various constants used in the model. The constants currently have the values:

- surface pressure = 980 mb
- surface height = 0 m above surface
- surface humidity = 30%
- storm efficiency = 0.6
- dimension of each cell = 10^4 cm
- maximum allowable precipitation in the updraft cell = 4.0 g/m^3
- collection efficiency for water = 0.9
- collection efficiency for ice = 0.9
- Marshall-Palmer intercept for water, $N_w = 1.0 \times 10^7 \text{ l/m}^4$
- Marshall-Palmer intercept for ice, $N_s = 5.0 \times 10^7 \text{ l/m}^4$
- cloud drop concentration = $5 \times 10^8/\text{m}^3$

If the user wishes to change any value, the change should be made in the following lines of code:

```
C
C ... INPUT SURFACE PRESSURE (PG), HUMIDITY (RHG)
      PG = 980.
      RHG = .3
      RH = RHG
C
      THRESH=4.0 ! MAX RAINSHAFT PRECIP CONCENT. (G/M3)
      E=0.6 ! INITIAL STORM EFFICIENCY
      ESAV=E
      XX=1E+4 ! DIMENSION OF EACH CELL
C
```

C ... INPUT COLLECTION EFFICIENCY FOR WATER (EW), AND ICE (EI)

EW=0.9

EI=EW*0.9 ! INCLUDE ICE DENSITY

C

C ... INPUT MAR.-PALMER No FOR WATER (XNO), AND ICE (BPNO)

XNO=1.E+7

BPNO=5.E+7

AXNO=XNO**0.222*EW

C

C ... INPUT CLOUD DROP NUMBER (#/M3)

DROPS=5.E8

The minimum cloud top temperature is -50°C, and maximum ice crystal concentrations are set at 30/liter. To change these limits, adjust the following lines of code in in WETUP:

C

C ... LIMIT CLOUD TOP TEMP TO > -50 DEG C

IF(TT .LT. -50.) TT=-50.

TOP=TT

C

C ... TOTAL CRYSTAL NUMBER (XNC-#/M3) DEPENDS ON CLOUD TOP TEMP

TTOP=-TOP

IF(TTOP .LE. 0.) TTOP=0.

XNC=6.104E-3*TTOP**4.

C

C ... LIMIT THE TOTAL CRYSTAL NUMBER TO 30000/M3 (=30/L) TO

C SIMULATE AGGREGATION AND PERMIT EARLIER RIMING

IF(XNC .GT. 3.E+4) XNC=3.E+4

DATA(8)=XNC ! CRYSTAL CONCENTRATION FOR RAINSHAFT

C

To change the number of vertical grids used in the model (currently 20), the constant L, which equals the number of vertical grids, must be changed to the desired value in the PARAMETER statement of common block SCAV.FOR.

5.2 DATA OUTPUT

The specific logical units assigned to the following outputs are

- Unit 6: Hardcopy output and diagnostics from WETUP, MAINUP, WATER3, WATER4, and WINDUP.
- Unit 9: Updraft cell data, storm duration, precipitation rate from WETUP and MAINUP.
- Unit 10: Rainshaft cell data from WETUP and MAINUP.
- Unit 11: Interim storage in MAINUP.

The MAINUP and WETUP routines write 25 variables to mass storage. To store more variables, the WRITE(9), WRITE(10), READ(11), and WRITE(11) statements must be modified in MAINUP and WETUP. In addition, the constant LSAVE in the common block SCAV.FOR must be changed to L+1 times the number of variables (L is number of vertical grids), and the new variables must be added to the common block SET.

The variables currently being stored are

- cloud water for the updraft and rainshaft cells (g/g)
- precipitation water for the updraft and rainshaft cells (g/g)
- updraft speed for the updraft and rainshaft cells (cm/s)
- scavenging coefficient for the updraft and rainshaft cells (1/s)
- deposition or evaporation rate for the updraft and rainshaft cells (1/s)
- precipitation rate for the updraft and rainshaft cells (mm/h)
- precipitation fall speed for the updraft and rainshaft cells (cm/s)
- autoconversion rate for the updraft and rainshaft cells (1/s)
- precipitation water dumped from updraft to rainshaft ($\text{g}/\text{cm}^2/\text{s}$)
- flux from the environment to the rainshaft cell (cm^2/s)
- flux from the rainshaft cell to the environment (cm^2/s)
- flux from the rainshaft cell to the updraft cell (cm^2/s)

- flux from the environment to the updraft cell (cm^2/s)
- flux from the updraft cell to the environment (cm^2/s)
- flux from the updraft cell to the rainshaft cell (cm^2/s)
- actual storm precipitation rate (mm/h)
- actual storm duration (min)

Units can be changed by adjusting the conversion factors in subroutine UNITS (called at the end of MAINUP).

5.3 EXECUTION SPEED

The code has been executed on a VAX 11/780 using the VMS operating system. Execution takes approximately 12 seconds for each set of input data.

6.0 DIAGNOSTIC MESSAGES FROM THE CLOUD-GENERATION ROUTINES

Several routines produce diagnostic messages in case of problems. The messages and their interpretations are listed below. The user will see the words in capital letters followed by the values of the indicated variables. The sign or magnitude of these variables will often indicate whether the problem is serious or the message is just informative.

6.1 DIAGNOSTIC MESSAGES FROM WATER3

- NEAR 3002 WATER3, NO ZERO, LEVEL,DIFF=.... No solution for precipitation water concentration exists at the indicated level for the current input conditions. If DIFF is less than 0, then the precipitation concentration is very small, and a split flow results. The SPLIT FLOW diagnostic will be written and program execution will continue normally. If DIFF is greater than zero, then too much water is being dumped into the rainshaft, and control will return to MAINUP, which will search for a solution at a higher storm efficiency.
- KL TOO BIG, LOOP 2000, WATER3, I, KL=... No convergence has been reached at level I after 30 tries. Some discontinuity has caused estimates for cloud or precipitation water to vary widely. Control will return to MAINUP where a new attempt will be made with a higher storm efficiency.
- SPLIT FLOW, TRY SUBSIDENCE IN ENTIRE RAINSHAFT I, J, RC(I), W(I), VP, VD,=..... The updraft in the upper part of the rainshaft is too large, causing precipitation to move upward. Level I is where the problem occurred, and W(I) and VP are the updraft and fall speeds, respectively. Control will return to MAINUP where a negative, parabolic-shaped, vertical velocity profile will be tried in the entire rainshaft.

6.2 DIAGNOSTIC MESSAGES FROM WATER4

- KL TOO BIG, LOOP 2000, WATER4, I, KL=... No convergence has been reached at level I after 60 tries. Some unanticipated situation has

prevented convergence. The routine will have to be carefully debugged with the current input conditions to identify the problem. Execution will halt.

- J=100, NO CONV AFTER ZERO START, WATER4 NEAR 260. The routine was unable to converge to a solution after using a value of precipitation water concentration that was expected to be nearly zero the balance equation. Some unanticipated situation has prevented convergence. The routine will have to be carefully debugged with the current input conditions to identify the problem. Execution will halt.
- DIFF1<0, NEAR 258-WATER4. There was supposed to be a float level in the bin, but there is too much mass leaving the bin to produce an accumulation zone. Some unanticipated situation has occurred. The routine will have to be carefully debugged with the current input conditions to identify the problem. Execution will halt.
- WARNING,FLUXIN=0,DIFF1<0, 916-WATER4. Three values follow the message. If all are zero, the results are suspect; otherwise, there is no problem. The warning message occurs during iteration at grids above the maximum updraft or at grids below the freezing level in the updraft cell. No mass is passing into the bin because of a lower-level float zone or because precipitation has not yet developed at lower levels. Autoconversion or vapor deposition are the only mass sources in the bin. Their growths are the only source for mass passing out of the bin.

6.3 DIAGNOSTIC MESSAGES FROM WETUP

- TG=... TT=... XNO=...
These three variables indicate the surface temperature ($^{\circ}\text{C}$), cloud top temperature ($^{\circ}\text{C}$), and N_0 for rain ($1/\text{m}^4$), respectively.
- CLOUD TOP IS LOWER THAN FREEZING LEVEL FOR H=...,HFLVL=... Cloud top (H) is required to be higher than the freezing level (HFLVL) in the model. If it is not, the last acceptable data will be used for the current input conditions.

- CLOUD BASE > THAN H, DEFAULT USED. H,ZBASE=...The cloud base (ZBASE) is greater than cloud top (H). The last set of good data will be used for current input conditions.
- PREVIOUS VALUES USED FOR STORM WITH RATE=... ACTUAL RATE=... DURATION=... For a constant cloud top height and surface temperature, it may be more efficient to vary storm duration than to compute new cloud parameters for a new hourly precipitation amount. When the previously computed values of cloud parameters can be used to define a storm producing the desired precipitation in one hour, WETUP calls DEFAULT to substitute the previous values. See Section 4.1 for details.
- DESIRED FREEZING LEVEL NOT ACHIEVED, WETUP. Some unanticipated situation has prevented convergence. The program will have to be carefully debugged with current input conditions to identify the problem. Execution will halt.

6.4 DIAGNOSTIC MESSAGES FROM MAINUP

- MAX W TOO LARGE, INCREASE E OR THRESH. To float at the current values of W, the ice fall speeds must be greater than 0.8 of the equivalent rain fall speed. Before THRESH is increased, the storm efficiency, E, will be increased up to 0.9 because a larger E may relax the moisture flux-in requirement and lead to a smaller maximum updraft. If the storm efficiency is as large as allowed, the upper limit of precipitation concentration in the updraft (THRESH) will be incremented by 2 g/m^3 .
- MAX WINDS TOO LOW, DECREASE EFFICIENCY. To achieve the current efficiency, the maximum updraft velocity was less than 2 m/s or the storm duration has been reduced to less than 20 minutes. The routine is unable to obtain large float concentrations at that low updraft speed. MAINUP will decrease the efficiency to increase updraft winds.

- UNREALISTIC CLOUD WATER REQUIRED, CLDMAX=: ... Cloud water concentrations near cloud top are required to be unrealistically large (greater than 3.5 g/m^3) to satisfy current input conditions. CLDMAX is the required concentration. A default mode will be initiated with a call to DEFAULT.
- EFFICIENCY CHANGED TO... A change has been made for the reason stated in the previous diagnostic message. If E equals 1.0, then the threshold value for maximum precipitation will be incremented by 2 g/m^3 , and E will be set at 0.9.
- NO CONVERGENCE IN MAIN. Three values follow. After 30 tries, MAINUP has been unable to obtain a solution. If the absolute value of the second value printed out is less than 0.005, then the current solution will be accepted. Otherwise, DEFAULT is called, and the next diagnostic message follows.
- NO SOLUTION IN MAINUP, DEFAULT USED M, J, E, THRESH, FLOATL=..... Default has occurred because one of the above values is outside its acceptable limits. The acceptable ranges are
 - M < 30 (counter for WATER4 convergence)
 - J < 20 (counter for WATER3 convergence)
 - E = Storm efficiency between 0.1 and 0.9
 - THRESH < 12 (maximum precipitation concentration in g/m^3)
 - FLOATL < 50 (maximum precipitation dumped into the rainshaft)

The last good data will be used for the current input conditions.

7.0 INTERPOLATION ROUTINES

The relatively slow cloud-generation routines provide tables of data for rapid access of information at a later time. Several routines have been developed to interpolate within these tables. Table 2 shows the relationships between these routines. The first routines to be executed are TABLEH and HEIGHT. These routines remove tabulated profiles from two mass storage devices and combine them to form a single table. The second group of routines to be executed contains BEGIN, SELECT, and INTERP. These routines interpolate from the table formed by TABLEH to obtain vertical profiles at desired input conditions. The function of each subroutine is briefly described in this section. The execution procedures will be described in Section 8.0.

7.1 PROGRAM TABLEH

Routine TABLEH unloads the convective cloud data from Units 9 and 10 and forms a new table on Unit 8 with the user's desired grid spacing. The user's vertical grid is assumed to have linear height coordinates for its vertical scale.

To understand the unloading sequence in TABLEH, it is necessary to understand how the variables are stored by the main convective cloud routines. In these routines, each variable to be saved is written in a particular order to two mass storage devices: to Unit 10 for the convective rainshaft data and to Unit 9 for the convective updraft data. Each variable is stored as if it were being loaded into a multidimensional array R with the dimensions $R(NVM, NP, NH, NF)$. Here, NVM is the number of vertical levels used in the cloud model program, NP is the number of discrete surface precipitation amounts considered, NH is the number of cloud top heights evaluated, and NF is the number of freezing level heights used. As in normal array storage, the leftmost subscripts vary most rapidly.

The TABLEH routine reads in a variable's vertical profile for a particular combination of NP, NH, and NF, and places the profile values in arrays with names ending with the letter 'E'. Program HEIGHT is then called to map the data onto the vertical grid spacing selected. These new data are stored in

TABLE 2. Interpolation Routines

<u>Main Routine</u>	<u>Subroutines</u>	
	<u>Level 1</u>	<u>Level 2</u>
TABLEH	HEIGHT	
BEGIN	SELECT	INTERP

arrays with names ending with the letter 'X'. These newly interpolated 'X'-variables are written to Unit 8, thereby producing a new table of data loaded exactly as in R, but with the user's vertical grid spacing.

Currently, TABLEH is configured to read and operate on 15 variables:

- cloud water (g/g)
- precipitation water (g/g)
- updraft speed (cm/s)
- scavenging coefficient (1/s)
- deposition or evaporation rate (1/s)
- precipitation rate (mm/h)
- precipitation fall speed (cm/s)
- autoconversion rate (1/s)
- precipitation water dumped from updraft to rainshaft ($\text{g}/\text{cm}^2/\text{s}$)
- flux from the environment to the rainshaft cell (cm^2/s)
- flux from the rainshaft cell to the environment (cm^2/s)
- flux from the rainshaft cell to the updraft cell (cm^2/s)
- flux from the environment to the updraft cell (cm^2/s)
- flux from the updraft cell to the environment (cm^2/s)
- flux from the updraft cell to the rainshaft cell (cm^2/s)

7.2 SUBROUTINE HEIGHT

Subroutine HEIGHT interpolates data from the vertical grid spacing used in the convective cloud models to the vertical grid spacing of the user's model. A linear grid spacing must be used for both. The vertical coordinate is in

terms of height rather than pressure or sigma coordinates. The present interpolation is based on a third-order Taylor-series approximation to the true solution.

The convective cloud variables are passed into HEIGHT in arrays with names ending in the letter 'E'. Interpolated variables are passed out of the routine in arrays with names ending in the letter 'X'. The variable JJ serves as the counter for storage position in the interpolated arrays.

7.3 PROGRAM BEGIN

The BEGIN routine is a dummy program provided to illustrate the calling sequence for the interpolation routines SELECT and INTERP. These two routines are necessary for interpolating cloud data from the table produced with the user's vertical grid spacing. In the call to SELECT, the variable LIM must equal 2. The call must also contain the variables Q(i) where

Q(1) = hourly surface precipitation amount (mm)

Q(2) = cloud top height (m)

Q(3) = freezing level height (m)

and the variable IIN, which is the number of the device containing the tabulated cloud data.

7.4 SUBROUTINE SELECT

The subroutine SELECT reads the tabulated convective cloud data from unit IIN into memory and places the interpolated cloud variables into common block TABLE.FOR. The actual interpolation is done by subroutine INTERP.

All interpolated rainshaft variables are placed in arrays ending in 'R', and all interpolated updraft variables are placed in arrays ending in 'I'. This placement is done near the end of the routine. In the first pass through loop 6000 in SELECT, the rainshaft variables are in arrays ending in 'I'. These are copied into the correct 'R'-ending arrays in loop 675. The second pass through loop 6000 ends with the updraft variables in the correct 'I'-ending arrays.

Variables are placed into arrays as follows:

R Arrays

- RR = cloud water in rainshaft (g/g)
- RCR = precipitation water in rainshaft (g/g)
- WR = updraft speed in rainshaft (cm/s)
- ALPHR = scavenging coefficient in rainshaft (1/s)
- DEPOSR = deposition or evaporation rate in rainshaft (1/s)
- PRATER = precipitation rate in rainshaft (mm/h)
- FALLR = precipitation fall speed in rainshaft (cm/s)
- AUTOR = autoconversion rate in rainshaft (1/s)
- FEN_CDR = flux from the environment to the rainshaft cell (cm^2/s)
- FCD_ENR = flux from the rainshaft cell to the environment (cm^2/s)
- FTOCELR = flux from the rainshaft cell to the updraft cell (cm^2/s)

I Arrays

- RI = cloud water in updraft cell (g/g)
- RCI = precipitation water in updraft cell (g/g)
- WI = updraft speed in updraft cell (cm/s)
- ALPHI = scavenging coefficient in updraft cell (1/s)
- DEPOSI = deposition or evaporation rate in updraft cell (1/s)
- PRATEI = precipitation rate in updraft cell (mm/h)
- FALLI = precipitation fall speed in updraft cell (cm/s)
- AUTOI = autoconversion rate in updraft cell (1/s)
- POUTI = precipitation water dumped from updraft to rainshaft
($\text{g}/\text{cm}^2/\text{s}$)
- FEN_CDI = flux from the environment to the updraft cell (cm^2/s)
- FCD_ENI = flux from the updraft cell to the environment (cm^2/s)
- FTOCELI = flux from the updraft cell to the rainshaft cell (cm^2/s)

IMPORTANT NOTE

Subroutine INTERP supplies the variables TIMEI and RATEI to subroutine SELECT. The variable TIMEI represents the interpolated storm duration in minutes and RATEI is the interpolated surface precipitation rate in mm/h. It is important to recognize that the input hourly precipitation amount is not

necessarily equal the precipitation rate expressed in mm/h. The constraints on storm efficiency mean that the convective cloud model often has to assume that the hourly precipitation amount falls in intervals smaller than one hour. Thus, the precipitation rate may be considerably larger than the input precipitation amount, and the storm duration may be much less than one hour. If the time and rate values are important to the user, they must be flagged in SELECT.

7.5 SUBROUTINE INTERP

The subroutine INTERP interpolates from the table of values supplied by SELECT to find vertical profiles at given values of cloud top height, surface precipitation amount, and freezing level height. A third-order interpolation scheme is used to determine the profile value at each point.

8.0 EXECUTION OF THE INTERPOLATION PROGRAMS

Execution of the interpolation programs depends on what data are available to the user. If the updraft and downdraft data are combined in a single table, then the routine BEGIN can be executed directly without any preliminary table manipulations. If, however, just the raw updraft and downdraft data are available on two separate tables, then these tables must be merged using TABLEH before BEGIN can be executed. Regardless of the form of the tables, some preliminary setup work must be done before either TABLEH or BEGIN can be executed. Section 8.1 discusses the preliminary setup procedures. Section 8.2 discusses how to combine tables. Section 8.3 describes the procedures to follow when using merged tables. Section 8.4 briefly discusses execution speed.

8.1 EXECUTION REGARDLESS OF STARTING POINT

Before executing TABLEH or BEGIN (SELECT), it is necessary to ensure that various parameters have proper values in statements in block INCLUDE 'TABLE.FOR'. In particular, values must be set for the number of cloud top intervals (NH), the number of precipitation intervals (NP), the number of freezing level height intervals (NT), and the number of vertical grids (NVM) used by the convective cloud model during the generation of tables. Also, the number of vertical grids in the user's model (NV) must be specified.

In TABLE.FOR, various increments and starting values must be defined that correspond to those used in the main convective cloud-generation routines. These values are supplied through the variable HMAX and arrays DV and VO in DATA statements.

When the user's model has height as the vertical coordinate, HMAX represents the upper boundary height. If height is not the vertical coordinate, HMAX is not used.

The array DV contains the increments of initial conditions used in the convective cloud model. The variable DV(1) represents increments of hourly surface precipitation used as input, currently 4.0 mm. The variable DV(2) is the increment for cloud top height, currently set in the cloud routines to

decrease from 12.0 km in 2000 m increments. Thus, the value of DV(2) is -2000. The variable DV(3) is the increment for freezing level height, currently 1750 m.

The array V0 contains the starting values for each of the intervals defined by DV. The variable V0(1) represents the first precipitation amount value and is taken to be 4.0 mm in one hour. Successive precipitation amounts are determined by DV(1) and thus are incremented by 4.0 mm. The variable V0(2) is the starting value for cloud top height, 12.0 km, and V0(3) is the starting value for freezing level height, 1000 m. Variables V0(2) and V0(3) are incremented by DV(2) and DV(3), respectively.

As an example, consider a table with cloud tops set at 12, 10, 8, and 6 km, hourly precipitation amounts of 4, 8, and 12 mm, and freezing level heights ranging from 1 km to 4.5 km in 1.75 km intervals. The appropriate lines of code in block TABLE.FOR would be

```
PARAMETER (.....NH=4,NP=3,NT=3,.....)
```

```
.  
.   
.
```

```
DATA DV /4.,-2000,1750,0,0,0/
```

```
DATA V0 /4.,12000,1000,0,0,0/
```

Notice that the initial values and the intervals are given in the DATA statements, while the total number of intervals is given in the PARAMETER statement.

After the proper control statements to load the data on the desired devices have been made and all desired variables have been written by the main convective cloud routine, no other changes or checks are required before the interpolation routines are executed. Be sure to recompile if changes are made in the TABLE.FOR block.

8.2 MERGING OF THE TABLES (TABLEH)

Routine TABLEH takes the convective updraft data from Unit 9 and the convective rainshaft data from Unit 10 and combines the data into a single table written on Unit 8. The input units must be numbered sequentially. The updraft data must be on the first (or lowest-numbered) unit, and the rainshaft data must be on the second (or next-higher-numbered) unit. To change the unit numbers such that the rainshaft data is on Unit X, change the statement

INUNIT = 11

to

INUNIT = X + 1

The table generated by TABLEH is output on Unit 8. The output device can be changed simply by setting IOUT equal to the new device number near the beginning of the routine.

8.3 EXECUTION USING MERGED TABLES

Routine BEGIN calls subroutine SELECT. SELECT expects to obtain the merged, tabulated data from Unit 8. If the input data is on some other device, that device number is set in the call to SELECT by the variable IIN.

Routine BEGIN asks for data to be entered from the console. The number 1 should be typed for a convective cloud. Following a prompt, appropriate values should be entered for rain amount, cloud top, and freezing level height.

SELECT will automatically provide output to Unit 6 (the console) for debugging purposes. To eliminate this printout, delete all statements from (and including)

WRITE (6,807) TIMEI, RATEI

to (and including)

501 Z + Z + DZ*XINTER

8.4 EXECUTION SPEED

The code has been executed on a VAX 11/780 using the VMS operating system. Execution takes approximately 0.5 seconds if the data table must be read in for each set of input data. If the data table resides in core memory, then execution takes between 0.1 and 0.2 seconds for each set of input conditions.

9.0 DIAGNOSTIC MESSAGES FROM THE INTERPOLATION ROUTINES

Only the INTERP subroutine produces a diagnostic message in case of problems, and it does so in only one case. The message and its interpretation are given below.

DIAGNOSTICS FROM INTERP

OUT OF TABLE LIMITS FOR VARIABLE K=...

DEFAULT OF Q(K) IS TAKEN TO BE....

The value chosen for Q(1) (precipitation amount), Q(2) (cloud height), or Q(3) (freezing level height) is either outside or at the boundary of the tabulated data. The program will substitute the value at the table boundary that is closest to the desired value. Interpolation will proceed with the substitute value placed in array position Q(K).

10.0 LIST OF MAJOR ASSUMPTIONS

The simplifying assumptions that allow the model to be developed are listed in this section.

The solutions are appropriate for convective clouds with horizontal dimensions of less than 50 km. The clouds are represented by an updraft cell and a downdraft cell.

The horizontal dimensions of each cell are the same in the x and y directions.

Only two condensed water categories are used: cloud water and precipitation water. Melting snow and freezing water categories are neglected.

The cloud and precipitation water concentrations are assumed to be horizontally uniform.

At the lateral external inflow boundaries of the cloud, the cloud and precipitation water concentrations are assumed to be zero.

Cloud water moves with the air's velocity, in contrast to precipitation water, which falls with respect to the air.

Arbitrary definitions of the horizontal winds, u and v , have been used at two of the boundaries for each cell.

The updraft velocity in the updraft cell has been selected to provide accumulation zones of precipitation in the upper portion of the cell. The updraft increases linearly from the ground to the height $z(\text{max})$, and then decreases linearly from $z(\text{max})$ to the cloud top.

