Documentation of Radiation and Cloud Routines in the NCAR Community Climate Model (CCM1)

JEFFREY T. KIEHL
RICHARD J. WOLSKI
BRUCE P. BRIEGLEB
V. RAMANATHAN
# TABLE OF CONTENTS

Section 1. Physical Parameterization of Radiation ........................................ 1-1
Section 2. Parameterization of Clouds ......................................................... 2-1
Section 3. Code Packages for Physical Parameterizations .............................. 3-1
Section 4. Radiation Subroutine Descriptions .............................................. 4-1
Section 5. Radiation Common Block Descriptions ......................................... 5-1
Section 6. Column Model Descriptions ......................................................... 6-1
ACKNOWLEDGEMENTS

The authors would like to thank Robert Dickinson and Julia Slingo for very perceptive comments on an earlier version of this manuscript. Patrick Downey contributed to the development of a number of radiation routines. The authors are indebted to the editing skills of Cynthia Vangor Emory and Eileen Boettner; their work has made the compilation of this document seem effortless. We would also like to thank Linda Bath for supplying Sections 4 and 5 of this document and for providing invaluable assistance in the editing of the document. Jeffrey T. Kiehl's efforts on this document were supported through a grant from the Department of Energy, DE–AI03–84SF12208.
PREFACE

This document describes the radiation and cloud algorithms employed in the NCAR Community Climate Model (CCM1). The document is divided into six sections. Section 1 describes the physical parameterization and finite difference techniques for both solar radiation and longwave radiation. Section 2 describes the parameterization of clouds. Section 3 gives detailed flow descriptions of the radiation and cloud routines, and also describes their dependence on the rest of the model subroutines. Section 4 provides descriptions of each subroutine. Section 5 lists the common blocks used in these subroutines and the variables associated with these common blocks. A column radiation model based on the CCM1 radiation code is also available to users. This model is designed to be used to calculate radiative quantities given certain input data. A description of the structure of this model is given in Section 6. The column model is available to users on request.

Sections 1 and 2 of this document also appear in Description of NCAR Community Climate Model (CCM1) by Williamson et al.(1987). Section 3 appears in Users' Guide to NCAR CCM1 by Bath et al.(1987). Sections 4 and 5 appear in Documentation of NCAR CCM1 Program Modules by Bath et al.(1987). For further information on CCM1 and the interface between the radiation and cloud routines, the reader is urged to consult these documents.
Physical Parameterization of Radiation

Solar Radiation

The radiative flux in the solar spectral region is divided into two regions. Radiation between 0–0.9 μm is denoted as ultraviolet and visible (UVV), while radiation between 0.9–4.0 μm is denoted as near-infrared (NIR). The net downward flux for these two intervals is $F_{UVV}^S$ and $F_{NIR}^S$, respectively. Each of these fluxes is, in turn, separated into three components—clear sky, nonoverlapped clouds, and overlapped clouds. These are denoted $F_{UVV0}^S$, $F_{UVV1}^S$, and $F_{UVVM}^S$ for the ultraviolet and visible wavelength region and $F_{NIR0}^S$, $F_{NIR1}^S$, and $F_{NIRM}^S$ for the near-infrared region.

Solar Insolation

The solar insolation at the top of the model atmosphere is determined by

$$S_I = S_o \cos \zeta f_d \epsilon,$$

where $S_o$ is the solar constant, $\zeta$ is the solar zenith angle, $f_d$ is the fractional amount of daylight, and $\epsilon$ is the eccentricity factor which is dependent on the time of year. $f_d$ and $\zeta$ depend on latitude. These quantities are calculated from information on the calendar day. The value of $S_o$ is 1370 W m$^{-2}$. No diurnal cycle is included in the standard version of the model.

Cloud Cover

Fractional cloud cover is calculated assuming random overlap of cloud layers. The clear–sky fraction for the total atmospheric column is given by

$$f_{clear} = \prod_{i=1}^{N} (1 - A_i),$$

where $A_i$ is the fractional cloud cover in layer $i$ and $N$ is the total number of atmospheric layers. The fractional cloud cover is calculated based on the convective parameterization and is described in Section 2.

The level structure employed for radiation calculations is shown in Figure 1.1. Note that the level indices for the radiation calculations are the inverse to those used by the rest of CCM1. Fluxes are calculated on half–sigma levels, while heating rates are calculated on full–sigma levels. Clouds are formed between half–sigma levels, and thus centered about full–sigma levels.
Figure 1.1. Level structure for radiation calculations.
The two other cloudy–sky conditions that are needed for the solar calculations are the nonoverlapped and overlapped cases. The nonoverlapped (single layered) case is defined as that fraction of the atmosphere having a cloud in a given layer \( k \), but clear sky above and below the cloud; the fraction of sky \( f_{1L}(k) \) with this condition is

\[
f_{1L}(k) = A_k \prod_{i \neq k}^N (1 - A_i).
\]

The overlapped (multiply layered) cloud case is defined as that fraction of the atmosphere having a cloud in layer \( k \) overlain by cloud in one or more higher levels and clear sky below the cloud,

\[
f_{ML}(k) = A_k \prod_{i=k+1}^N (1 - A_i) \left\{ 1 - \prod_{i=1}^{k-1} (1 - A_i) \right\}
= A_k \prod_{i=k+1}^N (1 - A_i) \left\{ 1 - \frac{\prod_{i=1}^N (1 - A_i)}{\prod_{i=k-2}^N (1 - A_i)} \right\}
\]

\[
f_{ML}(k) = A_k \prod_{i=k+1}^N (1 - A_i) \left\{ 1 - \frac{f_{\text{clear}}}{\prod_{i=k-2}^N (1 - A_i)} \right\}.
\]

Absorption Due to Gases

Fractional absorption in the ultraviolet and visible spectral regions due to ozone defined as \( \xi_{O_3}(p) \) is obtained from the expressions of Lacis and Hansen (1974). Absorption in the near–infrared spectral region by water vapor \( \xi_{H_2O}(p) \) is calculated from the parameterization of Kratz and Cess (1985) for the direct beam, and from the formulation of Lacis and Hansen (1974) for the reflected beam. The use of these two different formulations is for computational efficiency. Absorption in the near–infrared region due to carbon dioxide \( \xi_{CO_2}(p) \) is calculated from the expression of Sasamori et al. (1972). This expression agrees very well with the newer absorptance formulation of Kiehl et al. (1972) below 10 mb. However, above 10 mb, the Sasamori et al. expression underestimated near–infrared CO\(_2\) heating. Absorption due to oxygen \( \xi_{O_2}(p) \) is based on the parameterization of Kiehl and Yamanouchi (1985).

Clear Sky

The approach of modeling clear–sky flux at the surface follows Lacis and Hansen (1974) and accounts for direct–beam solar radiation reaching the surface and reflection of diffuse radiation between the surface and molecular (Rayleigh) scatter from the atmosphere.
For the ultraviolet and visible spectral region, this is given by

\[ F_{UVV0}^S = S_I f_{\text{clear}} (0.647 - \xi_{O_3}(p_s) - \alpha_{R}^{dr}) \left[ (1.0 - \alpha_{UVV}^{dr}) + \frac{\alpha_{UVV}^{df}(1.0 - \alpha_{UVV}^{df})}{1.0 - \alpha_{UVV}^{df} \alpha_{R}^{df}} \right], \]

where \( S_I \) is the solar insolation, \( f_{\text{clear}} \) is the fractional amount of clear sky, \( \xi_{O_3}(p_s) \) is the absorption due to ozone from the top of the atmosphere to the surface \( (p_s) \), and \( \alpha_{R}^{dr} \) is the direct-beam Rayleigh albedo,

\[ \alpha_{R}^{dr} = \frac{0.28}{1.0 + 6.43 \cos \zeta}, \]

where \( \zeta \) is the zenith angle. The factor 0.647 accounts for the fractional amount of energy of the solar spectrum that lies between 0 and 0.9 \( \mu \)m. The term \( (1.0 - \alpha_{UVV}^{df}) \) accounts for the amount of direct-beam ultraviolet and visible radiation absorbed at the surface; the second term accounts for the amount of diffuse radiation absorbed at the surface due to multiple reflections between the surface and Rayleigh scatterers in the atmosphere; \( \alpha_{UVV}^{df} \) is the direct-beam surface albedo for the ultraviolet and visible spectral region. Downward Rayleigh scattered radiation is included in the direct beam. This factor depends upon the following surface types: ocean, land, sea ice, and snow. \( \alpha_{UVV}^{df} \) is the diffuse-beam surface albedo, and \( \alpha_{R}^{df} \) is the diffuse-beam Rayleigh albedo,

\[ \alpha_{R}^{df} = 0.0685. \]

The near-infrared clear-sky net surface flux is

\[ F_{NIR0}^S = S_I f_{\text{clear}} (0.353 - \xi_{H_2O}(p_s) - \xi_{CO_2}(p_s) - \xi_{O_2}(p_s)) (1 - \alpha_{NIR}^{dr}). \]

The 0.353 factor accounts for the fractional amount of solar radiation between 0.9 and 4 \( \mu \)m. The terms \( \xi_{H_2O}, \xi_{CO_2}, \) and \( \xi_{O_2} \) are the absorptivities for \( H_2O, CO_2, \) and \( O_2 \), respectively. The absorption is over the total atmospheric column. (N.B. Although \( O_2 \) absorbs in the visible, the \( \xi_{O_2} \) term is included in the near-infrared region for simplicity, since accounting for multiple scattering with \( O_2 \) absorption would complicate the ultraviolet and visible calculations.) The term \( \alpha_{NIR}^{dr} \) is the near-infrared direct-beam surface albedo.

**Nonoverlapped Fraction of Clouds**

The contribution to the surface flux from the ultraviolet and visible spectral regions where nonoverlapped clouds are present is

\[ F_{UVV1}^S = S_I (0.647 - \xi_{O_3}(p_s)) (1.0 - \alpha_{UVV}^{df}) \sum_k \frac{(1.0 - \alpha_{c}^{dr}(k))}{(1.0 - \alpha_{c}^{df}(k) \alpha_{UVV}^{df})} f_L(k), \]
where the summation is over all layers containing clouds, while $f_{1L}(k)$ is the fraction in any of these layers that contains nonoverlapped cloud; $\alpha_c^{dr}(k)$ is the direct-beam cloud albedo,

$$\alpha_c^{dr}(k) = \frac{\alpha_{cl}(k)}{\alpha_{cl}(k) + \cos \xi},$$  \hspace{1cm} (10)

where $\alpha_{cl}(k)$ is a parameter dependent on cloud level. Presently, $\alpha_{cl}(k)$ can assume the values 0.6, 0.3, and 0.15 for levels 1-4, levels 5 and 6, and levels 7-13, respectively. The term $\alpha_c^{df}(k)$ is the diffuse-beam cloud albedo,

$$\alpha_c^{df}(k) = \frac{\alpha_{cl}(k)}{\alpha_{cl}(k) + 0.5}.$$ \hspace{1cm} (11)

The term in Eq. (9) after the summation sign accounts for attenuation by the clouds multiple scattering between the surface ($\alpha_c^{df}(k)$) and the cloud layer.