An inverted S-shaped profile is used to describe the vertical velocity in the rainshaft cell. The profile is parabolic, with upward motions above $H/2$ and downward motions below $H/2$, where H is the cloud top height. If solutions are unobtainable with the S-shaped profile, subsidence is assumed to occur in the entire rainshaft. The rainshaft profile is then taken to be parabolic from the surface to cloud top.

The cloud-top temperature is taken to be the same for both the rainshaft and the updraft cells, and the temperature lapse rate is assumed to be moist

adiabatic within the cloud and dry adiabatic below the cloud base. The cloud base is taken to be at the surface for the rainshaft cell and at the lifting condensation level for the updraft cell.

Cloud top temperatures warmer than 0°C are not permitted in the model.

The water-vapor mixing ratio is constant with height below cloud base.

The air density is a function of height alone.

The storm duration must be greater than 20 minutes.

The storm is assumed to be in a quasi-steady state.

The fall speeds and collection rates of precipitation depend on whether the precipitation is classified as graupel or rain, which in turn depends on the local air temperature.

The ratio of graupel fall speed to rain fall speed at the same mass must be between 0.4 and 0.8.

Graupel is assumed to be composed of spherical ice particles with bulk densities of 0.9.

Once the precipitation mass exceeds 0.2 g/m^3 at the top of a bin in the updraft cell, a certain fraction is considered to be too heavy to support and is dumped into the rainshaft cell. The mass that is removed contains all those precipitation particles whose fall speeds exceed the updraft speed.

Condensation onto liquid precipitation is neglected.

For the updraft cell, the number of ice crystals per unit volume remains constant at each height and is determined by temperature at that height.

For the rainshaft cell, the ice crystal concentrations at each height are determined by cloud top temperature.

The total ice crystal concentration is not allowed to exceed 30 crystals/liter.

The following values are constant:

- surface pressure = 980 mb
- surface height = 0 m above ground

- surface humidity = 30%
- storm efficiency = 0.6 (starting value)
- dimension of each cell = 10 km
- maximum allowable precipitation in the updraft cell = 4.0 g/m^3 (starting value)
- collection efficiency for water = 0.9
- collection efficiency for ice = 0.9
- Marshall-Palmer intercept for water, $N_w = 1.0 \times 10^7 \text{ l/m}^4$
- Marshall-Palmer intercept for ice, $N_s = 5.0 \times 10^7 \text{ l/m}^4$
- cloud drop concentration = $500/\text{cm}^3$.

11.0 REFERENCES

- Anthes, R. A. 1985. Summary of Meteorological Data Sets from PSV/NCAR Meso- α Scale (Regional) Limited-Area Model NCAR. Ms. 0901-85-02, National Center for Atmospheric Research, Boulder, Colorado.
- Berkowitz, C.M., and R. C. Easter. 1986. Interim User's Guide to the RADM Scavenging Module (RSM) (Version 1.0). Atmospheric Sciences Research Laboratory, U.S. Environmental Protection Agency, Research Triangle Park, North Carolina.
- Berry, E. X., and R. L. Reinhardt. 1974. "An Analysis of Cloud Drop Growth by Collection." J. of Atmos. Sci. 31:2118-2126.
- Easter, R. C., and J. M. Hales. 1984. PLUVIUS: A Generalized One-Dimensional Model of Reactive Pollutant Behavior, Including Dry Deposition, Precipitation Formation, and Wet Removal. 2nd ed. PNL-4046, Pacific Northwest Laboratory, Richland, Washington.
- Fletcher, N. H. 1969. The Physics of Rainclouds. Cambridge University Press, London.
- Kessler, E. 1969. "On the Distribution and Continuity of Water Substance in Atmospheric Circulation." Meteorological Monographs 10(32).
- Scott, B. C. 1982. "Predictions of In-Cloud Conversion Rates of SO_2 to SO_4 Based Upon a Simple Chemical and Kinematic Storm Model." Atmos. Environ. 16:1735.

APPENDIX A

SAMPLE OUTPUT

APPENDIX A

SAMPLE DUTPUT

TG= 1.000E+01 TT=-5.000E+01 XN0= 1.000E+07

STORM PRECIPITATION RATE-HOURLY AVERAGE (MM/HR) 8.000E+00

STORM DURATION (MIN)= 6.000E+01 STORM PRECIP RATE (MM/HR)= 8.000E+00

RAINSHAFT-CELL DATA

CELL HEIGHT (M)= 1.000E+04 CLOUD BASE HEIGHT (M)= 0.000E+00

FREEZING LEVEL HEIGHT (M)= 5.323E+02

SURFACE PRECIPITATION RATE (MM/HR)= 8.000E+00

SURFACE TEMPERATURE (K)= 2.831E+02

SURFACE HUMIDITY= 1.000E+00 STORM EFFICIENCY= 6.000E-01

WATER COLLECTION EFFICIENCY= 9.000E-01

ICE COLLECTION EFFICIENCY= 9.000E-01

<u>GRID</u>	<u>CLOUD WATER (G/G)</u>	<u>PRECIP WATER (G/G)</u>	<u>SCAV COEF (1/S)</u>	<u>DEPOS-EVAP COEF (1/S)</u>	<u>UPDRAFT SPFED (CM/S)</u>
1	0.000E+00	3.892E-04	2.487E-03	-7.924E-04	0.000E+00
3	0.000E+00	9.400E-04	2.770E-03	-1.213E-03	-8.182E+01
5	0.000E+00	1.370E-03	3.432E-03	-7.873E-04	-1.200E+02
7	0.000E+00	1.814E-03	3.934E-03	-3.469E-04	-1.146E+02
9	0.000E+00	2.390E-03	4.109E-03	-7.883E-05	-6.546E+01
11	0.000E+00	8.735E-04	1.803E-03	7.625E-05	2.455E+01
13	0.000E+00	5.459E-04	1.016E-03	1.357E-04	9.546E+01
15	0.000E+00	6.241E-04	9.552E-04	8.125E-05	1.227E+02
17	0.000E+00	4.916E-04	7.923E-04	2.458E-05	1.064E+02
19	0.000E+00	2.255E-04	1.827E-04	3.296E-05	4.637E+01

GRID	HEIGHT (M)	PRECIP RATE (MM/HR)	FALL SPEED (CM/S)	AUTO (1/S)
1	2.632E+02	8.000E+00	4.723E+02	0.000E+00
3	1.316E+03	1.243E+01	2.505E+02	0.000E+00
5	2.368E+03	1.872E+01	2.594E+02	0.000E+00
7	3.421E+03	2.225E+01	2.651E+02	0.000E+00
9	4.474E+03	2.324E+01	2.706E+02	0.000E+00
11	5.526E+03	4.768E+00	2.355E+02	0.000E+00
13	6.579E+03	1.542E+00	2.188E+02	0.000E+00
15	7.632E+03	1.240E+00	2.193E+02	0.000E+00
17	8.684E+03	9.263E-01	2.096E+02	0.000E+00
19	9.737E+03	4.907E-01	1.861E+02	0.000E+00

UPDRAFT-CELL DATA

CELL HEIGHT (M)= 1.000E+04 CLOUD BASE HEIGHT (M)= 2.051E+03

FREEZING LEVEL HEIGHT (M)= 1.024E+03

SURFACE PRECIPITATION RATE (MM/HR)= 8.000E+00

SURFACE TEMPERATURE (K)= 2.831E+02

SURFACE HUMIDITY= 3.000E-01 STORM EFFICIENCY= 6.000E-01

WATER COLLECTION EFFICIENCY= 9.000E-01

ICE COLLECTION EFFICIENCY= 9.000E-01

GRID	CLOUD WATER (G/G)	PRECIP WATER (G/G)	SCAV COEF (1/S)	DEPOS-EVAP COEF (1/S)	UPDRAFT SPEED (CM/S)
1	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
3	0.000E+00	0.000E+00	0.000E+00	0.000E+00	5.467E+01
5	3.694E-04	1.001E-08	9.807E-05	1.170E-04	1.093E+02
7	1.025E-03	1.689E-04	2.584E-05	5.931E-05	1.640E+02
9	9.580E-04	8.436E-04	4.282E-03	1.483E-04	2.187E+02
11	4.359E-04	1.552E-03	5.767E-03	1.058E-04	2.733E+02
13	1.724E-04	2.621E-03	5.114E-03	1.536E-04	3.280E+02
15	1.829E-05	6.142E-03	4.711E-03	2.864E-05	3.827E+02
17	2.600E-06	5.234E-03	4.603E-03	1.777E-05	4.373E+02
19	1.270E-06	1.867E-04	1.290E-04	0.000E+00	1.913E+02

GRID	HEIGHT (M)	PRECIP RATE (MM/HR)	FALL SPEED (CM/S)	AUTO (1/S)
1	2.632E+02	0.000E+00	0.000E+00	0.000E+00
3	1.316E+03	0.000E+00	0.000E+00	0.000E+00
5	2.368E+03	-3.379E-05	1.547E+01	0.000E+00
7	3.421E+03	0.000E+00	1.640E+02	0.000E+00
9	4.474E+03	-5.095E-01	1.978E+02	0.000E+00
11	5.526E+03	-1.620E+00	2.330E+02	0.000E+00
13	6.579E+03	-2.038E+00	2.944E+02	0.000E+00
15	7.632E+03	-7.809E-01	3.765E+02	0.000E+00
17	8.684E+03	-2.092E+00	4.154E+02	0.000E+00
19	9.737E+03	-1.305E-01	1.478E+02	0.000E+00

APPENDIX B

PROGRAM LISTINGS

APPENDIX B

PROGRAM LISTINGS

```
C ... COMMON LIST IN 'SCAV.FOR'
C
C ... L=NUMBER OF VERTICAL GRIDS, LP1=L+1, LSAVE=LP1*NUMBER OF
C   VARIABLES IN BLOCK /SET/
C
      PARAMETER (L=20,LP1=L+1,LSAVE=336)
      COMMON /SET/ R(LP1),RC(LP1),W(LP1),ALPH(LP1),DEPOS(LP1)
1,PRATE(LP1),FALL(LP1),AUTO(LP1)
2, POUT(LP1),COUT(LP1),FEN_UP(LP1),FUP_EN(LP1),FUP_DN(LP1)
3, FEN_DN(LP1),FDN_EN(LP1),FDN_UP(LP1)
      COMMON /SET2/ AXNO,BPNO,EI,EW,Z10,TOP,ZBASE,HFU,HFD,NOPRE,
1 C012(LP1),DROPS
      COMMON /SET4/ MULT,AIR(3),Q(5),QI(5),AV(5),AVI(5)
      COMMON /SET5/ VAP(LP1),SORCE(LP1),SORCEI(LP1),SDEP(LP1),RS(LP1),
1 RCS(LP1),VAPI(LP1)
      COMMON /BK1/ G(LP1),UOUT(LP1),VIN(LP1),VO(LP1)
      COMMON /BK2/ DATA(12),TEMP(L),RHO(L)
      COMMON /BK4/ PRESSD(L),TEMPO(L),RHOD(L)
      COMMON SAVE(LSAVE),SAVER(LSAVE)

C .....

C ... PROGRAM WETUP
C
      INCLUDE 'SCAV.FOR'
      COMMON /BK3/ZLZ(L),PRESS(L)
```

DIMENSION XLIM(7)

DATA XLIM/0.,0.,1.E-8,1.E-4,0.,1.E5,-1./

C

C.....

C

C ... MAIN PROGRAM FOR CLOUD/SCAVENGING MODULE

C ... PRIMARILY SETUP AND INITIALIZATION OPERATIONS ARE DONE HERE

C

C.....

C

C ... CONVECTIVE CLOUD MODULE

C

C ... RELEASED NOV, 1986

C

C ... FOR INFORMATION, CONTACT:

C B.C. SCOTT OR C.M. BERKOWITZ

C PACIFIC NORTHWEST LABORATORY

C P.O. BOX 999, RICHLAND, WA. 99352

C

C.....

C

C ... UNITS ARE:

C LENGTHS--METERS

C PRESSURE---MB.

C VELOCITY--METER/SECOND

C TEMPERATURE--DEG C

C RAINRATE--MM/HR

C HUMIDITY--DECIMAL

C

C.....

C

C ... NF=# OF FREEZING LEVEL INTERVALS BEING CONSIDERED

C ... NH=# OF CLOUD HEIGHT INTERVALS BEING CONSIDERED

C ... NP=# OF PRECIPITATION RATE INTERVALS BEING CONSIDERED

```

C
    NF=3
    NH=4
    NP=3

C
C
C ***** INPUT HEIGHT AS HEIGHT ABOVE GROUND LEVEL *****
C
C ... HFO=INITIAL FREEZING LEVEL HEIGHT TO CONSIDER (METERS)
C ... HO=INITIAL CLOUD TOP HEIGHT TO CONSIDER (METERS)
C ... XJO=INITIAL SURFACE HOURLY PRECIP AMOUNT TO CONSIDER
C                                     (MM IN 1 HR)
C
    WRITE(6,907)
907  FORMAT(1H , ' CLOUD TOP HEIGHT (M), FREEZING LEVEL HT (M)')
    ACCEPT *,HO,HFO
    WRITE(6,904)
904  FORMAT(1H , ' RAINSHAFT PRECIP AMOUNT (MM IN 1 HR)')
    ACCEPT *,XJO

C
C ... DH=INCREMENT TO USE WHEN VARYING CLOUD TOP AND BASE HEIGHT (M)
C ... DJ=INCREMENT TO USE WHEN VARYING SURFACE PRECIP AMOUNT
C                                     (MM IN 1 HR)
C ... DHF=INCREMENT TO USE WHEN VARYING FREEZING LEVEL HT (M)
C
    DH=-2000.
    DJ=4.0
    DHF=1750.

C
C ... INPUT SURFACE PRESSURE (PG), HUMIDITY (RHG)
    PG=980.
    RHG=.3
    RH=RHG
C

```

```

    THRESH=4.0 ! MAX RAINSHAFT PRECIP CONCENT. (G/M3)
    E=0.6 ! INITIAL STORM EFFICIENCY
    ESAV=E
    XX=1E+4 ! DIMENSION OF EACH CELL
C
C ... INPUT COLLECTION EFFICIENCY FOR WATER (EW), AND ICE (EI)
    EW=0.9
    EI=EW*0.9 ! INCLUDE ICE DENSITY
C
C ... INPUT MAR.-PALMER No FOR WATER (XNO), AND ICE (BPNO)
    XNO=1.E+7
    BPNO=5.E+7
    AXNO=XNO**0.222*EW
C
C ... INPUT CLOUD DROP NUMBER (#/M3)
    DROPS=5.E8
C
    REWIND 10 ! RAINSHAFT DATA
    REWIND 9 ! UPDRAFT DATA
C
C ... LOOP 200 VARIES SURFACE TEMPERATURE
    HDESIR=HF0
    DO 200 I=1,NF
C
    H=H0
C
C ... LOOP 400 VARIES STORM HEIGHT
    DO 400 K=1,NH
C
C ... ZERO STORAGE ARRAYS
    DO 500 M=1,LSAVE
        R(M)=0.
500    SAVE(M)=0.
C

```



```

C ... DETERMINE TEMPERATURE AT CLOUD TOP (TT)
C
C ... LAPSE IS MOIST ADIABATIC IN CLOUD OR RAINSHAFT
C
C ... ADJUST SFC TEMP TO GET PROPER FREEZING LEVEL HEIGHT
      IF(HDESIR .GE. H) GO TO 406
      TG=10.
      DO 475 J=1,30
C
      ZBASE=H*(1-0.5/FLOAT(L-1)) ! INITIAL GUESS
      DO 450 M=1,2
      CALL SIGMA(H,TG,PG) ! PRESSURE,DENSITY,TEMP PROFILES
C
                                (UPDRAFT)
      IF(M .EQ. 2) GO TO 450
      CALL CLOUD_BASE(TEMP,ZLZ,PRESS,ZBASE,RHG) ! GET CLD BASE
450  CONTINUE ! READJUST P,T RHO WITH PROPER CLD BASE
C
      HFU=H+1.
      CALL FREZLVL(TEMP,ZLZ,L,HFU) ! UPDRAFT FREEZING LEVEL
      DIFF1=HDESIR-HFU
      CALL NEWTON(DIFF1,TG,XLIM,J,IFLAG) ! ADJUST TG TO GET HFU
      IF(IFLAG .EQ. 1) GO TO 476
C
475  CONTINUE
      WRITE(6,807)
807  FORMAT(1H0,' DESIRED FREEZING LEVEL NOT ACHIEVED, WETUP')
      STOP
C
476  IF(ZBASE .LT. H) GO TO 404 ! SEE IF H IS HIGH ENOUGH
      WRITE(6,806) ZBASE,H
806  FORMAT(1H0,' CLOUD BASE > H, DEFAULT USED. H,ZBASE= ',2(1X,1
1 PE10.3))
      GO TO 405
C

```

```

404 CALL SIGMAD ! RAINSHAFT P,T,RHO
C
      CALL FREZLV(L,TEMPD,ZLZ,L,HFD) ! RAINSHAFT FREEZING LEVEL
      TT=TEMP(L)-273.16 ! CLOUD TOP TEMP (C)
C
C ... SEE IF CLOUD TOP IS BELOW FREEZING
406 WRITE(6,803) H,HDESIR
803 FORMAT(1H0,' CLOUD TOP IS LOWER THAN FREEZING LEVEL FOR H= ',
      11PE10.3,' HFLVL=',1PE10.3)
C
405 DO 403 J=1,NP
C
      CALL DEFALT(3)
      WRITE(10) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO
      1 ,FEN_DN,FEN_EN,FEN_UP
      CALL DEFALT(1)
      WRITE(9) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO
      1 ,FEN_UP,FUP_EN,FUP_DN
403 WRITE(9) POUT,TNEXT,ACTJ
      GO TO 400
C
402 XJ=XJO
C
C ... LIMIT CLOUD TOP TEMP TO > -50 DEG C
      IF(TT .LT. -50.) TT=-50.
      TOP=TT
C
C ... TOTAL CRYSTAL NUMBER (XNC-#/M3) DEPENDS ON CLOUD TOP TEMP
      TTOP=-TOP
      IF(TTOP .LE. 0.) TTOP=0.
      XNC=6.104E-3*TTOP**4.
C
C ... LIMIT THE TOTAL CRYSTAL NUMBER TO 30000/M3 (=30/L) TO
C SIMULATE AGGREGATION AND PERMIT EARLIER RIMING

```

```

      IF(XNC .GT. 3.E+4) XNC=3.E+4
      DATA(8)=XNC  ! CRYSTAL CONCENTRATION FOR RAINSHAFT
C
      TIME=61.
      XJSAVO=XJ0
      ACTJ=XJ
C
C ... LOOP 100 VARIES SURFACE RAIN AMOUNT (XJ)
      DO 100 J=1,NP
      E=ESAV
      IF(TIME .GE. 60.) GO TO 104
C
      TNEXT=XJ/XJSAVO*TIME  ! DURATION OF STORM WITH NEW RATE, XJ
      IF(TNEXT .GT. 60.) GO TO 104
      WRITE(6,805) XJ
805  FORMAT(1H0,' PREVIOUS VALUES USED FOR STORM WITH RATE=',
1 1PE10.3)
      WRITE(6,804) ACTJ,TNEXT
804  FORMAT(1H , ' ACTUAL RATE=',1PE10.3,' DURATION=',1PE10.3)
C
      CALL DEFALT(3)
      WRITE(10) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO
1  ,FEN_DN,FEN_EN,FEN_UP
      CALL DEFALT(1)
      WRITE(9) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO
1  ,FEN_UP,FUP_EN,FUP_DN
      GO TO 102
C
104  WRITE(6,801) TG,TT,XNO
801  FORMAT(1H0,' TG=',1PE10.3,' TT=',1PE10.3,' XNO=',1PE10.3)
      XJSAVE=XJ
C
C ... THE BULK OF CALCULATIONS ARE DONE IN 'MAIN' OR 'WATER'
C

```

```

101  CALL MAINUP(E,H,XJ,RH,XX,WBAR,THRESH)
C
      ACTJ=XJ    ! ACTUAL XJ USED IN 'WATER' CALCULATIONS
      IF(J .EQ. 1) TIME=XJSAVE*60./ACTJ    ! DURATION OF BASE STORM
      TNEXT=XJSAVE*60./ACTJ
C
      XJ=XJSAVE
900  FORMAT(1H ,I3,6(1X,1PE10.3))
C
C ... OUTPUT INFO IS STORED IN COMMON BLOCK /SET/
102  WRITE(9) POUT,TNEXT,ACTJ
C
      XJ=XJ+DJ
C
100  CONTINUE
      E=ESAV
C
400  H=H+DH
C
200  HOESIR=HDESIR+DHF
      END

```

```

C .....

```

```

      SUBROUTINE FREZLVL(TEMP,ZLZ,L,HF)
      DIMENSION TEMP(1),ZLZ(1)
C
C ... FIND THE FREEZING LEVEL HEIGHT
C
      IF(TEMP(1) .LT. 273.16) RETURN    ! SFC BELOW FREEZING
      DO 550 M=1,L
      IF(TEMP(M) .GT. 273.16) GO TO 550
      IF(TEMP(M-1) .LE. 273.16) GO TO 550

```

```

C
      HF=(273.16-TEMP(M-1))*(ZLZ(M)-ZLZ(M-1))/(TEMP(M)-TEMP(M-1))
1    +ZLZ(M-1)
      HF=HF/100.    !FREEZING LEVEL HEIGHT ABOVE GROUND-METERS
C
550  CONTINUE
      RETURN
      END

```

```

C .....

```

```

      SUBROUTINE MAINUP(E0,H,XJO,RHGO,XX,WBAR,THRESHO)
C
      INCLUDE 'SCAV.FOR'
      DIMENSION XLIM(7)
      DATA XLIM/0.,0.,1.E-8,1.E-4,0.,1.E5,-1./
      IOUT=6

```

```

C

```

```

C .....

```

```

C
C ... THIS ROUTINE ADJUSTS VERTICAL VELOCITY TO GIVE PROPER SURFACE
C    PRECIPITATION RATE. MUCH OF HARDCOPY OUTPUT IS DONE HERE.
C .....

```

```

C
C ... CONVECTIVE CLOUD MODULE
C
C ... RELEASED NOV, 1986
C

```

C ... FOR INFORMATION, CONTACT:
C B.C. SCOTT OR C.M. BERKOWITZ
C PACIFIC NORTHWEST LABORATORY
C P.O. BOX 999, RICHLAND, WA. 99352
C

C.....
C

C

C

C ... VARIABLES IN CALL LIST:

C EO=STORM EFFICIENCY (PRECIP RATE/VAPOR INFLOW)--INPUT
C H=STORM HEIGHT (M)--INPUT
C XJO=PRECIP RATE (MM/HR)--INPUT
C RHGO=SURFACE RELATIVE HUMIDITY (DECIMAL NOT %)--INPUT
C XX=STORM WIDTH (M)--INPUT
C WBAR=AVERAGE VERTICAL VELOCITY (M/S)--COMPUTED
C THRESHO=MAX PRECIP CONCENTRATION IN UPDRAFT CELL
C

C.....
C

C ... FLAGS:

C IFLAG (FROM NEUTON)=0 WHEN NO CONVERGENCE OCCURS FOR
C VERTICAL VELOCITY
C =1 WHEN VERTICAL VELOCITY CONVERGES
C IFLG,KFLG = COUNTER TO DETERMINE 1ST AND SUCCESSIVE
C TIMES THROUGH 'WINDUP' , AND 'WATER3'
C

```

C.....
C
    HF=HFD ! FOR RAINSHAFT OUTPUT
    THRESH=THRESH0
C
    DT=5
    T=60.
    WBAR=0.75 ! INITIAL GUESS OF W IN UPDRAFT-CELL
    E=E0
    DO 1500 M=1,30
    UPW=THRESH**0.125*4.1537 ! MAX ALLOWABLE W WITH MAX RAYICE=0.8
    KFLG=0
1303 RHG=RHG0
    NOPRE=1
C
    IF(T .LT. 20) GO TO 1502
C
250 H0=H
C
    DATA(11)=WBAR
    DATA(1)=H
    DATA(7)=XJ0 ! INITIAL RATE FROM WETUP
C
C ... 'WINDS' USES W PROFILE AND EFFICIENCY REQUIREMENTS TO COMPUTE
C PROFILES OF HORIZONTAL WINDS U,V AND WMAX
    CALL WINDUP(E,RHG,XX,KFLG,T,DT,WMAX)
    IF(KFLG .EQ. 400) GO TO 1502 ! MUST DECREASE EFFICIENCY
    KFLG=KFLG+1
C
    IF(WMAX .GT. UPW) GO TO 1306 ! REDUCE W, INCREASE E
C
C ... 'WATER4' COMPUTES PROFILES OF CONDENSED WATER
    CALL WATER4(RHG,0,RAYICE,THRESH)
C

```

```

      UPMAX=0. ! FOR MAX RC IN UPDRAFT
      FLOATL=0. ! FOR MAX PRECIP DUMPED OUT OF UPDRAFT
      CLDMAX=0. ! FOR MAX CLOUD WATER AT CELL TOP
C
      DO 100 I=1,L ! COMPUTE UPMAX,FLOATL,CLOMAX
      IF(RC(I) .GT. UPMAX) UPMAX=RC(I) ! MAX RC IN UPDRAFT
      IF(I .LT. LL) GO TO 100
      IF(R(I) .GT. CLDMAX) CLDMAX=R(I) ! FIND MAX CLD WATER AT TOP
100  FLOATL=FLOATL+POUT(I) ! PRECIP DUMPED OUT
C
      IF(CLDMAX .LT. 3.5) GO TO 1200 ! KEEP CLD WATER < 4. AT TOP
      WRITE(6,807) CLDMAX
807  FORMAT(1H ,' UNREALISTIC CLOUD WATER REQUIRED, CLDMAX=: ',1PE
110.3)
      GO TO 1501
C
      FORMAT(1H ,I3,5(1X,1PE10.3))
C
1200 IF(FLOATL .GT. 50.) GO TO 1304 ! TOO MUCH REMOVED, REDUCE W
C
      REWIND 11 ! SAVE UPDRAFT DATA
      WBAR=DATA(11) ! FOR RESTARTING WINDUP
      WRITE(11) DATA(5),DATA(3),EI,EW,BPNO,H
      WRITE(11) DATA(11),HF,ZBASE,RHG
      WRITE(11) R,RC,ALPH,DEPOS,W,PRATE,FALL,AUTO
      1 ,FEN_UP,FUP_EN,FUP_DN
C
1350 NOPRE=0
      IFLG=0
      DATA(11)=.01 ! INITIAL GUESS FOR WBAR IN RAINSHAFT
      RHG=1. ! SURFACE RH IN RAINSHAFT
      ZB=0. ! RAINSHAFT CLOUD BASE
C

```



```

C ... LOOP 1400 ITERATES TO FIND VERTICAL VELOCITY CORRESPONDING
C   TO DESIRED SURFACE PRECIPITATION RATE, XJO.
C
1401 DO 1400 J=1,20
      DATA(7)=XJO ! INITIAL RATE FROM WETUP, MUST RESET BECAUSE
C           WINDUP KEEPS CHANGING TO ACTUAL RATE
C
      CALL WINDUP(E,RHG,XX,IFLG,T,DT,WMAX) ! GET RAINSHAFT WINDS
C
C ... 'WATER3' COMPUTES PROFILES OF CONDENSED WATER IN RAINSHAFT
      CALL WATER3(RHG,IFLG,RAYICE,XJ1)
      IFLG=IFLG+1
C
      IF(IFLG .EQ. 100) GO TO 1401 ! SPLIT FLOW IN WATER3
C
      IF(XJ1 .EQ. 0.) GO TO 1304 ! NO-SOLUTION FROM WATER3
C           INCREASE WINDS & E
C
C ... HERE XJ1 IS SURFACE PRECIP COMPUTED WITH CURRENT GUESS OF W,
C   AND DATA(7) IS THE DESIRED VALUE (SET BY WINDUP).
C
C ... 'NEUTON' ADJUSTS W (DATA[11]) SO THAT DIFF1=0
      DIFF1=DATA(7)-XJ1
      IF(IFLG .EQ. 1 .AND. DIFF1 .GT. 0.) GO TO 1402 ! IF TRUE,
C   FIRST GUESS DOESN'T PRODUCE ENOUGH PRECIP, INCREASE UPDRAFT.
      CALL NEUTON(DIFF1,DATA(11),XLIM,J,IFLAG)
      IF(IFLAG .EQ. 1) GO TO 1300 ! SOLUTION IF TRUE
C
      IF(DATA(11) .GT. .001 ) GO TO 1400
C
C ... IF HERE, TO OBTAIN DESIRED PRECIP RATE, W NEEDS TO BE APPROX=0.
C   CORRECT BY INTENSIFYING UPDRAFT AND DUMPING MORE PRECIP INTO
C   RAINSHAFT.
C

```

```

1402 T=T-DT
      GO TO 1500 ! RECOMPUTE UPDRAFT WITH LARGER PRECIP RATE
C
801  FORMAT(1H ,1X,I3,6(1X,1PE10.3))
802  FORMAT(1H0)
800  FORMAT(1H ,3(1X,I3),5(1X,1PE10.3))
C
1400 CONTINUE
      GO TO 1501
C
1502 WRITE(IOUT,820) ! HERE FROM UPDRAFT CALCULATIONS
820  FORMAT(1H0,' MAX WINDS TOO LOW, DECREASE EFFICIENCY')
      E=E-0.1
      WRITE (IOUT,817) E
      IF(E .LT. 0.1) GO TO 1501 ! NO SOLUTION
      GO TO 1305 ! START OVER
C
1306 WRITE(IOUT,825)
825  FORMAT(1H , ' MAX W TOO LARGE, INCREASE E OR THRESH')
C
C ... EFFICIENCY TOO LOW, MUST INCREASE AND START OVER
1304 E=E+0.1 ! HERE FROM RAINSHAFT CALCULATIONS
      WRITE(IOUT,817) E
817  FORMAT(1H , ' EFFICIENCY CHANGED TO =',1PE10.3)
      IF(E .LT. 1.) GO TO 1305
      THRESH=THRESH+2. ! ALLOWS UPW TO INCREASE
      E=0.9 ! MAKE IT EASY, KEEP E HIGH
      IF(THRESH .GT. 12.) GO TO 1501 ! NO SOLUTION
C
1305 DT=5.
      T=60.
C
1500 CONTINUE
C

```

```

C ... IF M=31, CONVERGENCE HAS'NT BEEN ACHIEVED
      WRITE(IOUT,814)
814  FORMAT(1H0,' NO CONVERGENCE IN MAIN')
      WRITE(IOUT,801) M,DIFF1,XJ1
      IF(ABS(DIFF1) .LT. 5.E-3) GO TO 1300  ! NO DEFAULT IF TRUE
C
1501 WRITE(IOUT,803)
803  FORMAT(1H0,' NO SOLUTION IN MAINUP, DEFAULT USED')
      WRITE(IOUT,824) M,J,E,THRESH,FLOATL
824  FORMAT(1H , ' M,J,E,THRESH,FLOATL=',2(I3,1X),3(1X,1PE10.3))
      CALL DEFAULT(3)  ! ONE OF ABOVE IS OVER LIMIT
      WRITE(10) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO
      1 ,FEN_DN,FEN_EN,FEN_UP
      CALL DEFAULT(1)
      WRITE(9) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO
      1 ,FEN_UP,FUP_EN,FUP_DN
      RETURN
C
C ... ALL OUTPUT BELOW HERE
C
1300 IF(IFLG .NE. 100) CALL UNITS  ! CONVERTS UNITS FOR RAINSHAFT
      IF(IFLG .NE. 100) WRITE(IOUT,815) XJ0
815  FORMAT(1H0,' STORM PRECIPITATION RATE-HOURLY AVERAGE (MM/HR)
1  ',1PE10.3)
      IF(IFLG .NE. 100) WRITE(IOUT,816) T,DATA(7)
816  FORMAT(1H0,' STORM DURATION (MIN)=' ,1PE10.3,' STORM PRECIP RATE
1 (MM/HR)=' ,1PE10.3)
      WBAR=DATA(11)
      IF(IFLG .EQ. 100) XJ0=DATA(7)  ! FOR RETURN TO 'WETUP'
      IF(IFLG .NE. 100) WRITE(6,805)
      IF(IFLG .EQ. 100) WRITE(6,806)
805  FORMAT(1H0,'          RAINSHAFT-CELL DATA ')
806  FORMAT(1H0,'          UPDRAFT-CELL DATA ')
C

```

```

      WRITE(IOUT,821) H,ZB,HF,DATA(7),TEMP(1),RHG,DATA(5),EW,EI/0.9
821  FORMAT(1H0,' CELL HEIGHT (M)=' ,1PE10.3,'      CLOUD BASE
      1HEIGHT (M)=' ,1PE10.3,/,' FREEZING LEVEL HEIGHT (M)=' ,1PE10.3
      2,/,' SURFACE PRECIP
      3ITATION RATE (MM/HR)=' ,1PE10.3,/,' SURFACE TEMPERATURE (K)=
      4' ,1PE10.3,/,' SURFACE HUMIDITY=' ,1PE10.3,'      STORM EFFICIE
      5NCY=' ,1PE10.3,/,' WATER COLLECTION EFFICIENCY=' ,1PE10.3,/,'
      6 ICE COLLECTION EFFICIENCY=' ,1PE10.3)
      DZ=L-1
      DZ=H/DZ
      WRITE(IOUT,802)
      WRITE(IOUT,813)
813  FORMAT(1H0,2X,'GRID  CLOUD      PRECIP      SCAV      DEPOS-EVAP
      1  UPDRAFT')
      WRITE(IOUT,810)
810  FORMAT(1H ,7X,'WATER      WATER      COEF      COEF
      1SPEED')
      WRITE(IOUT,822)
822  FORMAT(1H ,7X,' (G/G)      (G/G)      (1/SEC)      (1/S)
      1  (CM/S)' ,/)
C
      DO 600 I=1,1,2
600  WRITE(6,801)I,R(I),RC(I),ALPH(I),DEPOS(I),W(I)
      WRITE(IOUT,802)
      Z=-DZ/2.
C ... Z IS THE HEIGHT AT CENTER OF GRID
      WRITE(IOUT,811)
811  FORMAT(1H0,2X,'GRID  HEIGHT      PRECIP      FALL      AUTO')
      WRITE(IOUT,812)
812  FORMAT(1H ,'      RATE      SPEED')
      WRITE(IOUT,823)
823  FORMAT(1H ,9X,'(M)      (MM/HR)      (CM/S)      (1/S) ')
      WRITE(IOUT,804)
804  FORMAT(1H )

```

```

DO 1503 I=1,1,2
Z=I
Z=Z-0.5
Z=DZ*Z
1503 WRITE(IOUT,801) I,Z,PRATE(I),FALL(I),AUTO(I)
WRITE(IOUT,802)
IF(IFLG .EQ. 100) RETURN
IFLG=100
REWIND 11 ! COPY UPDRAFT DATA
READ(11) DATA(5),DATA(3),EI,EW,BPNO,H
READ(11) DATA(11),HF,ZBASE,RHG
C
CALL DEFALT(2)
WRITE(10) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO !RAINSHAFT
1 ,FEN_DN,FEN_EN,FEN_UP
READ(11) R,RC,ALPH,DEPOS,W,PRATE,FALL,AUTO
1 ,FEN_UP,FUP_EN,FUP_DN
CALL UNITS ! CONVERTS UNITS FOR UPDRAFT
CALL DEFALT(0)
WRITE(9) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO !UPDRAFT
1 ,FEN_UP,FUP_EN,FUP_DN
HF=HFU ! UPDRAFT OUTPUT
ZB=ZBASE ! UPDRAFT CLOUD BASE
GO TO 1300 !OUTPUT UPDRAFT DATA
END

```

C

SUBROUTINE WATER3(RHG,IFLG,RAYICE,XJ1)

C

C ... FOR RAINSHAFT, CONVECTIVE CLOUD ...

C.....

C

```

C
C ... CONVECTIVE CLOUD MODULE
C.....
C
C ... RELEASED NOV, 1986
C
C ... FOR INFORMATION, CONTACT:
C     B.C. SCOTT OR C.M. BERKOWITZ
C     PACIFIC NORTHWEST LABORATORY
C     P.O. BOX 999, RICHLAND, WA. 99352
C
C.....
C
C     INCLUDE 'SCAV.FOR'
C
C.....
C
C ... WATER3 COMPUTES PROFILES OF PRECIPITATION CONCENTRATION,
C     PRECIPITATION RATE, PARTICLE FALL SPEED, AND SCAVENGING
C     COEFFICIENTS FOR RAINSHAFT.
C
C.....
C
C ... VARIABLES IN CALL LIST:
C     RHG=SURFACE RELATIVE HUMIDITY (DECIMAL NOT %)
C     IFLG= COUNTER FOR DETERMINING 1ST AND SUCCESSIVE TIMES
C           THROUGH WATER3
C     XJ1=COMPUTED SURFACE PRECIPITATION RATE (MM/HR)
C     RAYICE= AVERAGE RAYICE FROM WATER4 (UPDRAFT) FOR Z>HF
C
C.....

```

```

C
C ... FLAGS:
C     IFLG= 0 WHEN CLOUD AND PRECIP PROFILES NEED TO BE INITIALIZED
C     = 1 WHEN INITIALIZATION ALREADY DONE
C
C .....
C
C     DIMENSION XLIM(7),AXIS(75),VALUE(75),FLUX(L)
C     DIMENSION RMEAN(L),SUMD(L),SUMC(L),PX(L),PXI(L),GX(L),SRC1(L)
C     XLIM(3)=1.E-12 ! CONVERGENCE LIMITS FOR NEUTON
C     XLIM(4)=7.E-6 ! CONVERGENCE CRITERIA FOR NEUTON
C
C .....
C     AXNO=VALUE OF  $E \cdot N_0^{**0.22}$  FOR RAIN
C     BPNO=VALUE OF  $N_0$  FOR SNOW
C     EW=WATER COLLECTION EFFICIENCY.
C     EI=ICE COLLECTION EFFIC.
C     HF=FREEZING LEVEL HEIGHT (M)
C     R(I)=CONCENTRATION OF CLOUDWATER AT LEVEL I (G/M3)
C     RC(I)=CONCENTRATION OF PRECIP WATER AT LEVEL I (G/M3)
C     ZZBASE=CLOUD BASE (M)
C     H=STORM HEIGHT (M)
C     XJ=PRECIPITATION RATE (MM/HR)
C     L=NUMBER OF VERTICAL GRIDS (DEFINED IN SCAV.FOR)
C .....
C
C     ZZBASE=1. ! CLOUD BASE AT GROUND
C     HF=HFD ! RAINSHAFT FREEZING LEVEL, DETERMINED IN WETUP.
C     BBP=BPNO**0.125
C
C ... UNLOAD DATA ARRAY
C     H=DATA(1) ! CLOUD TOP HEIGHT (M)
C     P=DATA(8) ! CRYSTAL CONC. (1/M3) FROM 'WETUP'
C     XJ=DATA(7) ! PRECIP RATE (MM/H)

```

```

C
C ... COMPUTE GRID SPACING (DZ)
      LL=L-1
      DZ=H/FLOAT(LL)

C
C ... CAN SKIP TO 110 IF PROFILES INITIALIZED ON FIRST PASS
      IF(IFLG .GT. 0) GO TO 110

C
C ... RNO AND RNO2 ARE No**.125 AND No**.250, RESP.
      RND=(AXNO/EW)**0.5625
      RNO2=RNO*RNO
      CONST1=38.8496*RAYICE/RNO  ! FALL SPEED LARGE GRAUPEL
      CONST2=385.267*RAYICE/RNO2 ! FALL SPEED SMALL GRAUPEL

C
C ... INITIALIZE CLOUD AND PRECIP PROFILES
C
C ... LOOP 100 COMPUTES FIRST GUESS OF PRECIPITATION PROFILE (RC)
      IBASE=1  ! CLOUD BASE ALWAYS AT THE GROUND
      Z=-DZ
      DO 100 I=1,L
        Z=Z+DZ
        RMEAN(I)=0.
        R(I)=0.
        RCS(I)=0.

C
C ... INITIALLY ASSUME PRECIP RATE VARIES LINEARLY FROM CLOUD BASE TO
C   CLOUD TOP.
C
      RC(I)=XJ*(1.-(Z-ZZBASE)/(H-ZZBASE))
      IF(RC(I) .LT. 0.) RC(I)=0.

C ... FROM PRECIP RATE DATA, COMPUTE PRECIP CONCENTRATION
      IF(Z .LT. HF) GO TO 102
      RC(I)=RC(I)**0.936*0.32*BBP
      GO TO 100

```



```

102  RC(I)=RC(I)**0.88889*1.238E-2*(AXNO/EW)**0.5
100  CONTINUE
C
C      *****DEPOSITIONAL ICE GROWTH CALCULATIONS*****
C      *****CONDENSATION CALCULATIONS*****
C
C ... LOOP 300 COMPUTES THE CONDENSATION RATE (SX--G/M3/S) AS A FUNCTION
C      OF Z AND STORES IT IN SORCE
      DO 300 I=2,LL
        SXI=ALOG(RHO(I+1)/RHO(I-1))/4.
        SX=VAP(I)-VAP(I-1)-(VAP(I)+VAP(I-1))*SXI
        SXI=VAPI(I)-VAPI(I-1)-(VAPI(I)+VAPI(I-1))*SXI
        SX=-SX
        SXI=-SXI
        IF(SX .LT. 0.) SX=1.E-8
        IF(SXI .LT. 0.) SXI=SX
        PX(I)=SX
300   PXI(I)=SXI
C
110   Z=-DZ
      D=0.
      DO 325 I=1,LL
        KK=I+1
        Z=Z+DZ
        IF(RC(I) .EQ. 1.E-7) RC(I)=1.1E-7 ! CHANGE STARTING GUESS
        V=-2.
        CALL COEF(I,DZ,Z,D,V,A1,GX(I)) ! BLOWOUT WINDS, GX
        SORCE(I)=(PX(I)*W(I)+PX(KK)*W(KK))/2.
325   SORCEI(I)=(PXI(I)*W(I)+PXI(KK)*W(KK))/2.
C
C ... DEPOSITIONAL GROWTH RATES MUST BE DETERMINED ONLY 1ST TIME THROUGH
C
      IF(IFLG .GT. 0 .AND. IFLG .NE. 100 ) GO TO 351
C

```

```

C ... LOOP 350 COMPUTES PART OF THE DEPOSITIONAL GROWTH RATE.
C
      XNC=P**0.621*19.53*DZ
      Z=-DZ/2.
      DO 350 I=1,LL
      Z=Z+DZ
      IF(W(I) .LE. 0.) GO TO 302
      IF(Z .GT. HF) GO TO 303  ! ICE GROWTH ONLY ABOVE FREEZ. LVL.
302   SDEP(I)=0.
      GO TO 350
C
C ... HERE SDEP IS PART OF THE DEPOSITIONAL GROWTH RATE
C                                     (G/M2/S * M/S)
C   IT NEEDS TO BE DIVIDED BY THE PRECIP FALL SPEED TO GIVE A
C   GROWTH RATE.
C
303   T=ABS(TEMP(I)-273.16)  ! ABS VALUE OF T (C)
      S=1.
      IF(T .LT. 10.) S=0.98*T**0.5
      IF(T .LT. 20. .AND. T .GE. 10.) S=3.2
      IF(T .LT. 40. .AND. T .GE. 20.) S=5.4-0.11*T
      SDEP(I)=XNC*S*1.E-8
350   CONTINUE
C
C   END OF ICE GROWTH CALCULATIONS
C
C ... LOOP 2000 ITERATES TRYING TO BALANCE CLOUD WATER AND PRECIP WATER
C   BUDGETS.
351   DO 2000 KL=1,30
C
C   *****CLOUD WATER CALCULATIONS*****
C
C ... COMPUTE CLOUD WATER CONCENTRATION (R)
C

```

```

C
      IF( BARM2 .GT. 0.) BARS=BARM2
C
C ... LOOP 200 PROCEEDS FROM GROUND UPWARD TO CALCULATE CLOUD WATER (R)
C   AS A FUNCTION OF CURRENT PRECIP WATER CONTENT (RC)
      Z=-DZ
      DO 200 I=1,LL
      KK=I+1
      Z=Z+DZ
C
C ... SUBTRACT DEPOSITIONAL GROWTH FROM AVAILABLE WATER.  REMAINING
C   CONDENSED WATER GOES INTO CLOUD WATER.
      SRC1(I)=SORCE(I)  ! CONDENSATION SOURCE
      IF(RMEAN(I) .LE. 0. .AND. Z .GT. HF) SRC1(I)=SORCEI(I) ! ICE GROWTH ONLY
      SRC1(I)=SRC1(I)+COUT(I)*0.5  ! INCLUDE UPDRAFT BLOWOUT
C
      IF(KL .EQ. 1) GO TO 202 ! 1ST TIME, NO CHANGE IN INITIAL R
      IF(W(KK) .GT. 0.) GO TO 210 ! WATER SOURCE IN UPDRAFT
      IF(COUT(I) .LE. 0.) GO TO 202 ! DOWNDRAFT, NO SOURCE, R=0
C
210   FR=SRC1(I)-SUMD(I)  ! =0 IF SRC1(I) < 0
C
C ... COMPUTE AUTOCONVERSION COEFFICIENT
      AUTO(I)=0.
      RBAR=RMEAN(I)
      IF(RBAR .LE. 0.) GO TO 211
      XXC=RBAR/DROPS
C
C ... THRESHOLD FOR AUTOCONVERSION (MEAN RADIUS=8.75 MICRONS)
      IF(XXC .LT. 2.348E-9) GO TO 211
      AUTO(I)=5.5E+9*RBAR*RBAR*XXC**1.667*(1.-(2.35E-9/XXC)**1.33)
      1 *(1.-(2.04E-9/XXC)**0.333)*DZ
      IF(AUTO(I) .LT. 0.) AUTO(I)=0.