For the near-infrared spectral region, water-vapor absorption by the cloud is accounted for by enhancing the path length; for this region, the near-infrared surface flux for nonoverlapped clouds is

$$F_{NIR1}^S = S_1(1.0 - \alpha_{NIR}^{df}) \sum_k \frac{(1.0 - \alpha_c^{dr}(k))}{1.0 - \alpha_c^{df}(k)\alpha_{NIR}^{df}} (0.353 - \xi_{CO_2}(p_s)) .$$ \hspace{1cm} (12)

$$\xi_{CO_2}(p_s) = \xi_{H_2O}(p_{k+1}) f_{1L}(k).$$

$\alpha_{NIR}^{df}$ is the diffuse-beam near-infrared surface albedo. The term $\xi_{H_2O}(p_{k+1})$ is the water-vapor absorption for a path from the top of the atmosphere to the cloud top plus an enhanced path through the cloud, plus the water-vapor path from the cloud base to the surface. The path factor through the cloud with top at level $k$ is given by

$$\bar{u} = \frac{1}{0.8} (\bar{\xi} + 10.0 \times 1.8) (U_{H_2O}(p_{k+1}) - U_{H_2O}(p_k)),$$ \hspace{1cm} (13)

where $U_{H_2O}$ is the water-vapor column amount,

$$U_{H_2O} = \int_k^\infty \left( \frac{p}{p_o} \right) \rho_{H_2O} dz = \int_0^k \mu_{H_2O} \frac{dp^2}{2g},$$ \hspace{1cm} (14)

where $\mu_{H_2O}$ is the specific humidity, and

$$\bar{\xi} = \frac{1}{\sqrt{\xi^2 + 3.6 \times 10^{-4}}}.$$ \hspace{1cm} (15)

The factor of 10.0 multiplying the diffusivity factor of 1.8 accounts for scattering by liquid droplets. $\bar{\xi}$ is approximately the reciprocal of the cosine of the zenith angle. (N.B. The 0.8 factor is the cloud relative humidity. In CCM1, the cloud relative humidity is 1.0. This
discrepancy is not viewed as important, but should be changed in future versions of the
model.)

Overlapped Fraction of Clouds

For overlapped clouds, the surface flux evaluation is complicated by the multiple
reflections not only from clouds to surface, but between the various cloud layers. The
surface flux for the ultraviolet and visible region in the presence of multiply overlapped
clouds is

\[
F_{UVVM}^S = S_I \left( 0.647 - \xi_{O_3}(p_s) \right) (1.0 - \alpha_{df}^{dr}) \sum_k \frac{(1.0 - \bar{\alpha}(k))}{1.0 - \alpha_{UVV}^{df} \alpha_{clmax}^{df}(k)} f_{ML}(k),
\]

where the "effective" cloud albedo of the multiply overlapped system \( \bar{\alpha} \) is

\[
\bar{\alpha}(k) = \alpha^{dr}_c(k) + \alpha^{dr}_{clmax}(k)(1.0 - \alpha^{dr}_c(k))(1.0 - \alpha^{df}_c(k)) \frac{(1.0 - \Delta U(p_k))}{1.0 - \alpha^{dr}_{clmax}(k) \alpha^{df}_c(k)}.
\]

\( \alpha^{dr}_{clmax}(k) \) is the maximum cloud albedo of all cloud layers to direct radiation for all multiply
overlapped layers, and \( \alpha^{df}_{clmax}(k) \) is the analogous quantity for diffuse radiation. The factor

\( 1.0 - \Delta U(p_k) \)

accounts for absorption from cloud top to the surface by ozone, carbon
dioxide, oxygen, and an enhanced water-vapor path,

\[
\Delta U(p_k) = \xi_{O_3}(p_s) - \xi_{CO_2}(p_k) + \xi_{H_2O}(p_s, \bar{u}) - \xi_{H_2O}(p_k) + \xi_{CO_2}(p_s)
\]

\[
- \xi_{CO_2}(p_k) + \xi_{O_2}(p_s) - \xi_{O_2}(p_k),
\]

where \( \bar{u} \) is the water-vapor path from the cloud top to the surface and accounts for the
enhanced path through the cloud due to scattering [see Equation (13)]. (N.B. This path
length accounts for absorption due to all gases irrespective of the wavelengths in which
they absorb.)

The near-infrared surface–flux contribution for multiply overlapped clouds is

\[
F_{NIRM}^S = S_I \sum_k (0.353 - \xi_{H_2O}(p_k, \bar{u}) - \xi_{CO_2}(p_s) - \xi_{O_2}(p_s))
\]

\[
\frac{(1.0 - \bar{\alpha}(k))(1 - \alpha^{df}_N)}{1.0 - \alpha^{df}_{clmax}(k) \alpha^{df}_N} f_{ML}(k).
\]

Solar Surface Flux

The total ultraviolet and visible surface flux is

\[
F_{UVV}^S = F_{UVV0}^S + F_{UVV1}^S + F_{UVVM}^S,
\]
while that for the near-infrared spectral region is

$$F_{NIR}^S = F_{NIR0}^S + F_{NIR1}^S + F_{NIRM}^S.$$  \(21\)

The total net downward solar flux is

$$F^S = F_{UUV}^S + F_{NIR}^S.$$  \(22\)

**Solar Atmospheric Heating**

Solar heating within the atmosphere is separated into three regions—above clouds, within clouds, and below clouds. The "within cloud" case is further divided into the nonoverlapped and multiply overlapped cases.

Heating above cloud tops occurs through three processes—(1) direct-beam absorption of the downward flux, (2) absorption of radiation directly reflected off the cloud tops, and (3) a contribution from solar radiation that is scattered between clouds and the surface and eventually reaches above the clouds. Absorption of the directly reflected beams is accounted for by water vapor only. Absorption of the multiply reflected radiation is accounted for by ozone only.

The flux difference across a layer bounded by the pressure levels \(p_k, p_{k+1}\) above clouds is given by

$$\Delta F_{abu}(k) = R_{ref}^{dr} [\xi_{H_2O}(p_{k+1}) - \xi_{H_2O}(p_k)] + f_{clear} [F^\uparrow(p_{k+1}) - F^\uparrow(p_k)]$$

$$+ \left( R_{ref}^{dr} + R_{refM}^{df} + R_{ref1}^{df} \right) [\xi_{O_3}(p_{k+1}) - \xi_{O_3}(p_k)],$$  \(23\)

where \(F^\uparrow(p_k)\) is the downward clear-sky flux and \(R_{ref}^{dr}\) is the fraction of incident solar radiation that is reflected off nonoverlapping and overlapping clouds,

$$R_{ref}^{dr} = S_I \left[ \alpha_c^{dr}(p_{kc}) f_{1L}(p_{kc}) + \alpha_{c_{clmax}}^{dr}(p_{kc}) f_{ML}(p_{kc}) \right].$$  \(24\)

\(R_{refM}^{df}\) is the fractional amount of radiation multiply reflected by overlapped clouds which escapes above the highest cloud top,

$$R_{refM}^{df} = S_I f_{ML}(p_{kc}) \frac{1.0 - \alpha_{c_{clmax}}^{df}(p_{kc}) (1.0 - \alpha_{c_{clmax}}^{df}(p_{kc})) \alpha_{UVV}^{df}}{1.0 - \alpha_{UVV}^{df} \alpha_{c_{clmax}}^{df}(p_{kc})}.$$  \(25\)

\(R_{ref1}^{df}\) is the fractional amount of radiation multiply reflected by nonoverlapped clouds

$$R_{ref1}^{df} = S_I f_{1L}(p_{kc}) \frac{1.0 - \alpha_c^{dr}(p_{kc}) (1.0 - \alpha_c^{df}(p_{kc})) \alpha_{UVV}^{df}}{1.0 - \alpha_{UVV}^{df} \alpha_c^{df}(p_{kc})}.$$  \(26\)
The flux difference between layers located within nonoverlapped clouds is

\[
\Delta F_{\text{cld}}(p_{kc}) = F^S(p_{kc}) \left\{ \xi_{H_2O}(p_{kc}, \bar{u}) - \xi_{H_2O}(p_{kc}) + \xi_{CO_2}(p_{kc-1}) - \xi_{CO_2}(p_{kc}) \right. \\
+ \xi_{O_3}(p_{kc}) - \xi_{O_3}(p_{kc-1}) + \xi_{O_3}(p_{kc-1}) - \xi_{O_3}(p_{kc}) \left. \right\} \\
+ F^S(p_{kc}) \alpha^{df}_{\text{NIR}} \frac{(\xi_{H_2O}(\bar{u}) - \xi_{H_2O}(u))(1 - \alpha^{df}_{\text{c}}(p_{kc}))}{1.0 - \alpha^{df}_{\text{c}}(p_{kc}) \alpha^{df}_{\text{NIR}}},
\]

where \(\bar{u}\) is the water-vapor path through the cloud with top at \(p_{kc}\),

\[
\bar{u} = (u(p_{kc-1}) - u(p_{kc})) \left\{ \frac{1}{0.8}(\zeta + 10.0 \times 1.8) \right\},
\]

where the factor in brackets accounts for enhanced path due to scattering. The term \(\bar{u}\) is the water-vapor path through the cloud and down to the surface and back to the cloud base,

\[
\bar{u} = u + 2.0 \times 1.8 \times (u(p_s) - u(p_{kc-1})).
\]

and \(\bar{u}\) is the water-vapor path through the cloud, down to the surface and back to the cloud top at \(p_{kc}\),

\[
\bar{u} = u + 1.8(u(p_{kc-1}) - u(p_{kc})) \frac{10.0}{0.8},
\]

where once again the last term accounts for beam enhancement due to scattering. \(F^S(p_{kc})\) is the absorbed solar flux incident at the top of the clouds,

\[
F^S(p_{kc}) = S_I(1.0 - \alpha^{df}_{\text{c}}(p_{kc})) f_{1L}(p_{kc}).
\]

The heating below the clouds is calculated from the total absorption of solar radiation in the atmosphere from the cloud base to the surface. The total columnar absorption is then distributed within each layer below the clouds by weighting by the mass fraction within each layer.