```

```

        IF(AUTO(I) .GT. FR) AUTO(I)=FR
        FR=FR-AUTO(I)
C
211   FRR=GX(I)+SUMC(I)/2.
        R(KK)=(R(I)*(W(I)-FRR)+FR)/(W(KK)+FRR)
        IF(R(KK) .LT. 0.) R(KK)=0.
C
C ... IN ORDER TO REDUCE UNSTABLE WAVES RESULTING FROM ITERATION, THE
C     CURRENT VALUE OF MEAN WATER CONCENTRATION IS ACTUALLY THE AVERAGE OF
C     THE RECENTLY COMPUTED AND THE PREVIOUS AVERAGE VALUE.
C
202   RMEAN(I)=(R(I)+R(KK))/2. ! NECESARY IN PRECIP CALC.
        IF(KL .GT. 2 ) RMEAN(I)=(RCS(I)+RMEAN(I))/2.
        IF(RMEAN(I) .LT. 1.E-5) RMEAN(I)=0.
        RCS(I)=RMEAN(I)
200   CONTINUE ! CLOUD WATER LOOP
C
C
C     *****PRECIPITATION WATER CALCULATION*****
C
C
212   RSAV=0. ! VALUE OF RC AT BIN TOP
        RC(L)=0. ! UPPER BOUNDARY CONDITION
        Z=H+DZ/2.
        BARM2=0. ! ACCUMULATES TOTAL PRECIP WATER
C
C ... LOOP FROM CLOUD TOP TO GROUND TO COMPUTE PRECIPITATION WATER
C     CONCENTRATION AS A FUNCTION OF CURRENT CLOUD WATER CONCENTRATION
C
        DO 250 K=1,LL
            ISTART=2 ! IF=2, NO ZERO SEARCH IN BALANCE EQTN
            I=L-K
            Z=Z-DZ ! INTEGRATING FROM TOP DOWN
            FLUXIN=FLUX(I+1)*RC(I+1) ! FLUX IN BIN TOP

```

```

C
1999 KUP=1 ! COUNTER FOR ZERO SEARCH
      IF(ISTART .EQ. 1) RC(I)=1.E-4 ! STARTING VALUE OF RC IN ZERO SEARCH
C
      DO 3000 IVALUE=ISTART,2 ! SEARCH FOR ZERO IF ISTART = 1
C
C ... LOOP 260 ITERATES TO ADJUST RC(I) SUCH THAT A MASS FLUX EQUATION
C      IS BALANCED.
3001 DO 260 J=1,15 ! ITERATE TO FIND PROPER RC AT CURRENT VALUE OF RMEAN
      RPBAR=(RSAV+RC(I))/2.
      DMDT=0.
      C=0.
C
C ... COMPUTE SCAVENGING COEF. (C--1/S) AND FALL SPEED (VP) FOR PRECIP
C
C ... GRAUPEL FALL SPEEDS
      IF(RC(I) .GT. 0.11) VP=RC(I)**0.125*CONST1
      IF(RC(I) .LE. 0.11) VP=CONST2*RC(I)**0.25
      IF(Z-DZ*0.5 .LT. HF) GO TO 251 ! IF TRUE, USE RAIN EQTNS
      IF(RPBAR .GT. 0.11) VPBAR=RPBAR**0.125*CONST1 ! GRAUPEL
      IF(RPBAR .LE. 0.11) VPBAR=CONST2*RPBAR**0.25 ! GRAUPEL
C
      IF(RPBAR .GT. 0.) C=7.008E-4*BBP*EI*RAYICE/RPBAR**0.125 ! ICE
C
C ... 'VDEP' ACCOUNTS FOR A PSEUDO-DIFFUSION ARISING FROM A VARIATION
C      IN FALL SPEED WITH PARTICLE SIZE.
      VD=0.
      IF(VP .GT. 0.) CALL VDEP(1,DZ,BPNO,VP,I,VO)
      IF(ABS(VD) .GT. VP) VD=0 ! NEGLECT IF UNREASONABLE
C
C ... INCLUDE CONTRIBUTION OF DEPOSITIONAL MASS TO TOTAL MASS.
      VBAR=VPBAR-(W(I)+W(I+1))/2.
      IF(VBAR .LT. 0.1) VBAR=0.1
      OMOT=((RC(I+1)**0.62111802+SDEP(I)/VBAR)**1.61-RC(I+1))*VBAR

```

```

C
C ... LIMIT DEPOSITIONAL GROWTH RATE TO WATER CONDENSATION RATE
C   IF CLOUD WATER EXISTS, OTHERWISE LIMIT TO ICE CONDENSATION
C   RATE.
C
251  DM=SRC1(I)-DMDT ! INCLUDE UPDRAFT BLOWOUT AS SOURCE
      IF(DM .LT. 0.) DMDT=SRC1(I) ! GENERALLY TRUE IF W < 0
      IF(Z-DZ/2. .GE. HF) GO TO 253
C
      S6=0. ! IF HERE, Z IS BELOW HF
C
C ... SEARCH FOR THE FREEZING LEVEL
      IF(Z+DZ*.5 .LT. HF) GO TO 2252
C ... FREEZING LEVEL IS IN INTERVAL Z TO Z+DZ
C
C ... S REPRESENTS THE PRECIP CONCENTRATION AT THE FREEZING LEVEL
C   RPBAR IS THE HEIGHT WEIGHTED AVERAGE PRECIP IN THE INTERVAL
C   DZ.
C
      S=RC(I+1)+(RC(I+2)-RC(I+1))*(HF-Z-DZ/2.)/DZ
      RPBAR=((S+RC(I))*(HF-Z+DZ/2.)+(RC(I+1)+S)*(Z+DZ/2.-HF))/2./DZ
      IF(RPBAR .LT. 0.) RPBAR=0.
C
C ... USE AVERAGE OF ACCRETION COEFFICIENTS NEAR FREEZING HT
      C1=7.008E-4*RNO*EW*RC(I)**(-0.125)
      C2=0
      IF(RC(I+1) .GT. 0.) C2=7.008E-4*BBP*E1*RAYICE/RC(I+1)**0.125
      S6=(C1*RC(I)+C2*RC(I+1))/2.*0Z
      GO TO 2251
C
2252 IF(RPBAR .GT. 0.) C=7.008E-4*RNO*EW/RPBAR**0.125 ! C FOR RAIN
2251 VP=VP/RAYICE ! RAIN RATES
      VD=0.
      IF(VP .GT. 0.) CALL VDEP(2,DZ,RN02,VP,I,VD)

```

```

        IF(ABS(VD) .GT. VP) VD=0  ! NEGLECT IF UNREASONABLE
        IF(S6 .GT. 0.) GO TO 255

C
253   S6=C*DZ*RPBAR
255   C=S6*RMEAN(I)
C
C ... DEPOSITIONAL GROWTH PLUS OTHER SOURCES
      S3=DMDT+POUT(I)+COUT(I)*0.5  ! SOME CLOUD WATER ADDED ALSO
C ... HORIZONTAL PRECIPITATION FLUX TERM
      S5=RPBAR*GX(I)*2
C ... ACCRETION PLUS AUTOCONVERSION TERM
      SY=C+AUTO(I)
C ... FLUX INTO BIN TOP
      FLUX(I)=W(I)-VP-VD
C
C ... 'NEUTON' ADJUSTS RC(I) SUCH THAT DIFF1=0
      DIFF1=-FLUXIN+SY+S3-S5+FLUX(I)*RC(I)
      IF(IVALUE .EQ. 1) GO TO 270  ! IN SEARCH FOR ZERO
C
C
      SUMC(I)=S6  ! FOR CLOUD WATER EQTN
      SUMD(I)=DMDT  ! FOR CLOUD WATER EQTN
      IF(RC(I).EQ.1.E-7) GO TO 2552  ! RESULTS FROM
C          SEARCH FAILURE
      CALL NEUTON(DIFF1,RC(I),XLIM,J,IFLAG)
      IF(IFLAG .EQ. 1) GO TO 270  ! CONVERGENCE , IF TRUE
C
      IF(J .LT. 12) GO TO 260  ! LOOK FOR ZERO IF NO CONVERG AFTER 12 TRIES
      ISTART=1  ! CHECK TO SEE IF ANY SOLUTION EXISTS
      GO TO 1999
260   CONTINUE
C
270   IF(IVALUE .EQ. 2) GO TO 2552  ! IF TRUE, HAVE FOUND SOLUTION
C

```

```

        CALL ZERO1(30.,RC(I),DIFF1,VAL,KUP,JFLG) ! SEARCH FOR SOLTN
        GO TO (3001,3000,3002) JFLG
3002  WRITE(6,940) I,VAL
940   FORMAT(1H ,' NEAR 3002 WATER3, NO ZERO ,LEVEL,DIFF=',I3
        1,1X,1PE10.3)
        RC(I)=1.E-7
        IF(VAL .LT. 0.) GO TO 3000 ! TRY NEGLIGIBLE RC AT LEVEL I
        XJ1=0.
        RETURN
C
3000  CONTINUE ! GO TO TOP WITH NEW INITIAL GUESS NEAR THE ZERO
C
2552  R SAV=RC(I)
        ALPH(I)=SUMC(I)/DZ
        DEPOS(I)=SUMD(I)/DZ
        BARM2=BARM2+RC(I) ! TOTAL PRECIP WATER
C
C ... SAVE FALLSPEED (VP-M/S) AND PRECIP RATE (PRATE--G/M2/S)
        FALL(I)=VP
        PRATE(I)=-R SAV*FLUX(I)
        IF(PRATE(I) .GE. 0.) GO TO 250 ! IF TRUE, PRECIP IS FALLING DOWN
        WRITE(6,801) I,J,RC(I),W(I),VP,VD
801   FORMAT(1H ,' SPLIT FLOW, TRY SUBSIDENCE IN ENTIRE RAINSHAFT',/,
        1' I,J, RC(I),W(I),VP,VD,= ',2(1X,I3),6(1X,1PE10.3))
        IFLG=99 ! TRY SUBSIDENCE IN ENTIRE RAINSHAFT
        RETURN
C
250   CONTINUE ! COMPUTE AT NEXT LOWER LEVEL-LOOP
C
C ... CHECK FOR CONVERGENCE. IS THE TOTAL PRECIP WATER (BARM2) THE
C     SAME THIS ITERATION AS IN THE PAST ITERATION?
C
C ... GO AT LEAST 3 TIMES BEFORE A CONVERGENCE CHECK IS MADE
        IF(KL .LT. 3) GO TO 2000

```



```

        IF(ABS(BARS) .LT. 1.E-9) GO TO 2002
        BARS=(BARS-BARM2)/BARS
        IF(ABS(BARS) .LT. 5.E-5) GO TO 2002
C
C ... IF HERE, NO CONVERGENCE. TO REDUCE UNSTABLE WAVES IN VERTICAL
C PROFILE, USE, FOR THE NEW VALUE OF PRECIP CONCENTRATION, THE
C AVERAGE OF THE RECENTLY COMPUTED PRECIP WATER AND THE PREVIOUS
C AVERAGED VALUE OF PRECIP WATER.
C
        DO 405 I=1,L
        IF(KL .EQ. 3) GO TO 406
        R1=RS(I)
        RC(I)=(R1+RC(I))/2.
406    RS(I)=RC(I)
405    CONTINUE
C
2000    CONTINUE ! RECOMPUTE R & RC IN LOOP
C
        WRITE(6,950) I,KL ! NO CONVERGENCE
950    FORMAT(1H0,' KL TOO BIG,LOOP 2000,WATER3, I,KL=',2(1X,I3))
        XJ1=0.
        RETURN
C
C ... XJ1 IS THE COMPUTED VALUE OF PRECIP RATE AT THE SURFACE OR
C CLOUD BASE. COMPARE (IN 'MAINUP') WITH THE DESIRED VALUE.
2002    XJ1=PRATE(IBASE)*3.6
        DATA(3)=P/1000. ! FOR OUTPUT IN MAINUP.
C
        DO 475 I=1,L ! UNITS CONVERSION
        PRATE(I)=PRATE(I)*3.6 ! (MM/HR)
475    AUTO(I)=AUTO(I)/DZ ! (G/M3/S)
        RETURN
        END

```

```

C .....

      SUBROUTINE WATER4(RHG,IFLG,RAYAVE,THRESH)
C
C ... UPDRAFT, CONVECTIVE CLOUD ...
C
      INCLUDE 'SCAV.FOR'
C
C
C ... WATER4 COMPUTES PROFILES OF PRECIPITATION CONCENTRATION,
C      PRECIPITATION RATE, PARTICLE FALL SPEED, AND SCAVENGING
C      COEFFICIENTS.
C.....
C
C ... CONVECTIVE CLOUD MODULE
C
C ... RELEASED NOV, 1986
C
C ... FOR INFORMATION, CONTACT:
C      B.C. SCOTT OR C.M. BERKOWITZ
C      PACIFIC NORTHWEST LABORATORY
C      P.O. BOX 999, RICHLAND, WA. 99352
C
C.....
C
C ... VARIABLES IN CALL LIST:
C      RHG=SURFACE RELATIVE HUMIDITY (DECIMAL NOT %)
C      IFLG= COUNTER FOR DETERMINING 1ST AND SUCCESSIVE TIMES
C           THROUGH WATER4
C      RAYAVE=AVERAGE VALUE OF RAYICE ABOVE THE FREEZING LVL.
C      THRESH=MAX ALLOWABLE PRECIP CONCENTRATION IN UPDRAFT-CELL
C
C

```

```

C ... FLAGS:
C     IFLG= 0 WHEN CLOUD AND PRECIP PROFILES NEED TO BE INITIALIZED
C     = 1 WHEN INITIALIZATION ALREADY DONE
C
C ... OTHER DEFINITIONS:
C     AXNO=VALUE OF  $E \cdot N_0^{**0.22}$  FOR RAIN
C     BPNO=VALUE OF  $N_0$  FOR SNOW
C     EW=WATER COLLECTION EFFICIENCY.
C     EI=ICE COLLECTION EFFIC.
C     HF=FREEZING LEVEL HEIGHT (M)
C     R(I)=CONCENTRATION OF CLOUDWATER AT LEVEL I (G/M3)
C     RC(I)=CONCENTRATION OF PRECIP WATER AT LEVEL I (G/M3)
C     ZBASE=CLOUD BASE (M)
C     H=STORM HEIGHT (M)
C     XJ=PRECIPITATION RATE (MM/HR)
C     L=NUMBER OF VERTICAL GRIDS (DEFINED IN SCAV.FOR)
C     QNO,QNO2= $N_0^{**0.125}$  AND  $N_0^{**250}$ , RESPECIVELY
C.....
C
C     DIMENSION XLIM(7),SUMC(L),SUMD(L),FLUX(L),RMAXX(L),GX(L)
C     XLIM(3)=1.E-12
C     ZLIM=XLIM(4)
C
C     BBP=BPNO**0.125
C     HF=HFU ! UPDRAFT FREEZING LEVEL
C
C ... UNLOAD DATA ARRAY
C     H=DATA(1) ! CLOUD TOP (M)
C     P=DATA(8) ! CRYSTAL CONCENTRATION USED BY 'NUKE'
C     XJ=DATA(7) ! PRECIP RATE (MM/HR)
C     SCALE=DATA(12) ! FOR BLOWOUT OF PRECIPITATION
C
C     LL=L-1
C     DZ=H/FLOAT(LL) ! GRID SPACING (M)

```

```

C
C ... CAN SKIP TO 110 IF PROFILES INITIALIZED ON FIRST PASS
      IF(IFLG .GT. 0) GO TO 110
C
C ... INITIALIZE CLOUD AND PRECIP PROFILES
C
      Z=-DZ
      IBASE=1
      QNO=(AXNO/EW)**0.5625 ! No**0.125
      QNO2=QNO*QNO ! No**0.250
      VCONST=QNO/(38.8496*THRESH**0.125) ! FOR RAYICE EVALUATION
      XLEVEL=0. ! FOR AVERAGE RAYICE
      RAYSUM=0. ! FOR AVERAGE RAYICE
C
C ... LOOP 100 COMPUTES FIRST GUESS OF PRECIPITATION PROFILE (RC) AND
C      DETERMINES GRID INDEX OF CLOUD BASE (IBASE) FOR THE UPDRAFT CELL
C
      DO 100 I=1,L
      IF(W(I) .LT. 0.) W(I)=0.
      POUT(I)=0.
      ALPH(I)=0.
      DEPOS(I)=0.
      PRATE(I)=0.
      FALL(I)=0.
      AUTO(I)=0.
      R(I)=0.
      Z=Z+DZ
C ... BELOW CLOUD BASE, CLOUD WATER CONCENTRATION (R) IS ZERO.
C
C ... MAKE INITIAL GUESS EQUAL TO RMAX
      RAYICE=1.0
      IF(Z .LT. HF) GO TO 101
      RAYICE=VCONST*W(I)
      IF(RAYICE .LT. .4) RAYICE=0.4 ! LOWER LIMIT FOR RAYICE

```

```

XLEVEL=XLEVEL+1 ! NUMBER OF RAYICE VALUES ABOVE HF
RAYSUM=RAYSUM+RAYICE ! TO GET RAYICE AVERAGE
101 UPLIM=3.96*RAYICE ! BREAK POINT FOR VP FORMULAS
    IF(W(I) .LT. UPLIM) RC(I)=(W(I)*QNO2/(385.267*RAYICE))**4.
    IF(W(I) .GE. UPLIM) RC(I)=(W(I)*QNO/(38.8496*RAYICE))**8.
    RMAXX(I)=RC(I) ! INITIAL GUESS
C
    SUMC(I)=5.E-4
    SUMD(I)=5.E-4
    IF(Z .GT. ZBASE) GO TO 100
    RC(I)=0.
    SUMC(I)=0.
    IBASE=I+1 ! CLD BASE BETWEEN I & I+1
C
100 CONTINUE
    RAYAVE=RAYSUM/XLEVEL ! AVERAGE RAYICE
C
C ... CALCULATIONS AT CLOUD BASE
    RC(IBASE)=1.E-5 ! G/M3 AT CLD BASE
    IF(ZBASE .LT. HF) RC(IBASE)=0.
C
C *****DEPOSITIONAL ICE GROWTH CALCULATIONS*****
C *****CONDENSATION CALCULATIONS*****
C
C ... LOOP 300 COMPUTES THE CONDENSATION RATE (SX--G/M3/S) AS A FUNCTION
C OF Z AND STORES IT IN SORCE
110 SORCE(1)=0.
    SDRCEI(1)=0.
    DO 300 I=2,LL
        SXI=ALOG(RHO(I+1)/RHO(I-1))/4.
        SX=VAP(I)-VAP(I-1)-(VAP(I)+VAP(I-1))*SXI
        SXI=VAPI(I)-VAPI(I-1)-(VAPI(I)+VAPI(I-1))*SXI
        SX=-SX
        SXI=-SXI

```

```

      IF(SX .LT. 0.) SX=1.E-8
      IF(SXI .LT. 0.) SXI=SX
      SORCE(I)=SX*W(I)
300  SORCEI(I)=SXI*W(I)
C
      D=0.
      DO 325 I=1,LL
      KK=I+1
      V=-2.
      CALL COEF(I,DZ,Z,D,V,A1,GX(I)) ! BLOWOUT WINDS, GX
      SORCE(I)=(SORCE(I)+SORCE(KK))/2.
325  SORCEI(I)=(SORCEI(I)+SORCEI(KK))/2.
C
C ... DEPOSITIONAL GROWTH RATES MUST BE DETERMINED ONLY 1ST TIME THROUGH
C
C
C ... LOOP 350 COMPUTES PART OF THE DEPOSITIONAL GROWTH RATE.
C
      XNC=P**0.621*19.53*DZ
      Z=-DZ/2.
      DO 350 I=1,LL
      Z=Z+DZ
      IF(W(I) .LE. 0.) GO TO 302
      IF(Z .GT. HF) GO TO 303 ! ICE GROWTH ONLY ABOVE FREEZ. LVL
302  SDEP(I)=0.
      SUMD(I)=0.
      GO TO 350
C
C ... HERE SDEP IS PART OF THE DEPOSITIONAL GROWTH RATE
C                                     (G/M2/S * M/S)
C   IT NEEDS TO BE DIVIDED BY THE PRECIP FALL SPEED TO GIVE A
C   GROWTH RATE.
C
303  T=ABS(TEMP(I)-273.16) ! ABS VALUE OF TEMP (C)

```

```

S=1.
IF(T .LT. 10.) S=0.98*T**0.5
IF(T .LT. 20. .AND. T .GE. 10.) S=3.2
IF(T .LT. 40. .AND. T .GE. 20.) S=5.4-0.11*T
SDEP(I)=XNC*S*1.E-8
350 CONTINUE
C
C   END OF ICE GROWTH CALCULATIONS
C
C
C ... LOOP 2550 PROCEEDS FROM GROUND UPWARD TO CALCULATE CLOUD WATER (R)
C   AS A FUNCTION OF CURRENT PRECIP WATER CONTENT (RC)
C
Z=-DZ
DO 2550 I=1,L ! STEP UP THROUGH GRID
RAYICE=VCONST*W(I)
IF(RAYICE .LT. 0.4) RAYICE=0.4 ! LOWER LIMIT FOR RAYICE
UPLIM=3.96*RAYICE
Z=Z+DZ
IF(Z .LT. HF) RAYICE=1.0
KK=I-1
RSAV=RC(KK) ! VALUE FROM LOWER BIN
1999 KUP=1
RSOL=0. ! SAVES RC VALUE FOR ZERO IN ZERO SEARCH
IF(I .GT. IBASE) RC(I)=1.E-4 ! STARTING VALUE IN ZERO SEARCH
IF(I .EQ. L) RC(L)=0. ! UPPER BOUNDARY CONDITION
C
IF(RC(KK) .EQ. RMAXX(KK)) RSAV=0.
DO 3000 IVALUE=1,2 ! SEARCH FOR ZEROS IN FLUX EQTN
C
S6=0.
DMDT=0.
PDU1=0. ! SET FOR CLD BASE CALCS
C

```

```

3001 DO 2000 KL=1,60 ! ADJUST R AND RC TO BALANCE
C
C *****CLOUD WATER CALCULATIONS*****
C
      IF(I .LT. IBASE) GO TO 2550
      IF(I .EQ. 1) GO TO 211 ! CLOUD BASE AT GROUND
C
C ... CLOUD WATER IS ZERO BELOW CLOUD BASE.
C
C ... COMPUTE AUTOCONVERSION COEFFICIENT
      AUTO(KK)=0.
      RBAR=(R(I)+R(KK))/2.
      IF(RBAR .LE. 0.) GO TO 200 ! SKIP AUTOCONVERSION IF TRUE
      XXC=RBAR/DROPS
C
C ... THRESHOLD FOR AUTOCONVERSION (MEAN RADIUS=8.75 MICRONS)
      IF(XXC .LT. 2.348E-9) GO TO 200
      AUTO(KK)=5.5E+9*RBAR*RBAR*XXC**1.667*(1.-(2.35E-9/XXC)**1.33)
1 * (1.-(2.04E-9/XXC)**0.333)*DZ
      IF(AUTO(KK) .LT. 0.) AUTO(KK)=0.
C
200 BARS=BARM2
      FR=SORCE(KK)
      IF(I .EQ. IBASE) FR=FR/2.
      FR=FR-SUMD(KK)-AUTO(KK)
      DEPOS(KK)=SUMD(KK)/DZ
      FRR=GX(KK)+SUMC(KK)/2.
      R(I)=(R(KK)*(W(KK)-FRR)+FR)/(W(I)+FRR) ! CLOUD WATER CONC.
      RBAR=(R(KK)+R(I))/2.
      IF(RBAR .LT. 0.) RBAR=R(KK)/2.
      COUT(KK)=RBAR*FUP_DN(KK)/SCALE !FLUX MULT BY DZ/XX
      IF(R(I) .LT. 0.) R(I)=0.
C

```



```

C ... IN ORDER TO REDUCE UNSTABLE WAVES RESULTING FROM ITERATION, THE
C     CURRENT VALUE OF WATER CONCENTRATION IS ACTUALLY THE AVERAGE OF
C     THE RECENTLY COMPUTED AND THE PREVIOUS AVERAGE VALUE.
C
C     IF(KL .GT. 2 ) RBAR=(RCS(I)+RBAR)/2.
C     IF(RBAR .LT. 1.E-5) RBAR=0.
C     RCS(I)=RBAR
C
C
C     *****PRECIPITATION WATER CALCULATION*****
C
C ... COMPUTE PRECIPITATION WATER CONCENTRATION AS A FUNCTION OF
C     CURRENT CLOUD WATER CONCENTRATION
C
210  DMDT=0.
C     IF(I .GT. IBASE) GO TO 280
C
211  VP=385.267/QN02*RC(I)**0.25*RAYICE  ! CLOUD BASE VALUES
C     FLUX(I)=W(I)-VP
C     RSAV=RC(I)
C     GO TO 270  ! CALC CLD WATER VALUES AT CLD BASE
C
280  SUMC(KK)=0.
C     SUMD(KK)=0.
C     FLUXIN=FLUX(I-1)*RC(I-1)
C
C     RMAX=RMAXX(I)
C     DO 260 J=1,100  ! COMPUTE RC AT GIVEN R
C     POUT1=0.
C     RPBAR=(RSAV+RC(I))/2.
C     IF(RC(I) .GT. THRESH) RPBAR=(THRESH+RSAV)/2.  ! LIMIT RPBAR
C     DMDT=0.
C     C=0.
C

```

```

C ... COMPUTE SCAVENGING COEF. (C--1/S) AND FALL SPEED (VP) FOR PRECIP
C
C ... GRAUPEL FALL SPEEDS
C
  IF(RC(I) .GT. 0.11) VP=RC(I)**0.125/QNO*38.8496*RAYICE
  IF(RC(I) .LE. 0.11) VP=385.267*RAYICE/QNO2*RC(I)**0.25
  IF(VP .LT. 0.2) VP=0.2 ! LIMIT MIN FALL SPEED TO 20 CM/S
C
  IF(Z .LT. HF) GO TO 251 ! USE RAIN EQTNS FOR Z < HF
  IF(RPBAR .GT. 0.11) VPBAR=RPBAR**0.125/QNO*38.8496*RAYICE
  IF(RPBAR .LE. 0.11) VPBAR=385.267*RAYICE/QNO2*RPBAR**0.25
  IF(RPBAR .GT. 0.) C=7.008E-4*BBP*EI*RAYICE/RPBAR**0.125
  IF(Z-DZ .LT. HF)
1 CALL SEARCH(I,C,RPBAR,QNO,BBP,Z,DZ,RAYICE,THRESH)
C      ! SEARCH FOR FREEZING LEVEL
C
C ... 'VDEP' ACCOUNTS FOR A PSEUDO-DIFFUSION ARISING FROM A VARIATION
C IN FALL SPEED WITH PARTICLE SIZE.
  VD=0.
  IF(RC(KK) .GT. 0.) CALL VDEP(3,DZ,QNO2,VP,KK,VD)
  IF(ABS(VD) .GT. VP) VD=0. ! NEGLECT IF UNREASONABLE
C
C ... INCLUDE CONTRIBUTION OF DEPOSITIONAL MASS TO TOTAL MASS
C
  CALL NUKE(I,VPBAR,Z,H,DZ,XNC,DMDT,RSV)
C
C ... LIMIT DEPOSITIONAL GROWTH RATE TO WATER CONDENSATION RATE
C IF CLOUD WATER EXISTS, OTHERWISE LIMIT TO ICE CONDENSATION
C RATE.
C
251 SRC=SORCE(KK)
  IF(RBAR .LE. D. .AND. Z .GT. HF) SRC=SORCEI(KK)
  DM=SRC-DMDT