The absorbed flux for the entire column between the cloud base at \(p_{kc-1}\) and the surface \(p_s\) is given by
\[
\Delta F'_{b\ell w}(p_{kc}) = S_{fML}(p_{kc}) (1 - \bar{a}(p_{kc})) \left\{ \Delta U(p_{kc}) + \frac{(.353 - [\xi_{H_2O}(\overline{\theta}) + \xi_{CO_2}(p_s) + \xi_{O_2}(p_s)])}{1 - \alpha_{NIR}^d \alpha_{clmax}^d (p_{kc})} \times \alpha_{NIR}^d (1 - \alpha_{clmax}^d (p_{kc})) \right\},
\]

where \( \bar{a}(p_{kc}) \) is defined in Equation (17), and \( \Delta U(p_{kc}) \) is defined in Equation (18). This total absorbed flux is then distributed within each layer below the cloud base by mass weighting \( \Delta F'_{b\ell w}(p_{kc}) \) as follows:

\[
\Delta F_{b\ell w}(p_{kc}) = \Delta F'_{b\ell w}(p_{kc}) \frac{p_{k-1} - p_k}{p_s - p_{kc-1}}.
\]

The solar heating is then obtained from

\[
Q_{S}^{abv}(k) = \frac{g}{C_p} \frac{\Delta F_{abv}(k)}{\Delta p_k},
\]

above clouds, where \( g \) is the acceleration due to gravity and \( C_p \) is the specific heat of air. Solar heating within a nonoverlapped cloud is

\[
Q_{S}^{cld}(k_c) = \frac{g}{C_p} \frac{\Delta F_{cld}(k_c)}{\Delta p_k},
\]

while solar heating below clouds is given by

\[
Q_{S}^{blw}(k) = \frac{g}{C_p} \frac{\Delta F_{blw}(k)}{\Delta p_k}.
\]

**Longwave Radiation**

Fluxes are calculated at each model level in both up and down directions. The approach to solving the transfer equations is to employ absorptivities and emissivities. Thus, the clear-sky fluxes at a half-level \( k \) are

\[
F_{clr}^\uparrow(p_k) = B(0)\varepsilon(0, p_k) + \int_{0}^{p_k} \alpha(p', p_k) \frac{dB}{dp'}(p') dp',
\]

and

\[
F_{clr}^\downarrow(p_k) = \sigma_B T_S^4 - \int_{p_k}^{p'} \alpha(p', p_k) \frac{dB}{dp'}(p') dp',
\]
where \( B(p) = \sigma_B T^4(p) \) is just Stefan-Boltzmann's law, and the absorptivity is defined as

\[
\alpha(p, p') = \frac{1}{\frac{dB}{dT}(p')} \int A_{\tilde{\nu}}(p', p) \frac{dB_{\tilde{\nu}}}{dT}(p') d\tilde{\nu},
\]

and the emissivity is

\[
\epsilon(0, p) = \frac{1}{B(0)} \int A_{\tilde{\nu}}(0, p) B_{\tilde{\nu}}(0) d\tilde{\nu},
\]

where \( A_{\tilde{\nu}} \) is the absorptivity due to a given gas, \( B_{\tilde{\nu}}(p') \) is Planck's function, and \( \tilde{\nu} \) is the wavenumber in cm\(^{-1}\). An isothermal layer is assumed to exist at the model top \( p = 0 \). For \( \text{CO}_2 \) and \( \text{O}_3 \), the band-absorbtance technique is used to evaluate \( \alpha \) and \( \epsilon \). This method uses the fact that gas absorption is limited to a finite spectral width. The Planck functions are evaluated at the center of the bands, and integration over \( \tilde{\nu} \) is carried out for \( A_{\tilde{\nu}} \). Thus,

\[
\alpha_{\text{CO}_2}(p, p') = \frac{1}{4\sigma T^3(p')} \frac{dB_{\text{CO}_2}}{dT}(p') A_{\text{CO}_2}(p', p),
\]

\( B_{\text{CO}_2} \) is evaluated for \( \tilde{\nu} = 667 \) cm\(^{-1}\), where \( A_{\text{CO}_2}(p', p) \) is the broad-band absorptance from Kiehl and Briegleb (1987). Similarly,

\[
\epsilon_{\text{CO}_2}(0, p) = \frac{1}{\sigma T^4(0)} B_{\text{CO}_2}(0) A_{\text{CO}_2}(0, p).
\]

For ozone,

\[
\alpha_{\text{O}_3}(p, p') = \frac{1}{4\sigma T^3(p')} \frac{dB_{\text{O}_3}}{dT}(p') A_{\text{O}_3}(p', p),
\]

and

\[
\epsilon_{\text{O}_3}(0, p) = \frac{1}{\sigma T^4(0)} B_{\text{O}_3}(0) A_{\text{O}_3}(0, p),
\]

where \( A_{\text{O}_3} \) is the ozone broad-band absorptance from Ramanathan and Dickinson (1979). Water vapor cannot employ the broad-band absorptance method since \( \text{H}_2\text{O} \) absorption extends throughout the entire longwave region. The method of Ramanathan and Downey (1986) is used for water-vapor absorptivities and emissivities. The overlap treatment between water vapor and other gases is described in Ramanathan and Downey (1986). Thus, the total absorptivity is given by

\[
\alpha(p, p') = \alpha_{\text{CO}_2}(p, p') + \alpha_{\text{O}_3}(p, p') + \alpha_{\text{H}_2\text{O}}(p, p'),
\]

and the total emissivity is

\[
\epsilon(0, p) = \epsilon_{\text{CO}_2}(0, p) + \epsilon_{\text{O}_3}(0, p) + \epsilon_{\text{H}_2\text{O}}(0, p).
\]

Clear-sky fluxes are thus obtained by integrating Equations (37) and (38). Details of the integration of these equations are given below.
Surface Fluxes

The downward longwave clear-sky flux at the surface is, thus,

\[ F^\downarrow_{\text{clr}}(p_s) = B(0)\varepsilon(0, p_s) + \int_{0}^{p_s} \alpha(p', p_s) \frac{dB}{dp'}(p') dp', \]  

(47)

while the upward flux at the surface is just

\[ F^\uparrow(p_s) = \sigma_B T_S^4. \]  

(48)

The downward cloudy-sky flux at the surface is

\[ F^\downarrow(p_s) = F^\downarrow_{\text{clr}}(p_s) f_{\text{clear}} + \sigma T^4(p_{clb2}) \varepsilon(p_{clb2}) A(p_{cl2}) \]

\[ + \sum_{k=3}^{K_{\text{MAX}}} \left\{ \sigma T^4(p_{clb k}) + \int_{p_{clb k}}^{p_s} \alpha(p_s, p') \frac{dB}{dp'}(p') dp' \right\} \varepsilon(p_{cl k}) f_{\text{cld}}(k), \]

(49)

where \( p_{clb k} \) is the pressure level of the cloud base at \( k \) (see Figure 1.1), \( \varepsilon(p_{cl k}) \) is the cloud emissivity obtained from

\[ \varepsilon(p_{cl k}) = \begin{cases} 
\frac{1.0 - \exp(-r_{cl} q_c(k))}{1.0 - \exp(-2.5)} & r_{cl} q_c(k) \leq 2.5, \\
1.0 & r_{cl} q_c(k) > 2.5,
\end{cases} \]  

(50)

where \( r_{cl} = 1000 \) and \( q_c(k) \) is the cloud liquid-water content per time step in \( \text{gm cm}^{-2} \text{s}^{-1} \). This formulation for \( \varepsilon \) is from Ramanathan et al. (1983) and is based on the measurements of Griffith et al. (1980). \( f_{\text{cld}}(k) \) is the probability of a cloud existing in layer \( k \) and clear sky below this layer,

\[ f_{\text{cld}}(k) = A_k \prod_{i=2}^{k-1} (1 - A_i), \]  

(51)

or

\[ f_{\text{cld}}(k) = A_k \frac{\prod_{i=2}^{K_{\text{MAX}}} (1 - A_i)}{\prod_{i=k}^{K_{\text{MAX}}} (1 - A_i)}. \]  

(52)

The net longwave flux at the surface is

\[ F^N(p_s) = F^\uparrow(p_s) + F^\downarrow(p_s). \]  

(53)

Fluxes within the atmosphere are evaluated at each model half-layer for longwave heating-rate calculations on full \( \sigma \)-levels. The upward flux between the surface and the lowest cloud layer is equal to the clear-sky upward flux,

\[ F^\uparrow_{\text{blw}}(p_k) = F^\uparrow_{\text{clr}}(p_k). \]  

(54)
Within the layers that contain clouds, the upward flux is

\[
F_{cld}^\uparrow (p_k) = F_{clr}^\uparrow (p_k) f_{clear}(k) + \sum_{\ell = k_{low}}^k \left\{ \sigma T^4(p_{cltl}) \right. \\
\left. - \int_{p_{cltl}}^{p_k} \alpha(p', p_{cltl}) \frac{dB}{dp'} (p') dp' \right\} \times \epsilon(p_{\ell}) f_{cld}(\ell) \quad p_{k_{low}} \leq p_k \leq p_k+1,
\]

where \( f_{cld}(\ell) \) is the probability that a cloud is in layer \( \ell \) and clear sky exists above layer \( \ell \).

\[
f_{cld}(\ell) = A_{\ell} \prod_{i=\ell+1}^{K_{MAX}} (1 - A_i) = A_{\ell} \frac{\prod_{i=\ell+1}^{K_{MAX}} (1 - A_i)}{\prod_{i=k+1}^{K_{MAX}} (1 - A_i)},
\]

\( k_{low} \) is the lowest level of cloud, \( k_{HI} \) is the highest layer of cloud, and \( p_{cltl} \) is the cloud-top level (see Figure 1.1). The upward flux above the clouds is obtained from a similar expression. The downward flux above the cloudy region is equal to the clear-sky flux,

\[
F_{abu}^\downarrow (p_k) = F_{clr}^\downarrow (p_k).
\]

The longwave atmospheric heating rate is obtained from

\[
Q_{lw} (p_k) = \frac{g}{C_p} \frac{F^\uparrow (p_{k+1}) - F^\downarrow (p_{k+1}) - F^\uparrow (p_k) + F^\downarrow (p_k)}{p_{k+1} - p_k}.
\]

The continuous equations for the infrared calculations require a more sophisticated vertical finite-differencing scheme due to the integral term \( \int \alpha dB \) in Equations (37-38).
The reason for the additional care in evaluating this integral arises from the nonlinear behavior of \( \alpha \) across a given model layer. For example, if the flux at half-level \( p_k \) is required, an integral of the form \( \int_{p_k}^{p_{k+1}} \alpha(p',p_k)dB(p') \) must be evaluated. Now for the nearest layer to level \( p_k \), the following terms will arise:

\[
\int_{p_k}^{p_{k+1}} \alpha(p',p_k)dB(p') = \frac{[\alpha(p_{k+1},p_k) + \alpha(p_k,p_k)]}{2} \left[ B(p_k) - B(p_{k+1}) \right],
\]  

(61)

employing the trapezoidal rule. The problem arises with the second absorptivity \( \alpha(p_k,p_k) \) since this term is zero. It is also known that \( \alpha \) is nearly exponential in form within a layer. Thus, to accurately account for the variation of \( \alpha(p,p') \) across a layer, many more grid points are required than are available in the CCM1. The nearest layer must, therefore, be subdivided and \( \alpha \) must be evaluated across the subdivided layers. The algorithm that is employed in CCM1 is to use a trapezoid method for all layers except the nearest layer. For the nearest layer, a subdivision as illustrated in Figure 1.2 is employed.

![Figure 1.2. Nearest subdivided layer for Equations (62, 63).](image)

For the upward flux, the nearest layer contribution to the integral is evaluated from

\[
\int_{\sigma^k_H}^{\sigma^{k+1}_H} adB(\sigma') = \alpha_{22} \left[ B(\sigma^{k+1}_H) - B(\sigma^k) \right] + \alpha_{21} \left[ B(\sigma^k) - B(\sigma_H^k) \right],
\]  

(62)

while for the downward flux, the integral is evaluated according to

\[
\int_{\sigma^k_{H}}^{\sigma^{k+1}_H} adB(\sigma') = \alpha_{11} \left[ B(\sigma^k) - B(\sigma^k_H) \right] + \alpha_{12} \left[ B(\sigma^k_H) - B(\sigma^k) \right].
\]  

(63)
The $\alpha_{ij}$, $i = 1,2; j = 1,2$, are absorptivities evaluated for the subdivided paths shown in Figure 1.2. The path-length dependence for the absorptivities arises from the dependence on the absorptance $A(p,p')$ [e.g., Eq. (39)]. Temperatures are known at $\sigma$-levels. Temperatures at half-sigma levels are determined through linear interpolation in $\log \sigma$ between $\sigma$-level temperatures. Thus, $B(\sigma_k) = \sigma B T_k^\sigma$ can be evaluated at all required levels. The most involved calculation arises from the evaluation of the fraction of layers shown in Figure 1.2. In general, the absorptance of a layer can require the evaluation of the following path lengths:

$$\xi(p_k,p_{k+1}) = f(\bar{T})\bar{p}\Delta p,$$

and,

$$u(p_k,p_{k+1}) = g(\bar{T})\Delta p,$$

and,

$$\beta(p_k,k_{k+1}) = h(\bar{T})\bar{p},$$

where $f$, $g$, and $h$ are functions of temperature due to band parameters (see Kiehl and Ramanathan, 1983), and $\bar{T}$ is an absorber mass-weighted mean temperature.

These path lengths are, in particular, used extensively in the evaluation of $A_{O_3}$ (Ramanathan and Dickinson, 1979) and $A_{CO_2}$ (Kiehl and Briegleb, 1987). But path lengths dependent on both $p^2$ (i.e., $\xi$) and $p$ (i.e., $u$) are also needed in calculating the water-vapor absorptivity, $\alpha_{H_2O}$ (Ramanathan and Downey, 1986). To account for the subdivided layer, a fractional layer amount must be multiplied by $\xi$ and $u$, e.g.,

$$\bar{\xi}_{11} = \xi(\sigma_H^k,\sigma_H^{k+1}) \times UINPL(1,k),$$

$$\bar{u}_{11} = u(\sigma_H^k,\sigma_H^{k+1}) \times WINPL(1,k),$$

and

$$\bar{\beta}_{11} = \beta(\sigma_H^k,\sigma_H^{k+1}) \times PINPL(1,k),$$

where $UINPL$, $WINPL$, and $PINPL$ are factors to account for the fractional subdivided layer amount. These quantities are derived for the case where the mixing ratio is assumed to be constant within a given layer ($CO_2$ and $H_2O$). For ozone, the mixing ratio is assumed to interpolate linearly in physical thickness; thus, another fractional layer amount $ZINPL$ is required for evaluating $A_{O_3}(p,p')$ across subdivided layers.

Consider the subdivided path for $\alpha_{22}$; the total path length from $\sigma_H^k$ to $\sigma_H^{k+1}$ for the $p^2$ path length will be

$$\xi(\sigma_H^k,\sigma_H^{k+1}) \approx \bar{p}_H \left[ p_H^k - p_H^{k+1} \right],$$

where $\bar{p}_H = \frac{p_H^k + p_H^{k+1}}{2}$, or in terms of $\sigma$-coordinates,

$$\xi(\sigma_H^k,\sigma_H^{k+1}) \approx \bar{\sigma}_H \left[ \sigma_H^k - \sigma_H^{k+1} \right],$$
where $\overline{\sigma}_H \equiv \frac{\sigma^k + \sigma^{k+1}}{2}$. The total layer path length is, therefore, proportional to

$$\xi(\sigma_H^k, \sigma_H^{k+1}) \approx \frac{1}{2} (\sigma_H^k - \sigma_H^{k+1^2}).$$

(72)

Now the path length $\xi$ for $\alpha_{22}$ requires the mean pressure,

$$\overline{p}_{22} \approx \frac{1}{2} \left\{ \frac{\sigma^k + \sigma^{k+1}}{2} + \sigma_H^{k+1} \right\},$$

(73)

and the pressure difference

$$\Delta p_{22} \approx \frac{\sigma^k + \sigma^{k+1}}{2} - \sigma_H^{k+1}.$$  

(74)

Therefore, the path $\xi_{22}$ is

$$\xi_{22} \approx \overline{p}_{22} \Delta p_{22} = \frac{1}{2} \left\{ \left( \frac{\sigma^k + \sigma^{k+1}}{2} \right) - \sigma_H^{k+1^2} \right\}. $$

(75)

The fractional path length is obtained by normalizing this by $\xi(\sigma_H^k, \sigma_H^{k+1^1})$.

$$UINPL(2,k) = DAF3(k) \left\{ \left( \frac{\sigma^k + \sigma^{k+1}}{2} \right)^2 - \sigma_H^{k+1^2} \right\},$$

(76)

where

$$DAF3(k) = \frac{1}{\sigma_H^{k^2} - \sigma_H^{k+1^2}}.$$  

(77)

Similar reasoning leads to the following expressions for the remaining fractional path lengths, for $\alpha_{21},$

$$UINPL(3,k) = DAF3(k) \left\{ \left( \frac{\sigma^k + \sigma_k}{2} \right)^2 - \sigma_H^{k+1^2} \right\},$$

(78)

for $\alpha_{11}$

$$UINPL(1,k) = DAF3(k) \left\{ \sigma_H^{k^2} - \left( \frac{\sigma^k + \sigma_k}{2} \right)^2 \right\},$$

(79)

and for $\alpha_{12},$

$$UINPL(4,k) = DAF3(k) \left\{ \sigma_H^{k^2} - \left( \frac{\sigma^k + \sigma^{k+1}}{2} \right)^2 \right\}.$$  

(80)
The UINPL are fractional layer amounts for path length that scale as \( p^2 \), i.e., \( \xi_{ij} \).

For variables that scale linearly in \( p \), e.g., \( \overline{u}_{ij} \), the following fractional layer amounts are used:

\[
WINPL(1,k) = DAF4(k) \left\{ \frac{\sigma_k^H - \sigma_k^H}{2} \right\},
\]
\[
WINPL(2,k) = DAF4(k) \left\{ \frac{\sigma_k^H - \sigma_k^{H+1}}{2} \right\},
\]
\[
WINPL(3,k) = DAF4(k) \left\{ \left( \frac{\sigma_k^H + \sigma_k^H}{2} \right) - \sigma_k^{H+1} \right\},
\]
\[
WINPL(4,k) = DAF4(k) \left\{ \sigma_k^H - \left( \frac{\sigma_k^{H+1} + \sigma_k^H}{2} \right) \right\},
\]

where
\[
DAF4(k) = \frac{1}{\sigma_k^H - \sigma_k^{H+1}}.
\]

These fractional layer amounts are directly analogous to the UINPL, but since \( \overline{u} \) is linear in \( p \), the squared terms are not present.

The variable \( \overline{\beta}_{ij} \) requires a mean pressure for the subdivided layer. These are

\[
PINPL(1,k) = \frac{1}{2} \left\{ \frac{\sigma_k^H + \sigma_k^H}{2} + \sigma_k^H \right\},
\]
\[
PINPL(2,k) = \frac{1}{2} \left\{ \frac{\sigma_k^H + \sigma_k^{H+1}}{2} + \sigma_k^{H+1} \right\},
\]
\[
PINPL(3,k) = \frac{1}{2} \left\{ \frac{\sigma_k^H + \sigma_k^{H+1}}{2} + \sigma_k^{H+1} \right\},
\]
\[
PINPL(4,k) = \frac{1}{2} \left\{ \frac{\sigma_k^H + \sigma_k^{H+1}}{2} + \sigma_k^H \right\}.
\]

Finally, fractional layer amounts for ozone path lengths are needed, since ozone is interpolated linearly in physical thickness. These are given by

\[
ZINPL(1,k) = \frac{1}{2} \ln \left( \frac{\sigma_k^H}{\sigma_k^H} \right),
\]
\[
ZINPL(2,k) = \frac{1}{2} \ln \left( \frac{\sigma_k^H}{\sigma_k^{H+1}} \right).
\]
\[
ZINPL(2,k) = \frac{1}{2} \ln \left( \frac{\sigma_k^{k+1}}{\sigma_{N+1}^N} \right),
\]
\[
ZINPL(3, k) = ZINPL(1, k) + 2ZINPL(2, k),
\]
\[
ZINPL(4, k) = ZINPL(2, k) + 2ZINPL(1, k).
\]

**Boundary Data**

The radiation parameterization requires monthly mean ozone mixing ratios to be specified as a function of the latitude grid, 23 vertical pressure levels, and time. The ozone path lengths are evaluated from the mixing-ratio data. The path lengths are interpolated to the model \(\sigma\)-layer interfaces for use in the radiation calculation. In the standard version, these path lengths are independent of longitude. The model includes an option to make them a function of longitude to account for the vertical displacement of the \(\sigma\)-levels over mountains. As with the sea-surface temperatures, the seasonal version assigns the monthly averages to the mid-month date and updates them every 12 h via linear interpolation. The actual mixing ratios used in the standard version were derived by Chervin (1986) from analyses of Dütsch (1978).