```

```

      IF(DM .LT. 0.) DMDT=SRC
      IF(Z .GT. HF) GO TO 253

C
      IF(DMDT .GT. 0.) DMDT=0 ! ONLY ALLOW EVAPORATION OF RAIN
      IF(FLUXIN .GT. 0.) GO TO 225
      IF(AUTO(KK) .GT. 0.) GO TO 225 ! PRECIP SOURCE BELOW HF
      RC(I)=0. ! NO SOURCE OF PRECIP BELOW FREEZING LEVEL
      GO TO 270 ! OUT OF LOOP 260, RC(I)=0.

C
225  IF(RPBAR .GT. 0.) C=7.008E-4*QN0*EW/RPBAR**0.125
C
      VD=0.
      IF(RC(KK) .GT. 0.) CALL VDEP(2,DZ,QN02,VP,KK,VD)
      IF(ABS(VD) .GT. VP) VD=0. ! NEGLECT IF UNREASONABLE

C
253  S6=C*DZ
      C=S6*RPBAR

C
C ... DEPOSITIONAL GROWTH TERM
C ... HORIZONTAL PRECIPITATION FLUX TERM
      S5=RPBAR*GX(KK)*2.
C ... ACCRETION PLUS AUTOCONVERSION TERM
      SY=C*RPBAR+AUTO(KK)
      S6=S6*RPBAR
      ALPH(KK)=S6/DZ
      POUT1=RPBAR*FUP_ON(KK)/SCALE ! SCALE COMES FROM WINDUP
C ... FALLOUT TERM
      FLUX(I)=W(I)-VP-VD
      IF(I .EQ. L) GO TO 270 ! RC(L)=0 BECAUSE W(L)=0

C
C ... 'NEUTON' ADJUSTS RC(I) SUCH THAT DIFF1=0
      REMOV1=FLUXIN+SY+DMDT-S5
      REMOV2=FLUX(I)*RC(I)
      DIFF1=REMOV1-REMOV2

```

```

        IF(IVALUE .EQ. 1) GO TO 270  ! IN ZERO SEARCH MODE
C
        IF(RC(I) .EQ. RMAX) GO TO 258  ! NO STEADY STATE POSSIBLE
C
        CALL NEUTON(DIFF1,RC(I),XLIM,J,IFLAG)
        IF(IFLAG .EQ. 1) GO TO 270
C
260  CONTINUE  ! FIND RC AT CURRENT R
C
        WRITE(6,915)
915  FORMAT(1H0,' J=100,NO CONV AFTER ZERO START, WATER4 NEAR 260 ')
        STOP
C
C ... NO STEADY STATE POSSIBLE AT CURRENT LEVEL
258  IF(DIFF1 .LT. 0.) STOP ' DIFF1<0, NEAR 258-WATER4'
        FLUX(I)=0.
        POUT1=POUT1+DIFF1  !DIFF1 FALLS OUT TO BALANCE EQTN
C
270  POUT(I)=POUT1
        SUMC(KK)=S6
        SUMD(KK)=DMDT
C
C ... CHECK FOR CONVERGENCE.  IS THE CLOUD WATER (BARM2) THE
C     SAME THIS ITERATION AS IN THE PAST ITERATION?
C
C ... COMPUTE TOTAL PRECIP WATER
        BARM2=R(I)
        IF(BARM2 .EQ. 0.) BARM2=RC(I)
C
C ... GO AT LEAST 3 TIMES BEFORE A CONVERGENCE CHECK IS MADE
        IF(KL .LT. 3) GO TO 406
        IF(ABS(BARS) .LT. 1.E-9) GO TO 250
        BARS=(BARS-BARM2)/BARS
        IF(ABS(BARS) .LT. ZLIM) GO TO 250

```

```

C
C ... IF HERE, NO CONVERGENCE. TO REDUCE UNSTABLE WAVES IN VERTICAL
C PROFILE, USE, FOR THE NEW VALUE OF PRECIP CONCENTRATION, THE
C AVERAGE OF THE RECENTLY COMPUTED PRECIP WATER AND THE PREVIOUS
C AVERAGED VALUE OF PRECIP WATER.
C
      IF(RC(I) .EQ. RMAX) GO TO 406
      R1=RS(I)
      IF(IVALUE .EQ. 2) RC(I)=(R1+RC(I))/2.
406  RS(I)=RC(I)
C
2000  CONTINUE ! ADJUST R ND RC AT FIXED LEVEL,I
C
      WRITE(6,917) I,KL ! NO CONVERGENCE, KL > 60 AT LEVEL I
917  FORMAT(1H0,' KL TOO BIG, LOOP 2000,WATER4, I,KL=',2(1X,I3))
      STOP
C
250  IF(RC(I) .EQ. 0.) GO TO 2550 ! BELOW FREEZING LEVEL, OR AT TOP
C
      IF(I .EQ. IBASE) GO TO 2552 ! AT CLD BASE?
      IF(IVALUE .EQ. 2) GO TO 2552 ! IN ZERO SEARCH MODE?
C
      RSOL1=RC(I)
      CALL ZERO1(RMAXX(I),RC(I),DIFF1,VALUE,KUP,JFLG)
      GO TO (3001,3002,3003) JFLG
C
3002  RSOL=RSOL1 ! A POSSIBLE SOLUTION
      GO TO 3001 ! KEEP SEARCHING AT LARGER RC
C
3003  RC(I)=RMAXX(I) ! SOLTN IF NONE EXISTED FOR SMALLER RC
      IF(RSDL .EQ. 0.) GO TO 3004
      RC(I)=RSOL ! TAKE LARGEST RC < RMAX FOR SOLUTION
      GO TO 3000
C

```

```

3004 IF(VALUE .GT. 0.) GO TO 3000 ! CHECK SIGN OF FIRST GUESS
    IF(RSAV .GT. 0.) GO TO 3006
C
    WRITE(6,916)
916  FORMAT(1H0,' WARNING,FLUXIN=0,DIFF1<0, 916-WATER4')
    RC(I)=(AUTO(KK)+DMDT)/(W(I)-0.2) ! NEW STARTING GUESS
    WRITE(6,900) I,AUTO(KK),DMDT,RC(I)
    GO TO 3000
C
3006 RSAV=0. ! FLOAT IS JUST ABOVE CURRENT BIN BASE
    FLUX(KK)=0.
    POUT(KK)=POUT(KK)+FLUXIN ! DROP OUT FLUXIN FROM LOWER BIN
    GO TO 1999 ! START OVER WITH FLOAT AT LOWER BIN TOP
C
3000 CONTINUE ! RERUN WITH NEW GUESS OF RC NEAR ZEROS OF EQTN
C
2552 IR=1 ! DROP PRECIP OUT NEAR FLOAT SIZES
    IF(Z .GT. HF) IR=2
    CALL FALOUT(IR,W(I),RC(I),QNO,QNO2,RAYICE,RCUT,VP,FALLOUT)
    IF(FALLOUT .LE. 0.) GO TO 2551
    POUT(I)=POUT(I)+FALLOUT ! ADD FALLOUT FROM SPLITTING MASS
    FLUX(I)=RCUT*(W(I)-VP) ! RCUT=MASS OUT TOP OF BIN
C
                                NO DEPOSITION VEL IN SPLITTING
C
2551 FALL(I)=VP ! FALL SPEED (M/S)
    PRATE(I)=-RC(I)*(W(I)-FALL(I)) ! PRECIP RATE (G/M2/S)
C
2550 CONTINUE ! GO TO NEXT HIGHER LEVEL
C
900  FORMAT(1H ,I3,6(1PE10.3))
C
    DATA(3)=P/1000. ! OUTPUT FOR 'MAINUP'
C
    DO 475 I=1,L ! UNITS CONVERSION

```

```

      PRATE(I)=PRATE(I)*3.6  ! (MM/HR)
475  AUTO(I)=AUTO(I)/DZ  ! (G/M3/S)
      RETURN
      END

```

```

C .....

```

```

      SUBROUTINE WINDUP(E,RHG,XX,IFLG,T,DT,WMAX)
      INCLUDE 'SCAV.FOR'
      DIMENSION XLIM(7),USAVE(L)
      DATA XLIM/D.,0.,1.E-8,1.E-4,0.,1.E+5,-1./

```

```

C
C.....
C
C ... COMPUTES VERTICAL PROFILES OF U,V,W GIVEN A STORM EFFIC. AND/OR
C    AN AVERAGE STORM UPDRAFT VELOCITY.
C

```

```

C.....

```

```

C ... CONVECTIVE CLOUD MODULE

```

```

C
C ... RELEASED NOV, 1986
C
C ... FOR INFORMATION, CONTACT:
C    B.C. SCOTT OR C.M. BERKOWITZ
C    PACIFIC NORTHWEST LABORATORY
C    P.O. BOX 999, RICHLAND, WA. 99352
C

```

```

C.....

```

```

C
C ... VARIABLES IN CALL LIST:
C    E=STORM EFFICIENCY (PRECIP RATE/VAPOR INFLOW)--INPUT
C    RHG=SFC RELATIVE HUMIDITY (DECIMAL)

```

```

        XX=STORM WIDTH (M)--INPUT
C      IFLG=COUNTER--INPUT
C      T=STORM DURATION
C      DT=DECREMENT TO STORM DURATION WHEN TRYING TO MATCH EFFIC.
C      WMAX=MAX UPDRAFT-CELL VERTICAL VELOCITY--COMPUTED
C
C.....
C
C ... FLAGS:
C      IFLG=COUNTER GIVING NUMBER OF PREVIOUS CALLS TO 'WINDS'
C      JFLG=0 WHEN CONVERGENCE HAS NOT OCCURRED FOR PROPER EFFICIENCY
C          =1 WHEN CONVERGENCE HAS OCCURRED FOR EFFICIENCY
C      NOPRE=0 FOR RAINSHAFT-CELL
C          =1 FOR UPDRAFT-CELL
C
C.....
C
C ... UNITS ARE:
C      LENGTHS--METERS
C      VELOCITY--METERS/SECOND
C      HUMIDITY--DECIMAL RATHER THAN %
C      DENSITY--GRAM/M3
C
C.....
C
C ... OTHER DEFINITIONS:
C      ZBASE=CLOUD BASE (M) (IN COMMON)
C      L=NUMBER OF VERTICAL GRIDS (DEFINED IN SCAV.FOR)
C
C.....
C
        H=DATA(1)  ! STORM HEIGHT (METERS)
C
        LL=L-1

```



```

DZ=H/FLOAT(LL) ! VERTICAL GRID THICKNESS (M)
SCALE=XX*XX/DZ*1.E+4 ! GIVES UNITS CM2/S IN FLUXES
DATA(12)=SCALE ! SAVE FOR SUBROUTINE 'FLUX'

C
COE=RHG*SAT(TEMP(1))*18.E+6/RHO(1) ! SFC MIXING RATIO
WRATIO=.875 ! (HT OF WMAX)/(STORM HEIGHT)

C
DO 300 K=1,300 ! TIME (T) BEING CHANGED
IF(T .GT. 20.) GO TO 305 ! KEEP TIME GREATER THAN 20 MIN
IFLG=400
RETURN ! TIME TOO SMALL, RETURN TO MAINUP

C
305 XJ=DATA(7)*60./T
XJ=XJ/3.6 ! PRECIPITATION RATE (G/M2/S)

C
C ... LOOP 350 ADJUSTS WBAR TO GET PROPER EFFICIENCY
C
DO 350 J=1,50 ! DATA(11) [UPDRAFT SPEED] BEING CHANGED
Z=-DZ
A1=2.*DATA(11) ! CONSTANT FOR UPDRAFT-CELL
IF(NOPRE .EQ. 0) A1=6.*DATA(11)/H ! CONSTANT FOR RAINSHAFT-CELL

C
C ... LOOP 100 COMPUTES PROFILES OF VERTICAL VELOCITY (W), VAPOR DENSITY
C (VAP) AND AIR DENSITY
DO 100 I=1,L
Z=Z+DZ
IF(IFLG .LT. 100) GO TO 850 ! NORMAL SOLUTION
W(I)=-A1*(Z-Z*Z/H) ! SUBSIDENCE IN ENTIRE RAINSHAFT
GO TO 998

C
850 IF(NOPRE .EQ. 1) GO TO 999 ! IF TRUE, COMPUTE FOR UPDRAFT-CELL
C
IF(Z .LT. H/2.) W(I)=-2.*A1*(Z-2.*Z*Z/H) ! PARABOLIC, RAINSHAFT
IF(Z .GE. H/2.) W(I)=2.*A1*(3.*Z-H-2.*Z*Z/H)

```

```

      IF(I .EQ. L) W(I)=0.
      GO TO 998
C
999  IF(Z .LE. WRATIO*H) W(I)=A1*Z/WRATIO/H  ! UPDRAFT-CELL
      IF(Z .GT. WRATIO*H) W(I)=A1*(1.-(Z-WRATIO*H)/(H*(1.-WRATIO)))
C
998  IF(IFLG .GT. 0) GO TO 100  ! IF TRUE, SKIP MOISTURE CALCS.
      IF(K .GT. 1 .OR. J .GT. 1) GO TO 100
C
C ... OPERATIONS BELOW ONLY NEED TO BE DONE 1ST TIME THROUGH.
      CO12(I)=RHO(I-1)/RHO(I)  ! DENSITY RATIO
      IF(I .EQ. 1) CO12(I)=1.0
C
      V=-1.  ! VAPOR DENS. IN UPDRAFT
      IF(NOPRE .EQ. 0) V=0.  ! VAPOR DENS. IN RAINSHAFT IF TRUE
C ... 'COEF' RETURNS AVERAGE SATURATION VAPOR DENSITY IN THE INTERVAL
C    Z TO Z+DZ. VAP IS THE ACTUAL VAPOR DENSITY IN THE INTERVAL DZ.
      CALL COEF(I,DZ,Z,C,V,RWET,B)
C
      VAP(I)=RWET  ! IN RAINSHFT OR ABOVE CLOUD BASE
      VAPI(I)=B
C
100  CONTINUE
C
C
C ... COMPUTE U,V COMPONENTS BASED ON CONTINUITY EQUATION.
C
      SUM5=0.  ! FOR INFLOW VAPOR FLUX
      Z=-DZ
C
C ... LOOP 200 COMPUTES HORIZONTAL WIND SPEEDS AND INFLOW VAPOR FLUX
C
C ... DEFINITIONS--HORIZONTAL WINDS (M/S)
C    G=WIND IN X-DIRECTION AT X ( POSITIVE INTO STORM)

```

```

C      UOUT= WIND IN X-DIRECTION AT X+DX ( NEGATIVE INTO STORM)
C      VIN=WIND IN Y-DIRECTION AT Y (POSITIVE INTO STORM)
C      VOUT=WIND IN Y-DIRECTION AT Y+DY (NEGATIVE INTO STORM)
C ... ALL HORIZONTAL WIND SPEEDS HAVE BEEN MULTIPLIED BY DZ/XX
C
      WMAX=0. ! FOR MAX UPDRAFT WINDS
      DO 200 I=1,LL
      Z=Z+DZ
      WIND=(W(I+1)-CO12(I+1)*W(I))/(1.+CO12(I+1))
      IF(NOPRE .EQ. 0) GO TO 400 ! FOR RAINSHAFT
      IF(Z+DZ .GT. WRATIO*H) GO TO 401 ! FOR OUTFLOW PORTION
      IF(WIND .GT. 0.) GO TO 403
      W(I+1)=CO12(I+1)*W(I) ! NO OUTFLOW BELOW WIND MAX
      WIND=0.
403   G(I)=WIND
      UOUT(I)=0. ! NO INFLOW FROM RAINSHAFT TO UPDRAFT
      USAVE(I)=UOUT(I) ! SAVE FOR RAINSHAFT INPUT
      VIN(I)=G(I)/2. ! ARBITRARY SPECIFICATION
      VO(I)=-VIN(I) ! DU/DX = DV/DY
      GO TO 402
C
401   UOUT(I)=-WIND ! OUTFLOW INTO RAINSHAFT
      USAVE(I)=UOUT(I) ! SAVE FOR RAINSHAFT
      G(I)=0.
      VIN(I)=-UOUT(I)/2. ! ARBITRARY SPECIFICATION
      VO(I)=-VIN(I) ! ARBITRARY SPECIFICATION
      GO TO 402
C
400   UOUT(I)=-USAVE(I) ! INFLOW FROM RAINSHAFT
      G(I)=UOUT(I)+WIND
      VIN(I)=G(I)/2. ! ARBITRARY SPECIFICATION
      VO(I)=VIN(I)-WIND ! DU/DX = DV/DY
C
402   F=0.

```

```

      IF(NOPRE .EQ. 0) GO TO 200  ! NEGLECT FLUXES IN RAINSHAFT
      IF(WMAX .LT. W(I)) WMAX=W(I)
      IF(G(I) .GT. 0.) F=F+G(I)
      IF(UOUT(I) .LT. 0.) F=F-UOUT(I)
      IF(VIN(I) .GT. 0.) F=F+VIN(I)
      IF(VO(I) .LT. 0.) F=F-VO(I)
C
      RWET=VAP(I)
      RDRY=COE*(RHO(I)+RHO(I+1))/2.  ! CONSTANT MIXING RATIO
      IF(Z .LT. ZBASE) RWET=RDRY
C
C ... SUM5=INFLOW VAPOR FLUX, F=INFLOW WINDS
      SUM5=SUM5+F*RWET
C
200  CONTINUE
C
      IF(NOPRE .EQ. 0) GO TO 301  ! RAINSHAFT, NEGLECT EFFICIENCY
C
C ... EFF1=COMPUTED EFFICIENCY, E=DESIRED EFFICIENCY
      EFF1=XJ/SUM5
C
C ... 'NEUTON' TRIES TO ADJUST EFFICIENCY SO THAT DIFF1=0
      DIFF1=E-EFF1
C
C ... ITERATE ON WBAR TO GET PROPER EFFICIENCY
C
802  FORMAT(1H ,I3,5(1X,E13.6))
      CALL NEUTON(DIFF1,DATA(11),XLIM,J,JFLG)
      IF(JFLG .EQ. 1) GO TO 302  ! SOLUTION, IF TRUE
C
350  CONTINUE  ! RECOMPUTE WITH NEW WBAR
      GO TO 300  ! NO SOLUTION, TRY REDUCING STORM DURATION
C
302  IF(WMAX .GT. 2.0) GO TO 301

```

```

C
300  T=T-DT
C
301  IF(NOPRE .EQ. 1) CALL FLUX(FEN_UP,FUP_EN,FUP_DN)
      IF(NOPRE .EQ. 0) CALL FLUX(FEN_DN,FDN_EN,FDN_UP)
C
900  FORMAT(1H ,I3,6(1X,1PE10.3))
      DATA(5)=E
      DATA(7)=XJ*3.6
      RETURN
      END

C .....

      SUBROUTINE COEF(I,DZ,Z,C,V,A,B)
      INCLUDE 'SCAV.FOR'
C
C ... IF V=-1, 'COEF' RETURNS:
C      AVERAGE SATURATION VAPOR DENSITY IN THE INTERVAL Z TO Z+DZ
C      IN UPDRAFT REGION
C ... IF V=0, 'COEF' RETURNS:
C      AVERAGE SATURATION VAPOR DENSITY IN THE INTERVAL Z TO Z+DZ
C      IN RAINSHAFT REGION
C ... IF V=-2, 'COEF' RETURNS THE TERMS IN THE DENOMINATOR OF THE
C      MASS FLUX EXPRESSIONS, I.E.: M(J)=SOURCE/DENONINATOR
C
C ... TERMS IN THE CALL LIST:
C      I=HEIGHT INDEX
C      DZ=GRID INCREMENT WHEN V=-1
C      Z=HEIGHT WHEN V=-1
C      C=SCAVENGING COEF WHEN V=-2
C      V=A FLAG (SEE ABOVE)
C      A=COMPUTED SATURATION VAPOR DENSITY OVER WATER IF V=0,-1 (G/M3)

```

```

C      B=COMPUTED DENOMINATOR OF FLUX EXPRESSION IF V=-2 (M/S)
C      B=COMPUTED SATURATION VAPOR DENSITY OVER ICE IF V=0,-1 (G/M3)
C
      IF(V .EQ. -2) GO TO 102
C ... COMPUTE SATURATION VAPOR DENSITY
C
      IF(V .EQ. -1.) CALL VAPOR(TEMP,I,A,B) ! UPDRAFT WATERVAPOR CONCEN.
      IF(V .EQ. 0.) CALL VAPOR(TEMP,I,A,B) ! RAINSHAFT WATERVAPOR
      A=A*18.E+6      !AVERAGE BETWEEN I & I+1 IN G/M3
C
      B=B*18.E+6      !AVERAGE BETWEEN I & I+1 IN G/M3
      RETURN
C
C ... COMPUTE DENOMINATOR IN FLUX EQUATION
102      V=0.
      GG=C
      IF(V0(I) .GT. 0.) GG=V0(I)+GG
      IF(UOUT(I) .GT. 0.) GG=GG+UOUT(I)
      IF(G(I) .LT. 0.) GG=GG-G(I)
      IF(VIN(I) .LT. 0.) GG=GG-VIN(I)
      B=GG/2.
      RETURN
      END

C .....

      SUBROUTINE VAPOR(TEMP,I,A,B)
      DIMENSION TEMP(1)
C ... FUNCTION SAT,SSAT RETURNS SATURATION CONCENTRATION OF WATER
C ... VAPOR (G-MOLES/CC) AT TEMPK.
C
      A=(SAT(TEMP(I))+SAT(TEMP(I+1)))/2. ! SATURATION OVER WATER
      B=(SSAT(TEMP(I))+SSAT(TEMP(I+1)))/2. ! SATURATION OVER ICE

```

RETURN
END

C

SUBROUTINE DEFALT(I)
INCLUDE 'SCAV.FOR'

C

C ... THIS ROUTINE RESTORES PREVIOUS (OR DEFAULT) VALUES TO VARIABLES
C R, RC, W,.... IF CURRENT INPUT CONDITIONS RESULT IN UN-
C REALISTIC PROFILES.

C

C ... WHEN I=0, UPDRAFT VALUES PLACED IN ARRAY 'SAVE'
C ... WHEN I=1, UPDRAFT VALUES RESTORED TO VARIABLES R,RC,...
C ... WHEN I=2, RAINSHAFT VALUES PLACED IN ARRAY 'SAVER'
C ... WHEN I=3, RAINSHAFT VALUES RESTORED TO VARIABLES R,
C RC, W,... I.E., PLACED IN COMMON BLOCK /SET/.