The radiation parameterization also requires that surface albedo be specified on the model grid for land points. The land albedos are constants (independent of time or moisture conditions); land albedos for snow-covered points are 50% weighted values of snow albedos and the constant land albedos. The surface albedo data for the CCM1 are composed of five quantities—the fraction of strong zenith-angle dependence and four surface albedos (two zenith angles and two spectral range groups). The original source of these data is the Matthews (1983) \(1^\circ-\times-1^\circ\) global 32-type vegetation data set, which was reduced to ten vegetation types. Narrow-band (0.2–0.5 \(\mu\)m, 0.5–0.7 \(\mu\)m, 0.7–0.85 \(\mu\)m, and 0.85–4.0 \(\mu\)m) spectral albedos (for diffuse incident radiation) were ascribed to each of these ten types (Briegleb *et al.*, 1986). The ten surface types were segregated into two groups, based on solar zenith-angle dependence (strong or weak) and averaged to the CCM1 spectral intervals (0.2–0.9 \(\mu\)m, 0.9–4.0 \(\mu\)m). The \(1^\circ-\times-1^\circ\)-data set was then averaged to the required horizontal resolution of the CCM1.
Cloud Parameterization

The model forms clouds that interact with the radiation parameterization described in Section 1. The algorithm is essentially that used in CCM0 and described in Ramanathan et al. (1983). Convective clouds are formed when one or more layers undergo moist convective adjustment. The model assumes that the clouds in each layer are randomly overlapped with a maximum cloud cover of 30 percent in the convective column and cloud emissivity of 1.

Nonconvective clouds are formed wherever stable condensation occurs. The fractional cloud cover of these nonconvective clouds is assumed to be 95 percent and their emissivity is a function of liquid water content. No clouds of any type are formed in the very thin surface layer of the model nor in the top two layers of the standard 12-layer version.

Convective clouds form in the columns in which the moist convective adjustment takes place. The variables $K_{CLOW}$ and $K_{CHI}$ specify the lowest level (largest $k$) and one plus the highest level (smallest $k$), respectively, affected by the moist convective adjustment. Convective clouds are thus centered at the temperature levels $K_{CLOW}$ at the bottom to $K_{CHI} - 1$ at the top; the clouds are assumed to extend through the entire layer from $k + 1/2$ to $k - 1/2$ for each $K_{CLOW} \geq k \geq K_{CHI} - 1$. Clouds are not allowed to form in the very thin first layer of the model or in the top two layers so that for the following $K_{CLOW}$ and $K_{CHI}$ are constrained by

$$K_{CLOW} \leq K - 1, \quad K_{CHI} - 1 > 2. \quad (1)$$

The fractional cloud cover $A_k$ in each layer is given by

$$A_k = 0.3/(K_{CLOW} - K_{CHI} + 2), \quad K_{CLOW} \geq k \geq K_{CHI} - 1, \quad (2)$$

where $K_{CLOW} - K_{CHI} + 2$ is the number of layers having convective clouds. The total area of sky covered with these clouds $A_T$, assuming random overlap, is

$$A_T = 1 - \prod_{k=K_{CHI}-1}^{K_{CLOW}} (1 - A_k). \quad (3)$$

Thus, (2) and (3) imply that the total column cloud cover due to convective clouds is 30 percent. Note that clouds are formed at all levels between $K_{CLOW}$ and $K_{CHI} - 1$ even though moist convective adjustment may not have occurred at all these levels. The
convective cloud scheme also forms a cloud layer of 95 percent just above the top convective region and just below the lowest convective layer. These cloud layers have emissivities of 1. If the moist adjustment occurred between the bottom two levels only, the cloud is assumed to form at the second and third levels from the bottom, i.e., if $A_{K-1} \neq 0$ and $A_{K-2} = 0$, $A_{K-2} = A_{K-1}$ after $A_{K-1}$ has been halved.

Nonconvective clouds form in columns where moist convective adjustment did not take place and wherever the stable condensation takes place. Thus, if $q_{ctk}$, which is the total rate of precipitation, is not zero, nonconvective clouds are formed with 95 percent fractional cloud cover, i.e.,

$$\text{if } q_{ctk} > 0, A_k = 0.95, \quad 3 \leq k \leq K - 1. \quad (4)$$

Note again that no clouds form in the first model level $k = K$ or the top two levels.

A further modification is made. If clouds are formed in either the second or third layer, they are assumed to form in both. Thus,

$$\text{if } q_{ctK-1} > 0 \text{ or } q_{ctK-2} > 0,$$

$$A_{K-1} = 0.95,$$

$$A_{K-2} = 0.95. \quad (5)$$

The nonconvective clouds are assigned emissivities which are a function of liquid water content given in (50) in Section 1.

In the description above, the index $k$ corresponds to that of temperature levels (Figure 1.1), and the cloud is assumed to extend throughout the layer from $k + 1/2$ to $k - 1/2$. The radiation and cloud routines use a modified subscripting running from bottom to top and applied at half-sigma levels rather than that employed above to represent the same thing.
SECTION 3

Code Packages for Physical Parameterizations

The radiation and cloud parameterizations are implemented as separate code packages, independent of the rest of the Model in both control logic and memory utilization. The main purpose of this section is to describe the logical organization and data structures for these packages. Only rough outlines of the parameterization algorithms are included. Details of the physical or mathematical processes are given in Sections 1 and 2. Since both of these packages are implemented in basically the same manner, their common characteristics are described in a generic way.

Generic Description

Control Logic

All of the subroutines within a given parameterization package are entered through a single call to a control routine. The control routine is called by the Model routine PHYS, which is imbedded within the latitude loop in gridpoint space. The parameterization code therefore performs all necessary computations for all gridpoints at a single latitude. The computation sequence is then repeated for the next latitude, and so on.

The parameterization control routine is just a series of calls to other routines in the package. The general sequence is outlined below.

1. Parameterization initialization (set constants, etc.); these calls are conditional, and there may be several levels, e.g.:
   a. Beginning of Model run (initial and/or restart)
   b. Beginning of iteration
   c. Beginning of latitude line
2. Get input gridpoint fields from Model buffer
3. Perform computations
4. Accumulate statistics
5. Put output gridpoint fields into Model buffer

External Communication

Two different kinds of information are exchanged between the basic Model code and a given parameterization: 1) scalar resolution parameters, pointers, and control variables, and 2) arrays of gridpoint data. The two types of data are handled in slightly different ways.
The Model resolution parameters are PARAMETER constants defined in the Update Common Deck PARAMS. Much of the parameterization code references these parameters directly, but some code references another set of resolution parameters local to the parameterization. This local set is also maintained as PARAMETER constants defined in an Update Common Deck, but the definitions are in terms of the Model parameters so that they are automatically changed when the Common Deck PARAMS is changed. Other aspects of the parameterization, however, may be sensitive to resolution changes, particularly in the vertical.

All Model pointers and control variables are contained in COMMON blocks, each COMMON block in a separate Update Common Deck. This information is made available to a parameterization subroutine simply by adding the Update CALL directive to that subroutine. The variables contained in the basic Model COMMON blocks are not modified by the parameterization packages.

The gridpoint data in the Model buffers are not used directly in the parameterization computations. The input fields are first copied into a “local” storage area, and all computations are performed using this local storage. The output fields are then copied into the Model buffer. The “local” storage is the COMMON block /WORKSP/, which is shared by both the radiation and cloud parameterizations. This COMMON block is also used for communication between subroutines within a parameterization package; it is discussed in more detail in the following subsection.

Internal Communication

Virtually all internal communication (i.e., between subroutines within a given parameterization) is through COMMON blocks, with only a few variables passed through subroutine argument lists. In general, these “internal” COMMON blocks appear only in the subroutines within the parameterization, and only if they are referenced. Except for /WORKSP/, the names for these COMMON blocks (which are also the Update Common Deck names) have a three-character prefix which identifies the parameterization. The first character is always “C” (for COMMON block). The prefix CRD is used for radiation; there are no internal COMMON blocks in the cloud parameterization, since all computations are performed in a single subroutine.

The storage provided by /WORKSP/ is “local” only in the sense that each parameterization package uses it locally, i.e., the contents of this COMMON block are not maintained from one control routine call to the next. This implies that /WORKSP/ can contain data for only one latitude at a time. By using different definitions for /WORKSP/, each parameterization can use this memory pool in a different way. For example, the cloud parameterization uses a separate array for each gridpoint field, while the radiation code uses a single large array (F) subdivided by pointers. Potential problems caused by different parameterizations defining different lengths for /WORKSP/ are avoided by using Update Common Decks and equivalencing each version of /WORKSP/ to the dummy array WORK as follows:
The radiation computations use the definition of /WORKSP/ contained in the Common Deck CRDWRK, and the cloud computations use the definition in CLDWRK. Since both of these Common Decks call the Common Deck WORKSP and EQUIVALENCE the WORK array to their own version of /WORKSP/, the length of /WORKSP/ is the same for all definitions, provided LENWRK is at least as large as the largest amount of workspace needed by any parameterization. If LENWRK is inadvertently set too small, the Cray loader will use the longer length definition if it is encountered first; otherwise a fatal error is generated.

**Radiation Parameterization**

**Control Logic**

Since many of the computed radiation quantities change slowly with time, it is not necessary to recompute them for every Model iteration. The radiation parameterization is coded so that all quantities are recomputed only at some constant, prescribed frequency (referred to as a full computation). For all other iterations, only some quantities are recomputed (referred to as a partial computation). Since the radiation computations are relatively time-consuming, this procedure results in a significant reduction in total Model running time.

The radiation computations are divided into two parts: 1) solar, or shortwave (SW) radiation, and 2) infrared, or longwave (LW) radiation. Although there are separate input
parameters which specify the SW and LW full computation frequencies independently,
there is still some code which assumes the two frequencies are the same. The current
version of the Model therefore requires that the two input parameters be equal; otherwise
an error exit results.

Figure 3.1 contains the subroutine calling tree for the radiation parameterization. The
calling sequence is from top to bottom, and within a horizontal block, from left to right.
All subroutines in the package begin with the characters RAD. Figure 3.2 is a verbal outline
of the computation sequence, including conditionals. The LW and SW computations are
outlined separately in Figures 3.3 and 3.4, respectively. Note that heating rates above
the surface are computed only for full computation iterations. For partial computation
iterations, only the surface fluxes are computed, and only for points that are not open
ocean. (Since sea-surface temperatures are specified, the surface energy balance is not
applied over open ocean.)

The code provides for two different ways of determining the global distribution of
ozone. Zonally averaged mixing ratios on an arbitrary set of pressure levels are read from
an external dataset, then the mixing ratios are vertically interpolated to the Model’s sigma
levels. By default, this interpolation uses a constant surface pressure (1000 millibars),
resulting in ozone mixing ratios which are zonally uniform on the Model sigma surfaces.
It is also possible, however, to use the actual surface pressure (which is a function of both
longitude and time), resulting in ozone mixing ratios which vary with longitude and time
(even for perpetual date runs). Since switching between the two interpolation schemes
involves changing the dimensions of several arrays, the switch is implemented by using the
Cray Update conditionals IF and ELSE. The standard version of the Model uses zonally
uniform ozone mixing ratios. To activate the option making the mixing ratios a function
of longitude and time, run the code through Update with the following card image at the
beginning of the Update input file:

$DEFINE OZLONG

The Update Compile File can then be input to the compiler in the normal manner.

NOTE: This option is not as well tested as the standard Model’s zonally uniform ozone
distribution, and there are no control runs available for comparison.

Since only the Update source code contains the conditionals, a code listing generated
by the compiler shows only one version of the ozone distribution option, and CAUTION is
necessary when using a compiler listing to modify code containing references to the ozone
mixing ratios. Ozone conditionals are contained in the radiation Common Decks CRDCON
and CRDPLV, and in the following subroutine Decks: RADABS, RADCSW, RADO3E, RADO2Z,
and RADPLV. A sequenced listing of the complete, conditional code can be obtained by
running Update in the “Quick” mode on the 7 Decks mentioned above, and requesting
that a “Source” dataset be produced (S=sdn) with the sequencing option (SQ) turned on.
The Source dataset can then be disposed as desired. A search for the string "OZLONG" will locate the beginning of each Update conditional.

There are two DATA statements in subroutine RADCON (for arrays CALBR and QFIX) which are dependent on vertical resolution. The numbers of DATA values supplied for these arrays are indicated by independent PARAMETER variables which are checked for consistency with the vertical resolution PARAMETER. A mismatch results in an error exit.

**External Communication**

The workspace used by the radiation code is referenced by using pointers into the single array F. The first subscript of F varies with longitude, starting at the Greenwich meridian and progressing to the east, as in the Model buffers. The second subscript specifies a particular vertical level of a particular field. Subscripts pointing to the first (lowest) level of each field are maintained in the internal COMMON block /CRDNDX/. Multilevel fields are ordered bottom to top, so the levels must be inverted when copying to or from the Model buffers.

All input and output fields are described in Figures 3.5 and 3.6, respectively. Some of these fields are worth additional discussion. The F array fields with indices MABSB, MABSB1, MABSB2, and MABSB3 are input and output, but are never used by other parts of the Model, and are not written to the history tapes. These fields are computed only during full LW computations, but are needed for the partial LW computations. Since the F array cannot be used to store fields between calls to the radiation parameterization, the main Model buffers are used. Also, since the Model buffers are saved as part of the restart and regeneration data, a full computation is not required on the first iteration of restart or regeneration runs for purposes of continuing the radiation calculation. See Bath et al. (1987a) for a discussion of other restrictions applied to the restart and regeneration runs for seasonal experiments.

The F array field with index MCLD is input as cloud fraction, and used without modification for the SW computations. Before being used for the LW computations, however, it is multiplied by the cloud emissivity (CLOUDE), which comes directly (via COMMON block /CRDAKL/) from the cloud parameterization. With respect to the Model buffer, clouds are treated as a full level field (i.e., they are defined at layer midpoints). In the F array, however, the clouds for a given layer are assigned to the top interface of that layer. Clouds for the first layer have a vertical subscript of 2 in the F array, and the first level is never referenced. (Clouds cannot form at the surface).

The input fields ALBNIS, ALBNIW, ALBVSS, ALBVSW and FRCTST comprise the time-independent basis for computing surface albedo. These fields are input only once per run from an external dataset which must be consistent with the Model's horizontal resolution and land-ocean boundaries.

The ozone data is input from an external dataset as a series of 12 monthly averages. The array OZMIXN is used to read zonally averaged ozone mixing ratios on up to 26 pressure
levels. The array POZ, which is read from the same dataset, specifies the pressure levels for OZMIXN. The ozone data is interpolated to the appropriate vertical levels, and also in time, using the two bracketing monthly averages. The time interpolation code handles both seasonal cycle and perpetual date runs. The ozone dataset must be consistent with the Model’s latitudinal resolution.

Only three fields computed by the radiation parameterization are used in computations by the rest of the Model: the SW and LW heating rates for each layer (actually only the sum is used), and the net radiative flux into the surface. Only the surface flux (and only for non-ocean gridpoints) is output for partial computations. Most output fields are transferred to the Model buffer solely for the purpose of output to the history tape, and a few of these are modified in the process. The fields with F array subscripts MALBT, MFIRTP, MFRL, MFRS, and MSABTP are accumulated in the Model buffer, and averaged over the history tape write interval. Since these fields are only computed for full computations and are set to zero otherwise, the corresponding history tape fields (ALBT, FIRTP, FRLA, FRSA, and SABTP, respectively) are correctly averaged only if the history tape write interval is an exact multiple of the full computation interval.

Internal Communication

Virtually all of the communication between subroutines in the radiation parameterization is done through COMMON blocks. All of the internal COMMON blocks begin with the three characters “CRD”, and are maintained as Cray Update Common Decks with the same name. A few of these COMMON blocks contain control variables which are initialized in parts of the Model outside of the radiation parameterization, but their primary use is internal.

The internal COMMON blocks can be divided into two groups according to their maintenance requirements. Figure 3.7 lists those COMMON blocks which contain only variables that are reset with each execution of the radiation code. The memory used by these COMMON blocks is “potentially dynamic” because it could be used for other purposes outside of the radiation parameterization. The current Model version does not, however, take advantage of this overlap potential – all of this memory is used only for the radiation computations (most of it only for the LW part). Figure 3.8 lists the remainder of the internal COMMON blocks, all of which must be maintained between calls to the radiation parameterization; the memory used by these blocks is therefore “necessarily static”. Most of this memory is occupied by arrays which are initialized only once, during Model startup. In both figures, the column labelled “General Length” gives the length of the COMMON block in terms of the gridpoint longitude, latitude, and vertical dimensions (NLON, NLAT, and NLEV, respectively). The column labelled “R15,12L Length” gives the length for the standard 12-layer Model using R15 truncation.
Cloud Parameterization

Control Logic

Figure 3.9 includes the subroutine calling tree for the cloud parameterization. The calling sequence is from top to bottom, and within a horizontal block, from left to right. All of the subroutines in this package begin with the letters CLD, and all computations are performed in the subroutine CLDCMP. Figure 3.10 is a verbal outline of the sequence of operations, including conditionals.

Since the clouds are used only in computing radiative fluxes, their computation is conditional, based solely on the needs of the radiation parameterization. Therefore, clouds are not computed over open ocean for partial radiation computation iterations. (See the preceding description of the radiation parameterization.)

The lowest and highest levels at which clouds are allowed to form (KLDL and KLDH, respectively) are defined as PARAMETERS in Update Common Deck PRDRES. These values may need to be changed if the Model's vertical resolution is changed.

External Communication

The workspace used by the cloud parameterization (COMMON block /WORKSP/) is referenced as a separate, named array for each field. The input and output fields are described in Figure 3.11. With respect to the Model buffer, clouds are treated as a full level field (i.e., they are defined at layer midpoints). In the CLOUD array, however, the clouds for a given layer are assigned to the top interface of that layer. Clouds for the first layer have a vertical subscript of 2 in the CLOUD array, and the first level is never referenced. (Clouds cannot form at the surface). The cloud emissivities (array CLOUDE), are linked directly to the radiation parameterization through COMMON block /CRDACL/. CLOUDE is indexed in the same manner as CLOUD.

For coordination with the radiation parameterization, subroutine CLDCMP includes the radiation COMMON block /CRDCTL/.

Internal Communication

Since all cloud computations are done in a single subroutine, internal communication is limited to the field arrays in /WORKSP/, which have already been described.
RADIATION PARAMETERIZATION CALLING TREE

PHYS

RADCTL

RADINI
RADATE
RADOZ1
RADPLV
RADOZ2
RADINP
RADPTH
RADCSW
RADCLW
RADOUT

RADRDA
RADCJD
RADSLR

RADZEN
RADALB
RADPTL
RADEMS
RADCO2
RADO3E
RADABS
Figure 3.2. Radiation Parameterization Code Flow

RADCTL

IF (beginning of Model run) THEN

RADINI
Set resolution parameters and physical constants

RADRDA
Read surface albedo data from unit NUNIT (set to 58 in RADINI)
Set indices locating all fields in the F array

ENDIF

IF (first latitude) THEN
Set full computation flags
IF (full SW computation) THEN

RADSLR
Compute solar parameters from Julian Day

ENDIF

IF (full SW computation OR full LW computation) THEN
Print date and solar parameters to NOUT
ENDIF

IF (first iteration of run OR (Annual cycle run AND (full SW computation OR full LW computation))) THEN

RADOZ1

IF (annual cycle run OR first call) THEN
Get ozone mixing ratios for bracketing months (monthly data are read from unit NOZONE if not already available; each read results in a message printed to NOUT)
Linearly interpolate the monthly ozone mixing ratios to the current Model time
Compute ozone vertical path length integrals on pressure surfaces
Print ozone update message to NOUT

ENDIF

ENDIF:

ENDIF:

IF (OZLONG defined) THEN

RADPLV
Set pressures (for current latitude only) at half-sigma levels, using time and longitude dependent surface pressure

RADOZ2
Interpolate ozone path length integrals to half-sigma levels (current latitude only)
Figure 3.2. continued

(RADCTL)

ELSEIF (OZLONG undefined) THEN

IF (first iteration of run AND first latitude) THEN

RADPLV

Set pressures (same for all latitudes) at half-sigma levels, using a constant surface pressure

ENDIF

IF (first iteration of run OR full LW computation OR full SW computation) THEN

RADOZ2

Interpolate ozone path length integrals to half-sigma levels (current latitude only)

ENDIF

ENDIF

RADINP

RADZEN

Compute effective mean cosine of solar zenith angle and daylight fraction

Print summary of computed solar parameters

IF (full SW computation OR full LW computation) THEN

Compute vertically averaged temperature and print to unit NOUT

Copy radiation input fields from Model buffer to F array

Set lowest level of clouds (in F array) to zero

RADALB

Compute surface albedos

RADPTH

Limit water vapor mixing ratios to specified minima

Compute vertical, pressure-weighted pathlengths for water vapor and carbon dioxide

Compute total clearsky fraction

RADCSW

Compute shortwave (solar) heating rates and fluxes; see separate outline

RADCLW

Compute longwave heating rates and fluxes; see separate outline

RADOUT

Copy output quantities from F array to Model buffer

Accumulate radiation history tape statistics in Model buffer (initialization is done in LINEMS, normalization in WSHIST)
Figure 3.3. Long-Wave Radiation Computations Code Flow

RADCLW
Set vertical loop limit (KT) for full or partial computation
Multiply cloud fractions by cloud emissivity

RADTPL
Change units of carbon dioxide path lengths
Compute some temperature-related quantities
Compute some water vapor path length quantities