C

C

IF(I .EQ. 0) GO TO 50
GO TO (200, 400, 500) I

C

C ... SAVE CURRENT VALUES

50 DO 100 J=1,LSAVE

100 SAVE(J)=R(J)

RETURN

C

C ... RESTORE PREVIOUS VALUES

200 DO 300 J=1,LSAVE

300 R(J)=SAVE(J)

RETURN

C

C ... SAVE CURRENT VALUES, RAINSHAFT PORTION, I=2

```

400  DO 450 J=1,LSAVE
450  SAVER(J)=R(J)
      RETURN
C
C ... RESTORE PREVIOUS VALUES, RAINSHAFT PORTION, I=3
500  DO 550 J=1,LSAVE
550  R(J)=SAVER(J)
      RETURN
      END

C .....

      SUBROUTINE NEUTON(DIFF1,A,XLIM,KK,IFLAG)
C
      DIMENSION XLIM(1)
C.....
C
C ... CONVECTIVE/STRATIFORM CLOUD MODULE
C
C ... RELEASED NOV, 1986
C
C ... FOR INFORMATION, CONTACT:
C   B.C. SCOTT OR C.M. BERKOWITZ
C   PACIFIC NORTHWEST LABORATORY
C   P.O. BOX 999, RICHLAND, WA. 99352
C
C.....
C
C
C
C ... NEUTON'S METHOD ITERATES ON 'A' SUCH THAT THE FUNCTION F(A)=0;
C   I.E.,  $A=A_0 - F(A_0)/(DF/DA)$ . HERE THE FUNCTION F DESCRIBES:
C   1. THE SURFACE PRECIP RATE WHEN A= VERTICAL VELOCITY

```



```

C      2. MASS FLUX BALANCE WHEN A=PRECIP PARTICLE CONCENTRATION
C      3. STORM EFFICIENCY WHEN A= THE VARIABLE SHIR
C
C
C ... VARIABLES IN CALL LIST:
C      DIFF1=THE VALUE OF THE FUNCTION F AT A0
C      A=VALUE OF A IN ABOVE FORMULA
C      XLIM=ARRAY OF NUMBERS SUCH THAT
C          XLIM(1)=VALUE OF A0 IN FORMULA
C          XLIM(2)=PREVIOUS VALUE OF DIFF1 (=F(A0))
C          XLIM(3)=LARGEST ACCEPTABLE VALUE OF A-A0 FOR CONVERGENCE
C          XLIM(4)=CONVERGENCE CRITERIA FOR F=0
C          XLIM(5)=LOWER LIMIT FOR A
C          XLIM(6)=UPPER LIMIT FOR A
C          XLIM(7)= PREVIOUS DIRECTION OF ZERO SEARCH
C      KK=COUNTER FOR NUMBER OF PASSES THROUGH 'NEUTON'
C      IFLAG=CONVERGENCE FLAG
C
C ... FLAGS:
C      IFLAG=0 FOR NO CONVERGENCE
C          =1 FOR CONVERGENCE
C      IIN= -1 GUESS DECREASES SIZE OF A
C          = +1 GUESS INCREASES SIZE OF A
C
C .....
C ... BEGIN NEUTON'S METHOD
C
C      AOLD=A ! SAVE CURRENT A
C      A2=XLIM(1)
C      DIFF2=XLIM(2) ! OLD VALUE OF DIFFERENCE
C      ALOW=XLIM(5)
C      AHIGH=XLIM(6)
C      IIN=XLIM(7)
C      YLIM=XLIM(3)

```

```

        IF(KK .GT. 30) YLIM=1.E-8
        IF(KK .GT. 1) GO TO 204
C
C ... MUST DEFINE FIRST TWO GUESSES A2, AND 'A' TO START ALGORITHM
        ALOW=0 ! LOWER LIMIT OF A
        AHIGH=1.E+5 ! UPPER LIMIT OF A
        IIN=-1 ! FIRST GUESS DECREASES SIZE OF A
        A=.95*A
        GO TO 203
C
C ... DEFINE DELTA A (D) AND DELTA F FOR DERIVATIVE DF/DA (XM)
204  SIGN=DIFF1*DIFF2 ! SIGN NEGATIVE IF ZERO IS CROSSED
        IF(ABS(DIFF2) .LT. 1.E-15) GO TO 202
        IF(KK.EQ.2.AND.DIFF1/DIFF2.GT.1) IIN=-IIN
202  IF(SIGN .LE. 0.) IIN=-IIN ! CHANGE DIRECTION OF ZERO SEARCH
C
C ... COMPUTE NEW GUESS OF A
        D=AOLD-A2 ! CURRENT VALUE MINUS OLD VALUE
        IF(ABS(D) .LT. YLIM) GO TO 205
        XM=(DIFF1-DIFF2)/D
C
C ... SET LIMITS IF DERIVATIVE GETS TOO SMALL
        IF(ABS(XM) .LT. 1.E-9) XM=-100.
C
C ... COMPUTE NEW VALUE FOR 'A' BASED ON ABOVE FORMULA
        A=AOLD-DIFF1/XM
C
C ... KEEP 'A' FROM CHANGING TOO MUCH AND KEEP 'A' > 0
        A3=AOLD
        IF(AOLD .LT. XLIM(3)) A3=1.E-12
        IF(A/A3 .GT. 5) A=5.*AOLD
        IF(A .LE. 0.) A=(A+DIFF1/XM)*0.5
        IF(A .LE. 0.) A=-A
C

```

```

C ... CHECK CONVERGENCE CRITERIA
    IF(ABS(DIFF1) .GT. XLIM(4)) GO TO 203
    IF(ABS((A-AOLD)/A) .LT. 1.E-4) GO TO 205
203  DIFF2=DIFF1
    IFLAG=0
    IF(KK .GT. 1) GO TO 206
    GO TO 208
205  IFLAG=1
    RETURN

C
C ... CHECK REASONABLENESS OF NEW GUESS
206  IF(IIN .LT. 0) GO TO 207  ! GUESS DECREASING IF TRUE
    ALOW=AOLD  ! GUESS OF A MUST BE > THIS LOWER LIMIT
    IF(A .LT. ALOW) A=(ALOW+AHIGH)/2.
    IF(A .LT. AHIGH) GO TO 208
    A=(AOLD+AHIGH)/2.  ! KEEP A BELOW UPPER LIMIT
    GO TO 208

C
207  AHIGH=AOLD  ! GUESS OF A MUST BE < THE UPPER LIMIT
    IF(A .GT. AHIGH) A=(ALOW+AHIGH)/2.
    IF(A .GT. ALOW) GO TO 208
    A=(AOLD+ALOW)/2.  ! KEEP A ABOVE LOWER LIMIT

C
C ... SAVE OLD VALUES OF A AND F
208  XLIM(1)=AOLD
    XLIM(2)=DIFF2
    XLIM(5)=ALOW
    XLIM(6)=AHIGH
    XLIM(7)=IIN
    RETURN
    END

```

```

C .....

      SUBROUTINE FLUX(FEN_CD,FCD_EN,FTOCELL)
C
C ... COMPUTES WINDS INTO AND OUT OF CONVECTIVE UPDRAFT AND
C   DOWNDRAFT CELLS.
C
      INCLUDE 'SCAV.FOR'
      DIMENSION FEN_CD(L),FCD_EN(L),FTOCELL(L)
      SCALE=DATA(12) ! FROM WINDUP
C
      DO 100 I=1,L
      GG=0 ! OUTFLOW (POSITIVE)
      F=0 ! INFLOW (POSITIVE)
C
      IF(G(I) .GT. 0.) F=F+G(I)
      IF(VIN(I) .GT. 0.) F=F+VIN(I)
      IF(VO(I) .LT. 0.) F=F-VO(I)
C
C ... FLUX FROM ENVIRONMENT TO CLOUD (CM*CM/S)
      FEN_CD(I)=F*SCALE ! POSITIVE
C
      IF(G(I) .LT. 0.) GG=GG-G(I)
      IF(VIN(I) .LT. 0.) GG=GG-VIN(I)
      IF(VO(I) .GT. 0.) GG=GG+VO(I)
C
C ... FLUX FROM CLOUD TO ENVIRONMENT (CM*CM/S)
      FCD_EN(I)=GG*SCALE ! POSITIVE
C
C ... FLUX FROM CURRENT CELL TO ADJACENT CELL (CM*CM/S)
      FTOCELL(I)=0.
      IF(UOUT(I) .GT. 0.) FTOCELL(I)=UOUT(I)*SCALE
100  CONTINUE

```

RETURN
END

C.....

SUBROUTINE UNITS

C

C ... CONVERT TO RADN UNITS

C

INCLUDE 'SCAV.FOR'

C

DO 100 I=1,L

K=I+1

IF(K .GT. L) K=I

RBAR=(R(I)+R(K))/2.

RPBAR=(RC(I)+RC(K))/2.

IF(RBAR .GT. 0.) AUTO(I)=AUTO(I)/RBAR ! AUTOCONVERSION (1/S)

IF(RPBAR .GT. 0.) DEPOS(I)=DEPOS(I)/RPBAR ! DEPOS-EVAP (1/S)

C ... DENSITY IN UNITS OF MOLE/CM3

R(I)=3.44828E-8*R(I)/RHO(I) ! CLOUD WATER (G-WATER/G-AIR)

RC(I)=3.44828E-8*RC(I)/RHO(I) ! PRECIP WATER (G-WATER/G-AIR)

W(I)=W(I)*100. ! UPDRAFT (CM/S)

100 FALL(I)=FALL(I)*100. ! FALL SPEED (CM/S)

C

C ... P-DUMP (POUT) CHANGED IN INTERPOLATION ROUTINE

RETURN

END

C.....

SUBROUTINE ZERO1(RMAX,RC,DIFF1,VAL,K,IFLG)

DIMENSION VALUE(85),AXIS(85)

```

C
C ... IFLG=1 KEEP TRYING FOR SOLUTION
C     IFLG=2 SOLUTION FOUND
C     IFLG=3 NO SOLUTION EXISTS
C
      VALUE(K)=DIFF1
      VAL=VALUE(1)
      AXIS(K)=RC
      IF(K .GT. 1) GO TO 100
      X=1.
      Y=10.
      R0=RC
100    X=X+0.1*Y
      IF(X .LE. Y) GO TO 200
      X=X-0.1*Y
      Y=Y*10.
      X=X+0.1*Y
200    RC=X*R0
      VALUE(K)=DIFF1
      AXIS(K)=RC
C
      IFLG=1
      K=K+1
      IF(K .EQ. 2) RETURN
      SIGNX=VALUE(K-2)*VALUE(K-1)
      IF(SIGNX .GE. 0.) GO TO 300
      RC=AXIS(K-1)
      IF(K .GE. 3) RC=2.*AXIS(K-1)-AXIS(K-2)
      IFLG=2  ! SOLUTION
      RETURN
C
300    IF(RC .LT. RMAX) RETURN  ! KEEP TRYING
      IFLG=3  ! NO SOLUTION

```

```

      RETURN
      END

```

```

C.....

```

```

      SUBROUTINE SIGMA(HTOP,T,SFCPRES)

```

```

C

```

```

      INCLUDE 'SCAV.FOR'

```

```

      COMMON /BK3/ Z(L),PRESS(L)

```

```

C

```

```

C ... CONVECTIVE CLOUD, UPDRAFT-CELL PORTION....

```

```

C

```

```

c establish vertical grid: defines z(i) in centimeters.

```

```

      N=L

```

```

      nml=n-1          !L IS # OF LAYERS IN CLOUD MODEL

```

```

      dz = hTOP*100./float(nml) !centimeters

```

```

      call grdset(N,DZ,Z)

```

```

      ZBASE1=ZBASE*100.

```

```

      temp0 = T+273.15 !kelvin, sfc. temp.

```

```

      DO I = 1, N

```

```

        RATE=7.E-5

```

```

        IF(Z(I) .LT. ZBASE1) RATE=9.8E-5

```

```

        TEMP(I)=TEMPO-RATE*(Z(I)-Z(1))

```

```

      END DO

```

```

c initial guess at pressure profile:

```

```

      press0 = sfcpres*1.e3 !dynes/cm2, sfc. pressure

```

```

      DO 100 JJ=1,3

```

```

        CALL PRESSF( 1,PRESS0,PRESS,TEMP,RHO,Z,N )

```

```

C modify temp. to be along moist adiabat.

```

```

      TEMPX = TEMPO          !kelvin, sfc. temp.

```

```

        JFOG=0
        ZBOT=Z(1)
C
C ... DRY LAPSE BELOW LCL, WET LAPSE ABOVE
C
        DO 200 I=2,N
212    ZTOP=Z(I)
        IF(ZTOP .LT. ZBASE1) GO TO 210  ! DRY LAPSE RATE IF TRUE
        IF(JFOG .EQ. 1) GO TO 210  ! ABOVE CLD BASE IF TRUE
        ZTOP=ZBASE1  ! AT CLD BASE
C
210    TEMPX=TEMPX+(ZTOP-ZBOT)*RLAPSE(TEMPX,RHO(I-1),JFOG)
        TEMP(I)=TEMPX
        ZBOT=ZTOP
        IF(ZTOP .NE. ZBASE1) GO TO 200  ! ABOVE OR BELOW CLD BASE IF TRUE
        JFOG=1  ! AT CLD BASE, WET LAPSE ABOVE
        GO TO 212
C
200    CONTINUE
100    CONTINUE
        RETURN
        ENO

```

C.....

SUBROUTINE SIGMAD

```

C
C .... COMPUTES TEMPERATURE, PRESSURE IN RAINSHAFT PORTION
C
        INCLUDE 'SCAV.FOR'
        COMMON /BK3/ Z(L),PRESS(L)
C
        temp0 = TEMP(L) ! kelvin, top temp.

```



```

C
C *****
C ... FOR CONVECTIVE CLOUDS, INTEGRATING DOWN WET ADIABAT
C *****
C
      N=L
      RATE=7.E-5
      DO J = 1, N
        I=N-J+1
        TEMPD(I)=TEMPO+RATE*(Z(L)-Z(I))
      END DO

c  initial guess at pressure profile:
      press0 = PRESS(L) !dynes/cm2, TOP pressure
      DO 100 JJ=1,9
        CALL PRESSFD( 1,PRESS0,PRESSD,TEMPD,RHOD,Z,N )
C  modify temp. to be along moist adiabat.
      TEMPX = TEMPO          !kelvin, sfc. temp.
      JFOG=1  ! WET ADIABAT
      DO 200 J=2,N  ! INTEGRATE DOWN
        I=N-J+1
        K=I
        IF(K .LT. 1) K=1
        TEMPX = TEMPX - (Z(I+1)-Z(I))*RLAPSE( TEMPO(K),RHOD(K),JFOG )
        TEMPD(I) = TEMPX
200    CONTINUE
100    CONTINUE
      RETURN
      END

C .....

      SUBROUTINE PRESSFD( INIT,PRESS0,PRESS,C,CS,Z,N )

```

```

C
      DIMENSION PRESS(1),C(1),CS(1),Z(1)
C
C   CALCULATES PRESSURE BY INTEGRATING THE HYDROSTATIC EQUATION
C
C
C   NOTE:  PRESSURE UNITS = DYNES/CM**2
C          AIR DENSITY UNITS = MOLES/CM**3
C
C   SET PRESSURE AT TOP GRID
C
      PRESS(N) = PRESS0
      CS(N) = RHOAIR(PRESS(N), C(N))
C
C   INTEGRATE HYDROSTATIC EQUATION, INTEGRATING DOWN
C       $D(\ln(P))/DZ = -G/(R*T)$ 
C
      DO 1000 J = 2,N
      I=N-J+2
      DZDUM = Z(I)-Z(I-1)
      TEMPBB = .5*(C(I-1) + C(I))
      PRESS(I-1) = PRESS(I)*EXP(3.416E-4*DZDUM/TEMPBB)
      CS(I-1) = RHOAIR( PRESS(I-1), C(I-1))
1000  CONTINUE
      RETURN
      END
C-----
C

```

```

      SUBROUTINE GRDSET(N,DZ,Z)
C
      DIMENSION Z(1)
C  ESTABLISH UNIFORMLY SPACED GRID
C
      Z(1) = 0.      !HEIGHT ABOVE GROUND (CM)
      DO I = 2, N
        Z(I) = Z(I-1) + DZ
      END DO
      RETURN
      END

C.....

      SUBROUTINE PRESSF( INIT,PRESSO,PRESS,C,CS,Z,N )
C
      DIMENSION PRESS(1),C(1),CS(1),Z(1)
C
C  CALCULATES PRESSURE BY INTEGRATING THE HYDROSTATIC EQUATION
C  *** PRESSF SHOULD ALWAYS BE ENABLED (IPRESS=1) IF WAGRID IS NON-ZERO
C
C  WHEN WAGRID(1) IS NOT ZERO, THEN PRESS(1) - THE PRESSURE AT THE
C  BOTTOM OF THE GRID, MUST ALSO BE CALCULATED.  IN THIS VERSION,
C  THIS IS DONE BY INTEGRATING HYDROSTATIC EQN. OVER GRID HEIGHT
C  CHANGE (GLEVL - GLEVLOLD)
C
C  NOTE:  PRESSURE UNITS = DYNES/CM**2
C         AIR DENSITY UNITS = MOLES/CM**3
C
C  SET PRESSURE AT BOTTOM GRID
C
      IF (INIT.EQ.1) THEN

```

```

        PRESS(1) = PRESS0
ELSE
        DZDUM = GLEVL - GLEVLOLD
        PRESS(1) = PRESS(1)*EXP(-3.416E-4*DZDUM/C(1))
END IF
GLEVLOLD = GLEVL
CS(1) = RHOAIR(PRESS(1), C(1))
C
C   INTEGRATE HYDROSTATIC EQUATION
C    $D(\ln(P))/DZ = -G/(R*T)$ 
C
DO 1000 I = 2,N
        DZDUM = Z(I)-Z(I-1)
        TEMPBB = .5*(C(I-1) + C(I))
        PRESS(I) = PRESS(I-1)*EXP(-3.416E-4*DZDUM/TEMPBB)
        CS(I) = RHOAIR( PRESS(I), C(I))
1000 CONTINUE
        RETURN
END

C-----

        FUNCTION SAT (TEMPK)
C
C   ROUTINE FOR CALCULATING THE SATURATION CONCENTRATION OF
C   WATER VAPOR (GRAM-MOLES/CC) AT TEMPK DEGREES KELVIN
C   (ESAT IS SATURATION VAPOR PRESSURE IN MB)
C
        DATA A1,A2,A3,A4,A5,A6,A7/-21.2929759,-7629.01437,-450.173938,
110702.3730,-271.34792805,2479.85588,-296.2647743/
C
        ESAT=A1+A2/(A3+TEMPK+A4/(A5+TEMPK+A6/(A7+TEMPK)))

```

```

IF(TEMPK .LT. 220.) ESAT=10.** (9.2-2353./TEMPK)
SAT = (ESAT*1.E3)/(8.3144E7*TEMPK)
RETURN
END

```

C.....

```

FUNCTION SSAT (TK)

```

```

C
C  SSAT= THE SATURATION CONCENTRATION OF WATER VAPOR (GRAM MOLE/CC)
C      OVER ICE AT TK DEGREES KELVIN
C      (ESAT IS THE SATURATION VAPOR PRESSURE IN MB)
C
TC=TK-273.16
ESAT=6.109177956 +TC*(5.034698970E-1 + TC*(1.886013408E-2
+   +TC*(4.176223716E-4+TC*(5.824720280E-6
+   +TC*(4.838803174E-8 + TC*(1.838826904E-10))))))
SSAT=(ESAT*1.E3)/(8.3144E7*TK)
RETURN
END

```

C.....

```

FUNCTION RHOAIR( P, TEMPK )

```

```

C
C  RHOAIR = AIR DENSITY (MOLES/CC)
C  P      = PRESSURE (DYNES/CM**2)
C  TEMPK  = TEMPERATURE (K)
C
RHOAIR = P/(8.3144E7*TEMPK)
RETURN
END

```

```

C-----
      FUNCTION RLAPSE ( TEMPK, RHOA, JFOG )
C
C   RLAPSE = LAPSE RATE (1/CM)
C       ADIABATIC IF JFOG=0
C       PSEUDOADIABATIC OTHERWISE
C   TEMPK = TEMP (K)
C   RHOA = AIR DENSITY (MOLES/CC)
C
C   THERMODYNAMIC UNITS ARE CGS (CONSTANTS USE MOLES)
C   THE SAT FUNCTION MUST RETURN THE SATURATION VAPOR
C       DENSITY (MOLES/CC) AT TEMP TEMPK
      DATA GAMMAD/-9.76E-5/! DRY ADIABATIC LAPSE RATE (DEG K/CM)
      DATA CP/2.91E8/! AIR HEAT CAPACITY, CONSTANT PRESSURE (ERG/MOLE
      DATA R/8.314E7/! GAS CONSTATN (ERG/MOLE/DEG K)
      DATA HLC/4.5E11!LATENT HEAT OF CONDENSATION (ERG/MOLE WATER)
      RLAPSE = GAMMAD
      IF (JFOG.NE.0) THEN
        DUMMY1 = HLC*SAT(TEMPK)/(RHOA*R*TEMPK)
        RLAPSE = GAMMAD*(1. + DUMMY1)/(1. + DUMMY1*HLC/(CP*TEMPK))
      END IF
      RETURN
      END

```

```

C-----

      SUBROUTINE VDEP(IFLG,DZ,RNO,V,I,VD)
      INCLUDE 'SCAV.FOR'
C
C

```

```

C ... 'VDEP' COMPUTES THE DIFFUSION VELOCITY FOR RAIN AND SNOW. THE
C   DIFFUSION VELOCITY IS BASED ON A PSEUDO-DIFFUSIVITY RESULTING
C   FROM A VARIATION OF SETTLING VELOCITY WITH PARTICLE SIZE.
C
C ... THE PSEUDO-DIFFUSIVITY IS GIVEN BY V-SQUARED BAR MINUS V-BAR
C   SQUARED WHERE V IS THE MEDIAN VOLUME FALL SPEED.
C
C ... VARIABLES IN CALL LIST:
C   IFLG=FLAG FOR ICE OR WATER PARTICLES
C   DZ=GRID INTERVAL (M)
C   RNO=No FOR SNOW AND No** 0.25 FOR RAIN
C   V=MEDIAN VOLUME PRECIPITATION FALL SPEED (M/S)
C   I=GRID INDEX
C   VD=COMPUTED DIFFUSION VELOCITY (M/S)
C
C ... FLAGS:
C   IFLG=1 WHEN COMPUTING FOR ICE
C       =2 WHEN COMPUTING FOR WATER
C
C   IF(IFLG .EQ. 1) GO TO 100
C
C ... HERE DIFFUSIVITY, D, AND V ARE FOR WATER
C   RC IS THE PRECIPITATION CONCENTRATION (G/M3)
C    $D = RC(I) ** 0.25 / RNO * 96.4 * DZ / V$ 
C   GO TO 101
C
C ... HERE DIFFUSIVITY, D, AND V ARE FOR SNOW
100   $D = (RC(I) / RNO) ** 0.1373 * 0.95 * DZ / V$ 
101  VD=0.
C
C ... BOUNDARY CONDITIONS. L=INDEX OF UPPER GRID
C
C   IF(I .EQ. 1) GO TO 200

```

```

C ... UPPER BOUNDARY-SET DIFFUSIVITY AND DEPOSITION VELOCITY=0
      IF( I .EQ. L) RETURN

```

```

C

```

```

C ... INTERIOR GRID VALUES
      VD=C012(I)*(RC(I+1)+RC(I))/(RC(I)+RC(I-1))
      GO TO 202

```

```

C

```

```

C ... LOWER BOUNDARY
200  VD=C012(I)/2.*(RC(I+1)+RC(I))/RC(I)

```

```

C

```

```

202  VD=-D/DZ*ALOG(VD)
      VQ=-VD  ! CORRECT FOR SIGN ERROR IN WATER
      RETURN
      END

```

```

C.....

```

```

      SUBROUTINE FALOUT(I,W,RC,QN0,QN02,RAYICE,XMO,VP,FALLOUT)

```

```

C

```

```

C ... COMPUTES MASS OF PRECIP, XMO, WHOSE DIA IS < FLOAT DIA.

```

```

C

```

```

C ... QN02=No**.25

```

```

C ... I=1 FOR RAIN

```

```

C ... I=2 FOR GRAUPEL

```

```

C

```

```

      XMO=RC

```

```

      FALLOUT=0.

```

```

      IF(RC .LT. 0.2) RETURN

```

```

      RNO=QN02**4.

```

```

      XLAMDA=42.10052*QN02/RC**0.25  ! LAMDA; No SAME FOR ICE, RAIN

```

```

      D=(W/130.)**2.  ! CUTOFF DIA

```

```

      DL=D*XLAMDA

```

```

      IF(I .EQ. 2) DL=DL*6.087523  ! INCLUDE ICE DENSITY AND FALLSPEED

```



```

E=EXP(-DL)
XK=5.23599E+5*RNO ! CONSTANT FROM MASS INTEGRATION
IF(I .EQ. 2) XK=XK*0.9 ! ACCOUNT FOR ICE DENSITY
FACT=DL+DL*DL/2.+DL*DL*DL/6.
XMO=6.*XK*(1.-E*(1.+FACT))/XLAMDA**4.

```

C

```

VPP=XMO**0.125/QNO*38.8496*RAYICE
FALLOUT=XMO*(W-VPP)-RC*(W-VP)
IF(FALLOUT .LT. 0.) RETURN
VP=VPP ! FALL SPEED OF XMD AT BIN TOP
RC=XMD ! MASS PASSING THROUGH AT BIN TOP
RETURN
END

```

C.....

```

SUBROUTINE CLOUD_BASE(T_RADM,Z_RADM,P_RADM,ZCB,RHG)
DIMENSION T_RADM(1),Z_RADM(1),P_RADM(1),QV_RADM(15)

```

```

c calculates lifting condensation level from lowest layer of RADM
c temperature profile (T_RADM(K), Kelvin) and mixing ratio
c (QV_RADM(K), g/g).
c Lifts parcel from this lowest layer up a dry adiabat to next
c pressure layer of RADM (with values P_RADM(K)
c If the sat'n mixing ratio of lifted parcel exceeds surface value of
c mixing ratio of parcel, then an interpolation over pressure is done
c to find the pressure height at which these values are equal (= sat'n
c level).
c Output variables: ZCB = height at cloud base, METERS
c                   PCB = pressure at cloud base, dynes/cm2
c                   TCB = temperature at cloud base, Kelvin
c Also does interpolation of other variables to this pressure level
c
c to use for applications other than RADM:

```



```
ZCB=ZCB/100. ! CONVERT TO METERS
```

```
return
```

```
end
```

```
C.....
```

```
function temp_lifted_dry(t_lower, dp, p_lower)
```

```
real*4 k1, k2, k3, k4
```

```
k1 = dTdPdry(p_lower, T_lower)
```

```
k2 = dTdPdry(p_lower + .5*dp, T_lower - .5 * dp * k1)
```

```
k3 = dTdPdry(p_lower + .5*dp, T_lower - .5 * dp * k2)
```

```
k4 = dTdPdry(p_lower + dp, T_lower - dp * k3)
```

```
delta_temp = (-dp/6) * (K1 + 2.*K2 + 2.*K3 + K4)
```

```
temp_lifted_dry = t_lower + delta_temp
```

```
return
```

```
end
```

```
C.....
```

```
function dTdPdry(press, tempk)
```

```
Rma = 287.1E4
```

```
cp = 1.004E7
```

```
Rhoair = press/(Rma*tempk)
```

```
dTdPdry = -1./(Rhoair*cp)
```

```
return
```

```
end
```

```
C.....
```

```
FUNCTION CESAT (TK)
```

```
C
```

```
C (ESAT IS SATURATION VAPOR PRESSURE IN DYNES/CM2
```

```
ESAT = 10.** (11.40 - (2353./TK) ) !Pascals (Fleagle & Businger,p72)
```

```

      CESAT = ESAT*10.                ! dynes/cm2
C
      RETURN
      END

C .....

      SUBROUTINE NUKE(I,VP,Z,H,DZ,XNC,DMDT,PART)
C
      INCLUDE 'SCAV.FOR'
C
C ... DEPOSITIONAL GROWTH AND NUCLEATION
C
      DMDT=0.
      IF(W(I) .LE. 0.) RETURN
      LL=I-1      !INTEGRATING UP
      M=I
C
101  VBAR=VP-(W(M)+W(LL))/2.
      KK=M
      IF(VBAR .LT. 0.) KK=LL
      SD=(SDEP(M)+SDEP(LL))/2.
      RO=RC(KK)
      IF(PART .EQ. 0.) RO=0.
C
      VBAR=ABS(VBAR)
      T1=273.16-TEMP(M)
      IF(T1 .LT. 0.) T1=0
      T0=273.16-TEMP(LL)
      IF(T0 .LT. 0.) T0=0.
      XNT=6.104E-3*(T1**4.+T0**4.)*.5    ! #/M3 NUCLEATED
      XNT=XNT**0.621*19.53*DZ
      SD=SD/XNC*XNT

```

```

C
200  IF(VBAR .LT. .01) VBAR=.01
      DMDT=((R0**0.62111802+SD/VBAR)**1.61-R0)*VBAR
      RETURN
      END

C .....

      SUBROUTINE SEARCH(I,C,RPBAR,RNO,BBP,Z,DZ,RAYICE,THRESH)
C
      INCLUDE 'SCAV.FOR'
C
C ... SEARCH FOR THE FREEZING LEVEL
C ... FREEZING LEVEL IS IN INTERVAL Z-DZ TO Z
C
      K=I-2  !INTEGRATING UP
      LL=I-1
      M=I-1
      N=I
C
C ... S REPRESENTS THE PRECIP CONCENTRATION AT THE FREEZING LEVEL
C   RPBAR IS THE HEIGHT WEIGHTED AVERAGE PRECIP IN THE INTERVAL
C   DZ.
C
      RSTAR=RC(N)
      IF(RSTAR .GT. THRESH) RSTAR=THRESH
      S=RC(K)+(RC(LL)-RC(K))*(HF-Z+2.*DZ)/DZ
      RPBAR=((S+RC(M))*(HF-Z+DZ)+(RSTAR+S)*(Z-HF))/2./DZ
      IF(RPBAR .LT. 0.) RPBAR=0.
C
C ... USE AVERAGE OF ACCRETION COEFFICIENTS NEAR FREEZING HT
      C1=0.
      IF(RC(M) .GT. 0.) C1=7.008E-4*RNO*EW*RC(M)**(-0.125)  ! WATER

```

```

C2=7.008E-4*BBP*EI*RAYICE/RSTAR**0.125  ! ICE
C=(C1+C2)/2.
RETURN
END

```

```

C .....
C ... BLOCK 'TABLE.FOR'
C
C
C ... RELEASED AUGUST, 1986
C
C ... FOR INFORMATION, CONTACT:
C   B.C. SCOTT OR C.M. BERKOWITZ
C   PACIFIC NORTHWEST LABORATORY
C   P.O. BOX 999, RICHLAND, WA. 99352
C
C
C ... LX=THE # OF VERTICAL USER GRIDS * # OF HEIGHT INTERVALS
C   * # OF PRECIP INTERVALS * # OF TEMP INTERVALS.
C   = (NV) * (NH) * (NP) * (NT)
C
C ... NVM= NUMBER OF VERTICAL GRIDS FROM THE CLOUD MODEL
C ... NVM1= ONE PLUS THE # OF VERTICAL GRIDS IN CLOUD MODEL
C   NV= NUMBER OF DESIRED VERTICAL GRIDS IN THE USER'S MODEL
C ... NH= NUMBER OF HEIGHT INTERVALS
C ... NP= NUMBER OF PRECIP INTERVALS
C ... NT= NUMBER OF TEMP INTERVALS
C           OR FREEZING LEVEL HEIGHT INTERVALS
C ... NVAR= NUMBER OF VARIABLES BEING SAVED FOR RADN MODEL
C
C   PARAMETER (NV=20, NVM=20, NH=4, NP=3, NT=3, NVAR=12)
C   PARAMETER (LX=NV*NH*NP*NT, NVM1=NVM+1, NC=NH*NP*NT, LY=2*LX+1)
C   PARAMETER (NSPACE=NVAR*NV)

```



```

COMMON RX(LY),RCX(LY),WX(LY),POUTX(LY)
COMMON ALPHX(LY),DEPOSX(LY),PRATEX(LY),FALLX(LY),AUTOX(LY)
COMMON FEN_CDX(LY),FCD_ENX(LY),FTOCELX(LY)
COMMON RI(NV),RCI(NV),WI(NV),POUTI(NV)
COMMON ALPHI(NV),DEPOSI(NV),PRATEI(NV),FALLI(NV),AUTOI(NV)
COMMON FEN_CDI(NV),FCD_ENI(NV),FTOCELI(NV)
COMMON RR(NV),RCR(NV),WR(NV),POUTR(NV)
COMMON ALPHR(NV),DEPOSR(NV),PRATER(NV),FALLR(NV),AUTOR(NV)
COMMON FEN_CDR(NV),FCD_ENR(NV),FTOCELR(NV)
COMMON RE(NVM),RCE(NVM),WE(NVM),POUTE(NVM)
COMMON ALPHE(NVM),DEPOSE(NVM),PRATEE(NVM),FALLE(NVM),AUTOE(NVM)
COMMON FEN_CDE(NVM),FCD_ENE(NVM),FTOCELE(NVM)

```

C

```

DIMENSION J(7),DV(6),VO(6)
DATA J /NV,NP,NH,NT,0,0,0/
DATA DV /4.,-2000.,1750.,0.,0.,0./
DATA VO /4.,12000.,1000.,0.,0.,0./
DATA HMAX /12000./

```

C

```

C ... J=DIMENSION OF EACH VARIABLE; J(1)=NV VERTICAL GRIDS,
C      J(2)=NP PRECIP INCREMENTS, ETC

```

C

```

C ... DV IS THE TABULATED INCREMENT FOR EACH OF THE VARIABLES
C      DV(1)=INCREMENT FOR VALUES OF PRECIP RATE (1.0 MM/HR)
C      OR CLOUD BASE HEIGHT (METERS)
C      DV(2)=INCREMENT FOR VALUES OF STORM HEIGHT (-2000 M)
C      DV(3)=INCREMENT FOR VALUES OF TEMPERATURE (10 DEGREES)
C      OR FREEZING LEVEL HEIGHT INCREMENT (M)
C      ETC. FOR OTHER INCREMENTS

```

C

```

C ... VO IS THE STARTING VALUE OF EACH VARIABLE; E.G.,
C      VO(1)= 0.0 MM/HR, VO(2)= 10,000 M, VO(3)= -5 DEG
C      OR V(3)= 1000 METERS

```

```

C ... ALL HEIGHTS ARE HEIGHTS ABOVE GROUND.

```

```

C
C ... HMAX IS THE MAXIMUM DESIRED HEIGHT OF THE USER'S GRID.
C     IF USER'S GRID IS A HEIGHT GRID
C .....

C
C     CALLED 'TABLEH' IN DIRECTORY
C
C
C ... INTERPOLATION ROUTINES FOR CONVECTIVE CLOUD MODULE
C
C ... RELEASED AUGUST, 1986
C
C ... FOR INFORMATION, CONTACT:
C     B.C. SCOTT OR C.M. BERKOWITZ
C     PACIFIC NORTHWEST LABORATORY
C     P.O. BOX 999, RICHLAND, WA. 99352
C
C .....
C
C ... THIS CODE WAS DEVELOPED ON A VAX 11-780 USING VMS OPERATING
C     SYSTEM. THE ONLY CHANGES NEED TO RUN IT ON ANOTHER SYSTEM
C     ARE THE FOLLOWING:
C
C     1. THE INCLUDE STATEMENT FOUND AT THE START OF EACH PROGRAM UNIT
C     2. DEVICE NUMBERS FOR INPUT-OUTPUT
C
C .....
C ... THIS PROGRAM FORMS A TABLE OF VALUES WHICH HAVE BEEN GENERATED BY
C     CONVECTIVE CLOUD ROUTINE. TABLE IS OUTPUT ON DEVICE 8.
C
C
C     INCLUDE 'TABLE.FOR'
C     COMMON R(NVM1),RC(NVM1),W(NVM1)
C     COMMON ALPH(NVM1),DEPOS(NVM1),PRATE(NVM1)

```

```

COMMON FALL(NVM1),AUTO(NVM1),POUT(NVM1)
COMMON FEN_CD(NVM1),FCD_EN(NVM1),FTOCEL(NVM1)
DIMENSION Q(6)

C
C ... FLAGS:
C     ICLOUD, ISTAB = 0  WHEN INTERPOLATION IS DONE FOR STABLE CLOUD OR FOR
C     RAINSHAFT PORTION OF CUMULUS CLOUD
C     ICLOUD, ISTAB = 1 WHEN INTERPOLATION IS FOR UPDRAFT PORTION OF
C     THE CUMULUS CLOUD
C
C
C ... FORM A TABLE FOR EACH OF THE DESIRED VALUES.  LOADING OF THE
C     VALUES IS RX(NV,NP,NH,NT)
C
C
C     INUNIT=11  ! ONE PLUS INPUT DEVICE NUMBER OF RAINSHAFT DATA
C     IOUT=8    ! OUTPUT DEVICE NUMBER FOR NEW TABLE
C
C     WRITE(6,806)
806  FORMAT(1H0,' TYPE 0 FOR STABLE CLOUD, 1 FOR CONVECTIVE CLOUD')
C     ACCEPT *, ICLOUD
C     ISTAB=ICLOUD
C
C     IUNIT = INUNIT
C
C ... STABLE CLOUD AND RAINSHAFT PART OF CONVECTIVE CLOUD WRITTEN ON
C     UNIT 10. UPDRAFT PORTION OF CONVECTIVE CLOUD, RAINRATE, STORM
C     DURATION WRITTEN ON UNIT 9.
C
C     LIM=ICLOUD+1
C
C     ICLOUD=-1
C     JJ=1
C     DO 600 NNN=1,LIM

```

```

ICLOUD=ICLOUD+1
IUNIT=IUNIT-1
KK=0
DO 400 M=1,NT
H=V0(2)
DO 100 K=1,NH
DO 200 L=1,NP
KK=KK+1
IF(ISTAB .EQ. 0) READ (IUNIT) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO
IF(ISTAB .EQ. 1) READ (IUNIT) R,RC,W,ALPH,DEPOS,PRATE,FALL,AUTO
1 ,FEN_CD,FCD_EN,FTOCEL
IF(ICLOUD .EQ. 1) READ (IUNIT) POUT,TIMEX(KK),RATEX(KK)
901 FORMAT(1H ,4(I3,1X),2(1X,1PE10.3))
DO 300 I=1,NVM
RE(I)=R(I)
RCE(I)=RC(I)
WE(I)=W(I)
ALPHE(I)=ALPH(I)
DEPOSE(I)=DEPOS(I)
PRATEE(I)=PRATE(I)
FALLE(I)=FALL(I)
AUTOE(I)=AUTO(I)
POUTE(I)=POUT(I)
IF(ISTAB .EQ. 0) GO TO 300
FEN_CDE(I)=FEN_CD(I)
FCD_ENE(I)=FCD_EN(I)
FTOCELE(I)=FTOCEL(I)
300 CONTINUE
C
C ... INTERPOLATE TO PUT ALL DATA ON USER VERTICAL GRID SPACING. TABLES
C ARE GENERATED FOR CLOUD WATER (RX), PRECIP WATER (RCX), AND
C UPDRAFT VELOCITY (WX). DZ IS THE USER GRID SPACING.
CALL HEIGHT(H,DZ,JJ,ICLOUD,ISTAB)
200 CONTINUE

```

```

100  H=H+DV(2)
400  CONTINUE
600  CONTINUE
C
      WRITE(IOUT) DZ,TIMEX,RATEX,RX,RCX,WX,POUTX,ALPHX,DEPOSX,
      1 PRATEX,FALLX,AUTOX,FEN_CD_X,FCD_ENX,FTOCELX
      WRITE (6,850)
850  FORMAT(1H ,' DONE')
      END

C .....

      SUBROUTINE HEIGHT(HMAXA,DZU,JJ,ICLOUD,ISTAB)
C      THIS ROUTINE INTERPOLATES DATA FROM AN AVAILABLE HEIGHT GRID
C      PRODUCED IN PROGRAM 'WETUP'. THE DESIRED HEIGHT SPACING IS
C      DETERMINED BELOW. CURRENTLY A LINEAR GRID SPACING IS USED IN EACH.
C
C
C ... RELEASED AUGUST, 1986
C
C ... FOR INFORMATION, CONTACT:
C      B.C. SCOTT OR C.M. BERKOWITZ
C      PACIFIC NORTHWEST LABORATORY
C      P.O. BOX 999, RICHLAND, WA. 99352
C
      INCLUDE 'TABLE.FOR'
C
C ... VARIABLES IN CALL LIST:
C      HMAXA=AVAILABLE MAX HEIGHT (FROM CLOUD MODEL)
C      DZU=DESIRED GRID SPACING IN USER MODEL
C      JJ=COUNTER FOR STORAGE IN INTERPOLATION TABLE
C      ISTAB=0 FOR A STABLE CLOUD, =1 FOR CONVECTIVE CLOUD
C

```

```

C ... OTHER VARIABLES
C     HMAX=MAX DESIRED HEIGHT (FOR USER'S MODEL)
C     DZC=AVAILABLE GRID SPACING IN CLOUD MODEL
C     DZU=DESIRED GRID SPACING IN USER MODEL
C     NV=NUMBER OF DESIRED GRIDS IN USER'S MODEL
C     NVM=NUMBER OF AVAILABLE GRIDS IN THE CLOUD MODEL
C
C     DZU=HMAX/FLOAT(NV-1)
C     DZC=HMAXA/FLOAT(NVM-1)
C
C     MAXC=NVM  ! MAX CLOUD MODEL GRID INDEX
C     MAXU=NV   ! MAX USER MODEL GRID INDEX
C
C     RX(JJ)=RE(1)  ! SET VALUES AT USER GRID BASE
C     RCX(JJ)=RCE(1)
C     WX(JJ)=WE(1)
C     ALPHX(JJ)=ALPHE(1)
C     DEPOSX(JJ)=DEPOSE(1)
C     PRATEX(JJ)=PRATEE(1)
C     FALLX(JJ)=FALLE(1)
C     AUTOX(JJ)=AUTOE(1)
C     IF(ICLOUD.EQ.1) POUTX(JJ)=POUTE(1)
C     IF(ISTAB .EQ. 0) GO TO 150
C     FEN_CDX(JJ)=FEN_CDE(1)
C     FCD_ENX(JJ)=FCD_ENE(1)
C     FTOCELX(JJ)=FTOCELE(1)
C
C
150  I=1  ! USER GRID INDEX
      L=2  ! CLOUD MODEL GRID INDEX
      ZC=0.  ! STARTING HEIGHT, CLOUD GRID
      ZU=DZU  ! STARTING HEIGHT, USER GRID
C
105  JJ=JJ+1  ! ARRAY INDEX
      RX(JJ)=0.

```

```

RCX(JJ)=0.
WX(JJ)=0.
ALPHX(JJ)=0.
DEPOSX(JJ)=0.
PRATEX(JJ)=0.
FALLX(JJ)=0.
AUTOX(JJ)=0.
IF(ICLOUD .EQ. 1) POUTX(JJ)=0.
IF(ISTAB .EQ. 0) GO TO 104
FEN_CDX(JJ)=0.
FCD_ENX(JJ)=0.
FTOCELX(JJ)=0.

```

C

```

104 IF(L .LE. MAXC) GO TO 201 ! IF TRUE, STILL CLD DATA AVAIL.
    I=I+1
    IF(I .GT. MAXU) GO TO 103 ! DONE, IF TRUE
    GO TO 105 ! ZERO VALUES ABOVE CLOUD TOP

```

C

```

201 IF(ZU .GT. (ZC+DZC)) GO TO 101 ! INCREMENT CLD GRID AND HT
    I=I+1 ! INCREMENT USER GRID INDEX
    IF(I .GT. MAXU) GO TO 103 ! DONE IF TRUE
    DELTA=(ZU-ZC)/DZC
    RX(JJ)=RE(L-1)+(RE(L)-RE(L-1))*DELTA
    RCX(JJ)=RCE(L-1)+(RCE(L)-RCE(L-1))*DELTA
    WX(JJ)=WE(L-1)+(WE(L)-WE(L-1))*DELTA
    ALPHX(JJ)=ALPHE(L-1)+(ALPHE(L)-ALPHE(L-1))*DELTA
    DEPOSX(JJ)=DEPOSE(L-1)+(DEPOSE(L)-DEPOSE(L-1))*DELTA
    PRATEX(JJ)=PRATEE(L-1)+(PRATEE(L)-PRATEE(L-1))*DELTA
    FALLX(JJ)=FALLE(L-1)+(FALLE(L)-FALLE(L-1))*DELTA
    AUTOX(JJ)=AUTOE(L-1)+(AUTOE(L)-AUTOE(L-1))*DELTA
    IF(ICLOUD.EQ.1) POUTX(JJ)=POUTE(L-1)+(POUTE(L)-POUTE(L-1
1 ))*DELTA
    IF(ISTAB .EQ. 0) GO TO 100
    FEN_CDX(JJ)=FEN_CDE(L-1)+(FEN_CDE(L)-FEN_CDE(L-1))*DELTA

```

```

      FCD_ENX(JJ)=FCD_ENE(L-1)+(FCD_ENE(L)-FCD_ENE(L-1))*DELTA
      FTOCELX(JJ)=FTOCELE(L-1)+(FTOCELE(L)-FTOCELE(L-1))*DELTA
100   ZU=ZU+DZU
      JJ=JJ+1
      GO TO 201
C
101   L=L+1
      ZC=ZC+DZC
      GO TO 104
C
103   RETURN
      END

```

```

C .....

```

```

C   CALLED PROGRAM 'BEGIN' (DUMMY MAIN PROGRAM)

```

```

C

```

```

C

```

```

C ... INTERPOLATION ROUTINES FOR CONVECTIVE CLOUD MODULE

```

```

C

```

```

C ... RELEASED AUGUST, 1986

```

```

C

```

```

C ... FOR INFORMATION, CONTACT:

```

```

C   B.C. SCOTT OR C.M. BERKOWITZ

```

```

C   BATTELLE PACIFIC NORTHWEST LABORATORY

```

```

C   P.O. BOX 999, RICHLAND, WA. 99352

```

```

C

```

```

C .....

```

```

C

```

```

C ... THIS CODE WAS DEVELOPED ON A VAX 11-780 USING VMS OPERATING

```

```

C   SYSTEM. THE ONLY CHANGES NEED TO RUN IT ON ANOTHER SYSTEM

```

```

C   ARE THE FOLLOWING:

```

```

C

```



```

C      1. THE INCLUDE STATEMENT FOUND AT THE START OF EACH PROGRAM UNIT
C      2. DEVICE NUMBERS FOR INPUT-OUTPUT
C
C.....

C      THIS PROGRAM INTERPOLATES FROM TABLES TO OBTAIN DESIRED
C      VERTICAL PROFILES AT SOME INTERMEDIATE INPUT CONDITION.
C
C      DIMENSION Q(6)
C      IIN=8 ! INPUT DEVICE NUMBER
C
C
C      WRITE(6,806)
806  FORMAT(1H0,' TYPE 0 FOR STABLE CLOUD, 1 FOR CONVECTIVE CLOUD')
C      ACCEPT *, ICLOUD
C
C      LIM=ICLOUD+1
C
C
C ... INTERPOLATE FROM TABLE TO GET VERTICAL PROFILES AT SOME INTER-
C      MEDIATE INPUT CONDITION. DESIRED INPUT CONDITIONS ARE STORED
C      IN ARRAY Q
C
C
C ... Q(1)=DESIRED SURFACE PRECP HOURLY AMOUNT (MM)
C      OR CLOUD BASE HEIGHT (METERS)
C ... Q(2)=DESIRED CLOUD TOP HEIGHT (METERS)
C ... Q(3)=DESIRED SURFACE TEMPERATURE (DEGREES C)
C *****
C
C      GIVES 5 TRIES, REMOVE DO 100 WHENEVER DESIRED
C
C *****
C      DO 100 L=1,5

```

```

        WRITE(6,805)
805  FORMAT(1H0,' DESIRED RAIN AMOUNT (MM) OR CLD BASE HT (M),
        1 CLOUD TOP (M), SFC TEMP (C)',/, ' OR FRZING LVL HT (M)')
        ACCEPT *,Q(1),Q(2),Q(3)
        IF(Q(1) .LT. 5 .OR . LIM .EQ. 2) WRITE(6,801) Q(1)
        IF(Q(1) .GT. 100.) WRITE(6,804) Q(1)
        WRITE(6,802) Q(2)
        WRITE(6,803) Q(3)
801  FORMAT(1H0,' SURFACE PRECIPITATION AMOUNT=',1PE10.3)
802  FORMAT(1H ,' CLOUD TOP HEIGHT=',1PE10.3)
803  FORMAT(1H ,' SURFACE TEMP OR FREEZING LVL HT=',1PE10.3)
804  FORMAT(1H0,' CLOUD BASE HEIGHT=',1PE10.3)
C
C ... DO THE INTERPOLATION AND OUTPUT THE INTERPOLATED VALUES
C   IN ARRAYS ENDING IN 'I' AND 'R'.
C
        CALL SELECT(LIM,Q,IIN)
100  CONTINUE
        END

C .....

        SUBROUTINE SELECT(LIM,Q,IIN)
C
C ... THIS ROUTINE INTERPOLATES FROM PREVIOUSLY GENERATED TABLES TO
C   GET DATA FOR CLOUD PARAMETERS.
C
C ... RELEASED AUGUST, 1986
C