IF (full LW computation) THEN
RADEMS
Compute water vapor emissivities
RADCO2
Compute carbon dioxide emissivities
RADO3E
Compute ozone emissivities

RADABS
First call – compute water vapor, carbon dioxide, and ozone absorptivities between non-adjacent levels

RADABS
Second call – compute water vapor, carbon dioxide, and ozone absorptivities between adjacent levels

ENDIF

FOR ALL (longitude points)

IF (full LW computation OR not over ocean) THEN
Locate the lowest and highest levels with clouds
Compute cloud top and cloud bottom emission

IF (full LW computation) THEN
Compute total absorptivities between all levels
Store absorptivities between each level and the lowest level, and also for the lowest layer, in the F array, for use on subsequent partial computation iterations

ENDIF
Compute Planck function contribution for each half of each layer (finite differencing divides layers in half)

IF (full LW computation) THEN
Compute vertical sums using absorptivities just computed

ELSEIF (partial LW computation) THEN
Compute vertical sums using absorptivities saved from last full computation

ENDIF
Compute upward clearsky flux at the surface
Figure 3.3. continued

(RADCLW)

IF (full LW computation) THEN
    Compute upward and downward clearsky fluxes for all levels above the surface
ENDIF

Compute downward clearsky flux at surface

IF (clear sky) THEN
    Set final upward and downward surface fluxes to the clearsky values
    IF (full LW computation) THEN
        Set final upward and downward fluxes at all levels above the surface to the clearsky values
    ENDIF
ELSEIF (cloudy sky) THEN
    Set final upward surface flux to the clearsky value
    Compute final downward surface flux by using cloud fractions to modify the clearsky value
    IF (full LW computation) THEN
        Compute final upward and downward fluxes at all levels above the surface by using cloud fractions to modify the clearsky values
    ENDIF
ENDIF

Store the sum of the SW and LW downward surface fluxes in the F array

IF (full LW computation) THEN
    Store history tape statistics in the F array
    Compute LW heating rate at each level and store in the F array
ENDIF

ENDIF

END FOR ALL
Figure 3.4. Short-Wave Radiation Computations Code Flow

RADCSW

Set vertical loop limit (KT) for full or partial computation
Initialize output fluxes and heating rates to zero

FOR ALL (longitude points)

IF (daytime AND (full SW computation OR not over ocean)) THEN
  Locate the lowest and highest levels with clouds
  Compute path lengths and absorptivities for water vapor, carbon dioxide, ozone, and molecular oxygen
  Compute net diffuse (reflected upward) clearsky flux at each level
  Compute net direct (downward) clearsky flux at each level
  Compute combined ground and foliage albedo
  Set output fluxes and heating rates to clearsky values

IF (there are clouds) THEN
  Compute cloud albedos for both diffuse and direct radiation
  Compute cloud and surface absorption for overlapped clouds
  Compute cloud and surface absorption for non-overlapped clouds
  Modify output fluxes to account for clouds

IF (full SW computation) THEN
  Compute heating rate above the cloud top
  Compute heating rate within and below the clouds
  Modify output heating rates to account for clouds
ENDIF

ENDIF

ENDIF

END FOR ALL

Convert solar heating from flux units to degrees per second
### Figure 3.5. Radiation Parameterisation Input Fields

<table>
<thead>
<tr>
<th>Internal Array Name and Dimensions</th>
<th>Internal Field Description</th>
<th>Physical Dimensions</th>
<th>COMMON Block</th>
<th>Set In</th>
<th>Used In</th>
<th>External Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F(NLON, ABSB+k-1); k=1, NLEV+1$</td>
<td>Total LW absorptivity ($H_2O, CO_2, O_3$) between the lower interface of each layer and the surface, from the last full LW computation; input only for partial LW computations</td>
<td>fraction</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADCLW</td>
<td>BUF(NABSB)</td>
</tr>
<tr>
<td>$F(NLON, MABSB1); 1$ level</td>
<td>Total LW absorptivity ($H_2O, CO_2, O_3$) for the lowest 1/4 of the lowest layer, from the last full LW computation; input only for partial LW computations</td>
<td>fraction</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADCLW</td>
<td>BUF(NABSB1)</td>
</tr>
<tr>
<td>$F(NLON, MABSB2); 1$ level</td>
<td>Total LW absorptivity ($H_2O, CO_2, O_3$) for the lowest 3/4 of the lowest layer, from the last full LW computation; input only for partial LW computations</td>
<td>fraction</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADCLW</td>
<td>BUF(NABSB2)</td>
</tr>
<tr>
<td>$F(NLON, MABSBT); 1$ level</td>
<td>Boundary term used to compute the downward LW flux at the surface, from the last full LW computation; input only for partial LW computations</td>
<td>erg/cm²/sec</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADCLW</td>
<td>BUF(NABSBT)</td>
</tr>
<tr>
<td>$F(NLON, NCLD+k-1); k=1, NLEV+1$</td>
<td>Cloud fraction at layer interfaces (half-levels), starting at the surface; multiplied by cloud emissivity in RADCLW</td>
<td>fraction</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADCLW</td>
<td>BUF(NCLD)</td>
</tr>
<tr>
<td>$F(NLON, NOCEAN); 1$ level</td>
<td>Ocean flag set to 1. for open ocean grid-points, 0. otherwise</td>
<td>flag</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADALB</td>
<td>BUF(NORO)</td>
</tr>
<tr>
<td>$F(NLON, NPBR+k-1); k=1, NLEV$</td>
<td>Pressure at layer midpoints (full levels); not currently used</td>
<td>dyne/cm²</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>—</td>
<td>BUF(NLPSM1)</td>
</tr>
<tr>
<td>$F(NLON, NPNN+k-1); k=1, NLEV+1$</td>
<td>Pressure at layer interfaces (half-levels)</td>
<td>dyne/cm²</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADABS</td>
<td>BUF(NLPSN1)</td>
</tr>
<tr>
<td>$F(NLON, NQNM+k-1); k=1, NLEV$</td>
<td>Water vapor mixing ratio (Model specific humidity is copied without modification)</td>
<td>Kg/Kg</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADABR</td>
<td>BUF(NQNM1)</td>
</tr>
</tbody>
</table>
Figure S.5. continued

<table>
<thead>
<tr>
<th>Internal Array Name and Dimensions</th>
<th>Internal Field Description</th>
<th>Physical Dimensions</th>
<th>COMMON Block</th>
<th>Set In</th>
<th>Used In</th>
<th>External Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>F(NLON, MSCV); 1 level</td>
<td>Snowcover flag set to 1. for gridpoints with snowcover, -1. otherwise</td>
<td>flag</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADALB</td>
<td>BUF(NORD)</td>
</tr>
<tr>
<td>F(NLON, MSICE); 1 level</td>
<td>Sea ice flag set to 1. for sea ice gridpoints, 0. otherwise</td>
<td>flag</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADALB</td>
<td>BUF(NORD)</td>
</tr>
<tr>
<td>F(NLON, MTNM+k-1); k=1, NLEV</td>
<td>Temperature at layer midpoints (full levels)</td>
<td>°K</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADCSW</td>
<td>BUF(NTM1)</td>
</tr>
<tr>
<td>F(NLON, MTG); 1 level</td>
<td>Surface (ground) temperature</td>
<td>°K</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADCLW</td>
<td>BUF(NTS)</td>
</tr>
<tr>
<td>F(NLON, MTS); 1 level</td>
<td>Lowest free atmosphere temperature; MTS is set to MTNM in RADINI; anemometer level should be used if available</td>
<td>°K</td>
<td>/WORKSP/</td>
<td>RADINP</td>
<td>RADALB</td>
<td>BUF(NTM1)</td>
</tr>
<tr>
<td>F(NLON, MVEG); 1 level</td>
<td>Fractional vegetation cover used for ground albedo computation; currently set to zero everywhere</td>
<td>fraction</td>
<td>/WORKSP/</td>
<td>RADALB</td>
<td>RADCSW</td>
<td>—</td>
</tr>
<tr>
<td>ALBNIS(NLON, NLAT)</td>
<td>Near infrared albedo for strong zenith angle dependence; converted from percent</td>
<td>fraction</td>
<td>/CRDALB/</td>
<td>RADRDA</td>
<td>RADALB</td>
<td>NUNIT-58</td>
</tr>
<tr>
<td>ALBNIV(NLON, NLAT)</td>
<td>Near infrared albedo for weak zenith angle dependence; converted from percent</td>
<td>fraction</td>
<td>/CRDALB/</td>
<td>RADRDA</td>
<td>RADALB</td>
<td>NUNIT-58</td>
</tr>
<tr>
<td>ALEVSS(NLON, NLAT)</td>
<td>Visible albedo for strong zenith angle dependence; converted from percent on input</td>
<td>fraction</td>
<td>/CRDALB/</td>
<td>RADRDA</td>
<td>RADALB</td>
<td>NUNIT-58</td>
</tr>
<tr>
<td>ALEVSW(NLON, NLAT)</td>
<td>Visible albedo for weak zenith angle dependence; converted from percent on input</td>
<td>fraction</td>
<td>/CRDALB/</td>
<td>RADRDA</td>
<td>RADALB</td>
<td>NUNIT-58</td>
</tr>
<tr>
<td>FRCTST(NLON, NLAT)</td>
<td>Fraction of grid box with strong zenith angle dependence for albedo</td>
<td>fraction</td>
<td>/CRDALB/</td>
<td>RADRDA</td>
<td>RADALB</td>
<td>NUNIT-58</td>
</tr>
<tr>
<td>CLOUDE(NLON, NLEV+1)</td>
<td>Cloud emissivity; set directly by cloud parameterisation</td>
<td>fraction</td>
<td>/CRDACL/</td>
<td>CLDCNP</td>
<td>RADCLW</td>
<td>—</td>
</tr>
<tr>
<td>OZMIXN(NLAT, 26)</td>
<td>Zonally averaged ozone mixing ratios at up to 26 pressure levels; appropriate monthly average is read and used for time interpolation</td>
<td>Kg/Kg</td>
<td>/CONOZP/</td>
<td>RADOZ1</td>
<td>RADOZ1</td>
<td>NOZONE=1</td>
</tr>
<tr>
<td>POZ(26)</td>
<td>Pressures for OZMIXN levels</td>
<td>Pascal</td>
<td>/CONOZP/</td>
<td>RADOZ1</td>
<td>RADOZ1</td>
<td>NOZONE=1</td>
</tr>
<tr>
<td>Internal Array Name and Dimensions</td>
<td>Internal Field Description</td>
<td>Physical Dimensions</td>
<td>Set In</td>
<td>External Destination</td>
<td>External Use</td>
<td>History Tape Name</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>---------------------------</td>
<td>---------------------</td>
<td>--------</td>
<td>----------------------</td>
<td>--------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>( F(NLON, MABSB+1) ); ( k=1 ), ( NLEV+1 )</td>
<td>Total LW absorptivity (( H_2O, CO_2, O_3 )) between the lower interface of each layer and the surface; computed and output only during a full LW computation</td>
<td>fraction</td>
<td>RADCLW</td>
<td>BUF(MABSB)</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>( F(NLON, MABSB1) ); 1 level</td>
<td>Total LW absorptivity (( H_2O, CO_2, O_3 )) for the lowest 1/4 of the lowest layer; computed and output only during a full LW computation</td>
<td>fraction</td>
<td>RADCLW</td>
<td>BUF(MABSB1)</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>( F(NLON, MABSB2) ); 1 level</td>
<td>Total downward LW absorptivity (( H_2O, CO_2, O_3 )) for the lowest 3/4 of the lowest layer; computed and output only during a full LW computation</td>
<td>fraction</td>
<td>RADCLW</td>
<td>BUF(MABSB2)</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>( F(NLON, MABSBT) ); 1 level</td>
<td>Boundary term to compute the downward LW flux at the surface; computed and output only during a full LW computation</td>
<td>erg/cm^2/sec</td>
<td>RADCLV</td>
<td>BUF(MABSBT)</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>( F(NLON, MALBT) ); 1 level</td>
<td>Solar flux absorbed by atmosphere above top of the Model; in BUF, accumulated and averaged over history tape write interval, and replaced by average planetary albedo (fraction); written to history tape</td>
<td>erg/cm^2/sec</td>
<td>RADCSW</td>
<td>BUF(NALBT)</td>
<td>—</td>
<td>ALBT</td>
</tr>
<tr>
<td>( F(NLON, NCLRLS) ); 1 level</td>
<td>Net upward clearsky LW flux at the surface</td>
<td>erg/cm^2/sec</td>
<td>RADCLW</td>
<td>BUF(NCLRLS)</td>
<td>—</td>
<td>CLRRLS</td>
</tr>
<tr>
<td>( F(NLON, NCLRLT) ); 1 level</td>
<td>Upward clearsky LW flux at the top of the atmosphere</td>
<td>erg/cm^2/sec</td>
<td>RADCLW</td>
<td>BUF(NCLRLT)</td>
<td>—</td>
<td>CLRLT</td>
</tr>
<tr>
<td>( F(NLON, NCLRSS) ); 1 level</td>
<td>Net downward clearsky SW flux at the surface</td>
<td>erg/cm^2/sec</td>
<td>RADCSW</td>
<td>BUF(NCLRSS)</td>
<td>—</td>
<td>CLRSS</td>
</tr>
</tbody>
</table>