```

```

C ... FOR INFORMATION, CONTACT:
C   B.C. SCOTT OR C.M. BERKOWITZ
C   PACIFIC NORTHWEST LABORATORY
C   P.O. BOX 999, RICHLAND, WA. 99352
C
C
C ... VARIABLES IN CALL LIST:
C   LIM = FLAG SET TO 1 FOR STABLE CLOUDS ,2 FOR CONVECTIVE CLOUDS
C   Q = ARRAY OF INPUT CONDITIONS, SEE BELOW
C   IIN= INPUT DEVICE NUMBER
C
C   INCLUDE 'TABLE.FOR'
C   DIMENSION Q(6)
C   DATA IREAD /0/
C
C   IF(IREAD .GT. 0.) GO TO 100
C ... INPUT TABLE INTO COMMON BLOCK
C
C   READ(IIN) DZ,TIMEX,RATEX,RX,RCX,WX,POUTX,ALPHX,DEPOSX,PRATEX,
1 FALLX,AUTOX,FEN_CD,X,FCD_ENX,FTOCELX
C   IREAD=1
C
C
100  ISTAB=LIM-1
C   ICLOUD=-1 ! START INTERPOLATION SECTION
C   DO 6000 NNN=1,LIM
C   ICLOUD=ICLOUD+1
C   IF(NNN .GT. 1) GO TO 602
C
C
C ... INTERPOLATE FROM TABLE TO GET VERTICAL PROFILES AT SOME INTER-
C   MEDIATE INPUT CONDITON. DESIRED INPUT CONDITIONS ARE STORED
C   IN ARRAY Q
C
C
C

```

```

C ... Q(1)=DESIRED SURFACE PRECP HOURLY AMOUNT (MM)
C      OR CLOUD BASE HEIGHT (METERS)
C ... Q(2)=DESIRED CLOUD TOP HEIGHT (METERS)
C ... Q(3)=DESIRED SURFACE TEMPERATURE (DEGREES C)
C      OR FREEZING LEVEL HEIGHT (M)
C
C
C      N=-1
C      DO 601 I=1,7
C      IF(J(I) .GT. 1) N=N+1
601  CONTINUE
C
602  CALL INTERP(Q,N,ICLOUD,ISTAB,TIMEI,RATEI)
C
C
C      IF(ICLOUD .EQ. 0) GO TO 650
C
C *****
C
C ..... ALL WRITE STATEMENTS CAN BE ELIMINATED AFTER DEBUGGING
C
C *****
C
C      WRITE(6,807) TIMEI,RATEI
807  FORMAT(1H0,' STORM DURATION (MIN) =',1PE10.3,' STORM RATE (MM
      1/HR) =',1PE10.3)
650  CONTINUE
C ... INTERPOLATED VALUES ARE STORED IN:
C      RI,RR=INTERPOLATED PROFILES OF CLOUD WATER
C      RCI,RCR=INTERPOLATED PROFILES OF PRECIP
C      WI,WR=INTERPOLATED PROFILES OF VERTICAL VELOCITY
C      ETC....
C

```

```

C      '..R(j) ' MEANS RAINSHAFT
C      '..I(j) ' MEANS UPDRAFT
C
      IF(LIM .EQ. 1) GO TO 651
      IF(NNN .EQ. 1) WRITE(6,808) ! CONVECTIVE CD HEADINGS
808    FORMAT(1H0,' RAINSHAFT CONVECTIVE CLOUD')
      IF(NNN .EQ. 2) WRITE(6,809)
809    FORMAT(1H0,' UPDRAFT CONVECTIVE CLOUD')
      WRITE(6,812)
      GO TO 652
C
651    WRITE(6,810) ! STABLE CLOUD HEADINGS
810    FORMAT(1H0,' STABLE CLOUD')
811    FORMAT(1H0)
      WRITE(6,812)
C
652    Z=0
812    FORMAT(1H0,'          Z(M)          R(G/G)          RC(G/G)          W(CM/S)
1      VP(CM/S) FTOCEL(CM2/S)')
813    FORMAT(1H),'          Z(M)          SCAV(1/S) PRATE(MM/H) P-DUMP
1      FEN_CD      FCD_EN')
814    FORMAT(1H ,38X,'(G/CM2/S) (CM2/S) (CM2/S)')
      INTER=2
      XINTER=FLOAT(INTER)
      DO 500 I=1,NV,INTER
      IF(ISTAB .EQ. 0) PRINT 800,I,Z,RI(I),RCI(I),WI(I),FALLI(I)
      IF(ISTAB .EQ. 1) PRINT 800,I,Z,RI(I),RCI(I),WI(I),FALLI(I)
1      ,FTOCELI(I)
500    Z=Z+DZ*XINTER
C
      WRITE(6,811)
      WRITE(6,813)
      WRITE(6,814)
      Z=0.

```

```

DO 501 I=1,NV,INTER
  IF(ISTAB .EQ. 0) PRINT 800,I,Z,ALPHI(I),PRATEI(I)
  IF(ISTAB .EQ. 1) PRINT 900,I,Z,ALPHI(I),PRATEI(I),POUTI(I)
  1 , FEN_CDI(I),FCD_ENI(I)
900  FORMAT(1H ,I3,6(1X,1PE10.3))
501  Z=Z+DZ*XINTER
      IF(LIM .EQ. 1) GO TO 6000
      IF(NNN .EQ. 2) GO TO 6000
      DO 675 I=1,NSPACE ! SAVE RAINSHAFT FLUXES
675  RR(I)=RI(I)
6000  CONTINUE
800  FORMAT(1H ,I3,6(1X,1PE10.3))
      RETURN
      END

```

C

SUBROUTINE INTERP(Q,N,ICLOUD,ISTAB,TIMEI,RATEI)

```

C
C ... 'INTERP' INTERPOLATES FROM A TABLE OF VALUES TO FIND VERTICAL
C   PROFILES OF R AT GIVEN VALUES OF CLOUD TOP HEIGHT, SURFACE
C   PRECIP RATE, AND SURFACE TEMPERATURE. THE INTERPOLATED VALUES
C   ARE PLACED IN ARRAYS RI, RCI, WI, TI
C
C
C ... RELEASED AUGUST, 1986
C
C ... FOR INFORMATION, CONTACT:
C   B.C. SCOTT OR C.M. BERKOWITZ
C   PACIFIC NORTHWEST LABORATORY
C   P.O. BOX 999, RICHLAND, WA. 99352
C

```

```

        INCLUDE 'TABLE.FOR'
        DIMENSION V(6),DJ(6),M(7),L(128),Q(6)
        DIMENSION IA(128)
        INTEGER PROD

C
C
C ... VARIABLES IN CALL LIST:
C   Q= ARRAY OF POINTS WHERE INTERPOLATION IS DESIRED
C   N=NUMBER OF LEVELS FOR INTERPOLATION, I.E.; PRECIP,
C     STORM HEIGHT, SURFACE TEMPERATURE,.....
C   ICLOUD= SEE 'FLAGS' BELOW
C   ISTAB= SEE 'FLAGS' BELOW
C   TIMEI=STORM DURATION (MIN)-COMPUTED
C   RATEI=STORM RATE (MM/H)-COMPUTED
C
C ... FLAGS:
C   ICLOUD, ISTAB = 0  WHEN INTERPOLATION IS DONE FOR STABLE CLOUD OR FOR
C     RAINSHAFT PORTION OF CUMULUS CLOUD
C   ICLOUD, ISTAB = 1 WHEN INTERPOLATION IS FOR UPDRAFT PORTION OF
C     THE CUMULUS CLOUD
C
C
C ... RX=TABLE OF VALUES, RI= INTERPOLATED VALUES
C ... RX=RX(Z,XJ,H,T,...)=RX(J1,J2,J3,J4,...) WHERE:
C   J1=J(1)= # OF VERTICAL GRIDS,   J3=J(3)= # OF CLOUD TOP INTERVALS,
C   J2=J(2)= # OF PRECIP INTERVALS, J4=J(4)= # OF TEMPERATURE
C     INTERVALS, ETC.
C   THUS, J=DIMENSION OF EACH VARIABLE; J(1)=NV VERTICAL GRIDS,
C   J(2)=NP PRECIP INCREMENTS, ETC.
C
C ... N=NUMBER OF VARIABLES BEING CONSIDERED
C   VARIABLES ARE (IN ORDER):
C     PRECIPITATION RATE
C     STORM HEIGHT

```

```

C          SURFACE TEMPERATURE
C          OTHERS AS DESIRED
C ..... MAX N=6 .....
C
C ... 3RD ORDER INTERPLATION IS USED, THUS RX AT SOME
C INTERPOLATED POINT IS (AT A FIXED HEIGHT):
C
C       $RI = R_o + DR/D(XJ) * (XJ - XJ_o) + DR/DH * (H - H_o) + DR/DT * (T - T_o) + \dots$ 
C      + HIGHER ORDER TERMS
C
C ... V(I)=VALUE OF ith VARIABLE AT THE BEGINNING OF THE PROPER INTERVAL,
C      V(1)=XJo, V(2)=Ho, V(3)=To ETC.
C
C ... Vo IS THE STARTING VALUE OF EACH VARIABLE; E.G.,
C      Vo(1)= 0.0 MM/HR, Vo(2)=10,000 M, Vo(3)= -5 DEG
C
C ... DV IS THE TABULATED INCREMENT FOR EACH OF THE VARIABLES
C      DV(1)=INCREMENT FOR VALUES OF PRECIP RATE (1.0 MM/HR)
C      DV(2)=INCREMENT FOR VALUES OF STORM HEIGHT (-2000 M)
C      DV(3)=INCREMENT FOR VALUES OF TEMPERATURE (10 DEGREES)
C      ETC FOR OTHER INCREMENTS
C
C ... Q(I) IS THE DESIRED VALUE OF THE ith VARIABLE WHICH IS PRESUMABLY
C      SOMEWHERE BETWEEN THE TABULATED VALUES. Q(1)=DESIRED PRECIP RATE,
C      Q(2)=DESIRED STORM HEIGHT, ETC.
C
C      NN=N+1
C
C ... MAKE SURE DESIRED VALUES ARE WITHIN LOWER TABLE LIMITS
C      DO 550 K=1,N
C      QSAVE=Q(K)
C      IF(QSAVE .EQ. 0) QSAVE=1.E-8
C      IF(DV(K) .GT. 0.) GO TO 551
C ... HERE Q MUST BE < Vo TO START

```



```

      IF(Q(K) .LT. V0(K)) GO TO 550
      IF(V0(K) .GT. 0.) Q(K)=0.9999999*V0(K)
      IF(V0(K) .LT. 0.) Q(K)=1.00000001*V0(K)
      DELTQ=(QSAVE-Q(K))/QSAVE
      IF(ABS(DELTQ) .GT. 4.E-7 ) WRITE(6,800) K,Q(K)
      GO TO 550

C
C ... HERE Q MUST BE > V0 TO START
551  IF(Q(K) .GT. V0(K)) GO TO 550
      IF(V0(K) .GT. 0.) Q(K)=1.0000001*V0(K)
      IF(V0(K) .LT. 0.) Q(K)=0.9999999*V0(K)
      DELTQ=(QSAVE-Q(K))/QSAVE
      IF(ABS(DELTQ) .GT. 4.E-7 ) WRITE(6,800)K,Q(K)
550  CONTINUE
C
      DO 200 K=1,N
101  V(K)=V0(K)
      QSAVE=Q(K)
      IUP=J(K+1)
      DO 100 I=1,IUP
      IF(DV(K) .GT. 0. .AND. V(K) .GT. Q(K)) GO TO 200
      IF(DV(K) .LT. 0. .AND. V(K) .LT. Q(K)) GO TO 200
      M(K+1)=I
      DJ(K)=Q(K)-V(K)
100  V(K)=V(K)+DV(K)
C
C ... OUT OF TABLE LIMITS.  SELECT A DEFAULT VALUE
      IF(DV(K) .GT. 0.) Q(K)=0.9999999*(V(K)-DV(K))
      IF(DV(K) .LT. 0.) Q(K)=1.0000001*(V(K)-DV(K))
      DELTQ=(QSAVE-Q(K))/QSAVE
      IF(ABS(DELTQ) .GT. 4.E-7 ) WRITE(6,800) K,Q(K)
200  FORMAT(1H0,' OUT OF TABLE LIMITS FOR VARIABLE K=',I2,/,
1    ' DEFAULT OF Q(K) IS TAKEN TO BE ',2(1X,1PE10.3))
      GO TO 101

```

```

C
200  DJ(K)=DJ(K)/DV(K)
C
C ... FIND LOCATION OF STARTING INDEX ( L(1) )
C
      PROD=0.
      DO 600 I=1,N
      K=NN-I
600  PROD=(PROD+(M(K+1)-1))*J(K)
901  FORMAT(1H ,6(1X,I3))
      IUP=J(1)
C ... INTERPOLATE FOR EACH HEIGHT LEVEL (I)
      DZ1=HMAX/FLOAT(IUP-1)
C
C ... INDICIES FOR TIME & DURATION IN CONVECTIVE CLOUD
      IA(1)=PROD/J(1)+1
      ISUM=1
      DO 402 K=2,4
      IA(K)=IA(1)+ISUM
402  ISUM=ISUM*J(K)
C
      IA(5)=IA(2)+IA(3)-IA(1)
      IA(6)=IA(3)+IA(4)-IA(1)
      IA(7)=IA(2)+IA(4)-IA(1)
      IA(8)=IA(2)+IA(3)+IA(4)-2*IA(1)
      Z=0.
      RATEI=RATEX(IA(1))
      IF(ICLOUD .EQ. 1) PROD=LX+PROD
      DO 300 I=1,IUP
      RCI(I)=0.
      WI(I)=0.
      ALPHI(I)=0.
      DEPOSI(I)=0.
      PRATEI(I)=0.

```

```

    FALLI(I)=0.
    AUTOI(I)=0.
    POUTI(I)=0.
    IF(ISTAB .EQ. 0) GO TO 450
    FEN_CDI(I)=0.
    FCD_ENI(I)=0.
    FTOCELI(I)=0.
450  IF(Z .GT. Q(2)) GO TO 300
    L(1)=PROD+I
    ISUM=J(1)
    DO 400 K=2,4
    L(K)=L(1)+ISUM
400  ISUM=ISUM*J(K)
C
C ... COMPUTE INTERPOLATED VALUE. PUT AS MANY VARIABLES AS DESIRED.
C
    R01=RX(L(1))
    R02=RCX(L(1))
    R03=WX(L(1))
    R04=ALPHX(L(1))
    R05=DEPOSX(L(1))
    R06=PRATEX(L(1))
    R07=FALLX(L(1))
    R08=AUTOX(L(1))
    R12=POUTX(L(1))
    IF(ISTAB .EQ. 0) GO TO 451
    R09=FEN_CDX(L(1))
    R10=FCD_ENX(L(1))
    R11=FTOCELX(L(1))
451  RI(I)=R01
    RCI(I)=R02
    WI(I)=R03
    ALPHI(I)=R04
    DEPOSI(I)=R05

```

```

PRATEI(I)=R06
FALLI(I)=R07
AUTOI(I)=R08
POUTI(I)=R12
IF(ISTAB .EQ. 0) GO TO 452
FEN_CDI(I)=R09
FCD_ENI(I)=R10
452 DO 500 K=1,N
CALL ORDER1(RI(I),RX,L,K,R01,DJ)
CALL ORDER1(RCI(I),RCX,L,K,R02,DJ)
CALL ORDER1(WI(I),WX,L,K,R03,DJ)
CALL ORDER1(ALPHI(I),ALPHX,L,K,R04,DJ)
CALL ORDER1(DEPOSI(I),DEPOSX,L,K,R05,DJ)
CALL ORDER1(PRATEI(I),PRATEX,L,K,R06,DJ)
CALL ORDER1(FALLI(I),FALLX,L,K,R07,DJ)
IF(ICLOUD.EQ.1) CALL ORDER1(POUTI(I),POUTX,L,K,R12,DJ)
IF(ISTAB .EQ. 0) GO TO 453
CALL ORDER1(FEN_CDI(I),FEN_CDX,L,K,R09,DJ)
CALL ORDER1(FCD_ENI(I),FCD_ENX,L,K,R10,DJ)
CALL ORDER1(FTOCELI(I),FTOCELX,L,K,R11,DJ)
453 IF(I .GT. 1 .OR. ICLOUD .EQ. 0) GO TO 500
CALL ORDER1(RATEI,RATEX,IA,K,RATEX(IA(1)),DJ)
500 CALL ORDER1(AUTOI(I),AUTOX,L,K,R08,DJ)
C
C ... INDICIES FOR SECOND AND THIRD ORDER TERMS
L(5)=L(2)+L(3)-L(1)
L(6)=L(3)+L(4)-L(1)
L(7)=L(2)+L(4)-L(1)
L(8)=L(2)+L(3)+L(4)-2*L(1)
DJH=DJ(1)*DJ(2)
DTH=DJ(3)*DJ(2)
DTJ=DJ(3)*DJ(1)
DTJH=DTJ*DJ(2)

```

C

C ... SECOND ORDER TERMS

C

```
CALL ORDER21(RI(I),RX,L,DJH)
CALL ORDER21(RCI(I),RCX,L,DJH)
CALL ORDER21(WI(I),WX,L,DJH)
CALL ORDER21(ALPHI(I),ALPHX,L,DJH)
CALL ORDER21(DEPOSI(I),DEPOSX,L,DJH)
CALL ORDER21(PRATEI(I),PRATEX,L,DJH)
IF(ICLOUD.EQ.1) CALL ORDER21(POUTI(I),POUTX,L,DJH)
IF(ISTAB .EQ. 0) GO TO 454
CALL ORDER21(FEN_CDI(I),FEN_CDX,L,DJH)
CALL ORDER21(FCD_ENI(I),FCD_ENX,L,DJH)
CALL ORDER21(FTOCELI(I),FTOCEX,L,DJH)
454 IF(I .GT. 1 .OR. ICLOUD .EQ. 0) GO TO 601
CALL ORDER21(RATEI,RATEX,IA,DJH)
601 CALL ORDER21(FALLI(I),FALLX,L,DJH)
CALL ORDER21(AUTOI(I),AUTOX,L,DJH)
```

C

```
CALL ORDER22(RI(I),RX,L,DTH)
CALL ORDER22(RCI(I),RCX,L,DTH)
CALL ORDER22(WI(I),WX,L,DTH)
CALL ORDER22(ALPHI(I),ALPHX,L,DTH)
CALL ORDER22(DEPOSI(I),DEPOSX,L,DTH)
CALL ORDER22(PRATEI(I),PRATEX,L,DTH)
IF(ICLOUD.EQ.1) CALL ORDER22(POUTI(I),POUTX,L,DTH)
IF(ISTAB .EQ. 0) GO TO 455
CALL ORDER22(FEN_CDI(I),FEN_CDX,L,DTH)
CALL ORDER22(FCD_ENI(I),FCD_ENX,L,DTH)
455 IF(I .GT. 1 .OR. ICLOUD .EQ. 0) GO TO 602
CALL ORDER22(RATEI,RATEX,IA,DTH)
602 CALL ORDER22(FALLI(I),FALLX,L,DTH)
```

C

```
CALL ORDER23(RI(I),RX,L,DTJ)
CALL ORDER23(RCI(I),RCX,L,DTJ)
```

```

CALL ORDER23(WI(I),WX,L,DTJ)
CALL ORDER23(ALPHI(I),ALPHX,L,DTJ)
CALL ORDER23(DEPOSI(I),DEPOSX,L,DTJ)
CALL ORDER23(PRATEI(I),PRATEX,L,DTJ)
IF(ICLOUD.EQ.1) CALL ORDER23(POUTI(I),POUTX,L,DTJ)
IF(ISTAB .EQ. 0) GO TO 456
CALL ORDER23(FEN_CDI(I),FEN_CDX,L,DTJ)
CALL ORDER23(FCD_ENI(I),FCD_ENX,L,DTJ)
CALL ORDER23(FTOCELI(I),FTOCELX,L,DTJ)
456 IF(I .GT. 1 .OR. ICLOUD .EQ. 0) GO TO 603
CALL ORDER23(RATEI,RATEX,IA,DTJ)
603 CALL ORDER23(FALLI(I),FALLX,L,DTJ)
CALL ORDER23(AUTOI(I),AUTOX,L,DTJ)
C
C ... THIRD ORDER TERMS
C
CALL ORDER3(RI(I),RX,L,DTJH)
CALL ORDER3(RCI(I),RCX,L,DTJH)
CALL ORDER3(WI(I),WX,L,DTJH)
CALL ORDER3(ALPHI(I),ALPHX,L,DTJH)
CALL ORDER3(DEPOSI(I),DEPOSX,L,DTJH)
CALL ORDER3(PRATEI(I),PRATEX,L,DTJH)
CALL ORDER3(FALLI(I),FALLX,L,DTJH)
CALL ORDER3(AUTOI(I),AUTOX,L,DTJH)
IF(ICLOUD.EQ.1) CALL ORDER3(POUTI(I),POUTX,L,DTJH)
IF(ISTAB .EQ. 0) GO TO 457
CALL ORDER3(FEN_CDI(I),FEN_CDX,L,DTJH)
CALL ORDER3(FCD_ENI(I),FCD_ENX,L,DTJH)
CALL ORDER3(FTOCELI(I),FTOCELX,L,DTJH)
457 IF(I .GT. 1 .OR. ICLOUD .EQ. 0) GO TO 301
CALL ORDER3(RATEI,RATEX,IA,DTJH)
301 IF(FEN_CDI(I) .LT. 0.) FEN_CDI(I)=0. ! REMOVE INTERPOLATION ERROR
IF(FCD_ENI(I) .LT. 0.) FCD_ENI(I)=0.
IF(ABS(PRATEI(I)) .LT. 1.E-7) PRATEI(I)=0. ! REMOVE TRUNCATION ERROR

```

```

      POUTI(I)=POUTI(I)*1.E-4  ! CONVERT TO G/CM2/S
300  Z=Z+DZ1
      IF(ICLOUD .GT. 0) TIMEI=Q(1)/RATEI*60.
      RETURN
      END

```

```

      SUBROUTINE ORDER1(RI,RX,L,K,R01,DJ)
      DIMENSION RX(1),L(1),DJ(1)
      RI=RI+(RX(L(K+1))-R01)*DJ(K)
      RETURN
      END

```

```

      SUBROUTINE ORDER21(RI,RX,L,DJH)
      DIMENSION RX(1),L(1)
      RI=RI+DJH*(RX(L(5))-RX(L(2))-RX(L(3))+RX(L(1)))
      RETURN
      ENO

```

```

      SUBROUTINE ORDER22(RI,RX,L,DTH)
      DIMENSION RX(1),L(1)
      RI=RI+DTH*(RX(L(6))-RX(L(4))-RX(L(3))+RX(L(1)))
      RETURN
      END

```

```

      SUBROUTINE ORDER23(RI,RX,L,DTJ)
      DIMENSION RX(1),L(1)
      RI=RI+DTJ*(RX(L(7))-RX(L(4))-RX(L(2))+RX(L(1)))
      RETURN
      END

```

```
SUBROUTINE ORDER3(RI,RX,L,DTJH)
DIMENSION RX(1),L(1)
RI=RI+DTJH*(RX(L(8))-RX(L(7))-RX(L(6))+RX(L(4))
1      -RX(L(5))+RX(L(2))+RX(L(3))-RX(L(1)))
RETURN
END
```

\$

DISTRIBUTIONNo. of
CopiesNo. of
CopiesOFFSITE30 DOE Technical Information Center

J. Chang
National Center for Atmospheric
Research
P.O. Box 3000
Boulder, CO 80307

J. F. Clarke
U.S. Environmental Protection
Agency
Atmospheric Sciences
Research Laboratory
Research Triangle Park, NC 27711

R. E. Dennis
U.S. Environmental Protection
Agency
Atmospheric Sciences
Research Laboratory
Research Triangle Park, NC 27711

J. L. Durham
U.S. Environmental Protection
Agency
Atmospheric Sciences
Research Laboratory
Research Triangle Park, NC 27711

R. G. Lamb
U.S. Environmental Protection
Agency
Atmospheric Sciences
Research Laboratory
Research Triangle Park, NC 27711

F. G. Vukovich
Research Triangle Park
P.O. Box 121
Research Triangle Park, NC 27711

C. J. Walcek
National Center for Atmospheric
Research
P.O. Box 3000
Boulder, CO 80307

ONSITE2 DOE Richland Operations Office

R. A. Rochette
J. J. Sutey

25 Pacific Northwest Laboratory

C. M. Berkowitz (5)
E. G. Chapman
S. F. Crowder
D. W. Dragnich
R. C. Easter
L. K. Grove
J. M. Hales
P. C. Hays
O. J. Luecken
N. S. Laulainen
W. T. Pennell
B. C. Scott
J. A. Stottlemire
R. E. Wildung
Publishing Coordination (2)
Technical Report Files (5)