(NOTE: The following 3 fields are accumulated in BUF, then averaged over the history tape write interval and converted to W/m^2 before writing to the history tape.)
(NOTE: The following 4 fields are accumulated in BUF, then averaged over the history tape write interval and converted to W/m² before writing to the history tape.)

<table>
<thead>
<tr>
<th>Internal Array Name and Dimensions</th>
<th>Internal Field Description</th>
<th>Physical Dimensions</th>
<th>Set In</th>
<th>External Destination</th>
<th>External Use</th>
<th>History Tape Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>(F(\text{NLON}, N\text{CLRST})); 1 level</td>
<td>Net downward clearsky SW flux at the top of the atmosphere</td>
<td>erg/cm²/sec</td>
<td>RADCSW</td>
<td>BUF (NCLRST)</td>
<td>—</td>
<td>CLRST</td>
</tr>
<tr>
<td>(F(\text{NLON}, N\text{FIRTP})); 1 level</td>
<td>Net upward LW flux at the top of the Model</td>
<td>erg/cm²/sec</td>
<td>RADCLW</td>
<td>BUF (NFIRTP)</td>
<td>—</td>
<td>FIRTP</td>
</tr>
<tr>
<td>(F(\text{NLON}, N\text{FRLA})); 1 level</td>
<td>Net upward LW flux at the surface</td>
<td>erg/cm²/sec</td>
<td>RADCLW</td>
<td>BUF (NFRLA)</td>
<td>—</td>
<td>FRLA</td>
</tr>
<tr>
<td>(F(\text{NLON}, N\text{FRSA})); 1 level</td>
<td>Net LW heating rate for each layer; written to history tape</td>
<td>°K/sec</td>
<td>RADCLW</td>
<td>BUF (NFRSA)</td>
<td>DTRADS</td>
<td>QRL</td>
</tr>
<tr>
<td>(F(\text{NLON}, N\text{QRS}+k-1); k=1, \text{NLEV} )</td>
<td>Net LW heating rate for each layer; converted (within the (F) array) from a net flux into each layer (erg/cm²/sec) at the end of (RADCSV); written to history tape</td>
<td>°K/sec</td>
<td>RADCSW</td>
<td>BUF (NQRS)</td>
<td>DTRADS</td>
<td>QRS</td>
</tr>
<tr>
<td>(F(\text{NLON}, N\text{SA}BTP)); 1 level</td>
<td>Solar flux absorbed by all Model layers plus the surface; in BUF, accumulated and averaged over history tape write interval, and converted to W/m²; written to history tape</td>
<td>erg/cm²/sec</td>
<td>RADCSW</td>
<td>BUF (NSABTP)</td>
<td>—</td>
<td>SABTP</td>
</tr>
<tr>
<td>(F(\text{NLON}, N\text{SLWD})); 1 level</td>
<td>Total (LW + SW) net downward radiative flux at the surface; written to history tape</td>
<td>J/m²/sec</td>
<td>RADCLW</td>
<td>BUF (NDRP1)</td>
<td>TSCALC</td>
<td>SRAD</td>
</tr>
<tr>
<td>(F(\text{NLON}, N\text{SOLIN})); 1 level</td>
<td>Downward SW flux (solar input) at the top of the atmosphere; in BUF, accumulated and averaged over history tape write interval, and converted to W/m²; written to history tape</td>
<td>erg/cm²/sec</td>
<td>RADCSW</td>
<td>BUF (NSOLIN)</td>
<td>—</td>
<td>SOLIN</td>
</tr>
</tbody>
</table>
## Figure 3.7. Potentially Dynamic Radiation COMMON Blocks

<table>
<thead>
<tr>
<th>Block Name</th>
<th>General Length</th>
<th>R15,12L Length</th>
<th>No. of Arrays</th>
</tr>
</thead>
<tbody>
<tr>
<td>/CRDABS/</td>
<td>NLON<em>NLEV²+15</em>NLON<em>NLEV+3</em>NLON</td>
<td>15,696</td>
<td>6</td>
</tr>
<tr>
<td>/CRDC02/</td>
<td>3<em>NLON</em>NLEV+2*NLON</td>
<td>1,824</td>
<td>3</td>
</tr>
<tr>
<td>/CRDEMS/</td>
<td>4<em>NLON</em>NLEV+7*NLON</td>
<td>2,640</td>
<td>7</td>
</tr>
<tr>
<td>/CRDFLX/</td>
<td>5*(NLEV+1)</td>
<td>65</td>
<td>5</td>
</tr>
<tr>
<td>/CRDLAT/</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>/CRDPLV/</td>
<td>2*NLEV+1</td>
<td>25</td>
<td>2</td>
</tr>
<tr>
<td>/CRDPHT/</td>
<td>3<em>NLON</em>(NLEV+1)</td>
<td>1,872</td>
<td>3</td>
</tr>
<tr>
<td>/CRDRDT/</td>
<td>NLON*(NLEV+56)</td>
<td>3,264</td>
<td>29</td>
</tr>
<tr>
<td>/CRDSAV/</td>
<td>NLON</td>
<td>48</td>
<td>1</td>
</tr>
<tr>
<td>/CRDS03/</td>
<td>4<em>NLON</em>(NLEV+1)</td>
<td>2,496</td>
<td>5</td>
</tr>
<tr>
<td>/CRDTMP/</td>
<td>5<em>NLON</em>(NLEV+1)</td>
<td>3,120</td>
<td>5</td>
</tr>
<tr>
<td>/CRDTRN/</td>
<td>4</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>NLON*(NLEV²+35<em>NLEV+81)+7</em>NLEV+11</td>
<td>31,055</td>
<td>66</td>
</tr>
</tbody>
</table>

## Figure 3.8. Necessarily Static Radiation COMMON Blocks

<table>
<thead>
<tr>
<th>Block Name</th>
<th>General Length</th>
<th>R15,12L Length</th>
<th>No. of Arrays</th>
</tr>
</thead>
<tbody>
<tr>
<td>/CRDACL/</td>
<td>NLON*(NLEV+1)</td>
<td>624</td>
<td>1</td>
</tr>
<tr>
<td>/CRDALB/</td>
<td>5<em>NLON</em>NLAT</td>
<td>9,600</td>
<td>5</td>
</tr>
<tr>
<td>/CRDBND/</td>
<td>14</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>/CRDCAE/</td>
<td>16*NLEV+160</td>
<td>352</td>
<td>13</td>
</tr>
<tr>
<td>/CRDCON/</td>
<td>2<em>NLAT</em>NLEV+10*NLON</td>
<td>1,997</td>
<td>25</td>
</tr>
<tr>
<td>/CRDCTL/</td>
<td>4</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>/CRDDYR/</td>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>/CRDINT/</td>
<td>3<em>NLEV²+25</em>NLEV+22</td>
<td>754</td>
<td>10</td>
</tr>
<tr>
<td>/CRDNDX/</td>
<td>28</td>
<td>28</td>
<td>0</td>
</tr>
<tr>
<td>/CRDSLZ/</td>
<td>5</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>/CRDSNL/</td>
<td>2<em>NLON+4</em>NLAT+12</td>
<td>268</td>
<td>7</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>5<em>NLON</em>NLAT+2<em>NLAT+13</em>NLON+10<em>NLAT+3</em>NLEV²+50*NLEV+457</td>
<td>13,649</td>
<td>68</td>
</tr>
</tbody>
</table>

3-18
Figure 3.10. Cloud Parameterization Code Flow

CLDCTL

CLDINP
Copy cloud input fields from Model buffer to work arrays

CLDCMP
Initialize cloud emmissivities to 1.

FOR ALL (longitudes)

IF (full SW computation OR full LW computation OR not over ocean) THEN
Initialize cloud amounts to zero at all levels

IF (no convective instability in any layer) THEN

IF (condensation in layer 2 OR condensation in layer 3) THEN
Set (stratiform) cloud amount to 0.95 at the top of layers 2 and 3

FOR ALL (layers from 4 through 9, inclusive)

IF (condensation in the current layer) THEN
Set (stratiform) cloud amount to 0.95 at the top of the layer
Compute corresponding cloud emissivity
ENDIF

ENDIF

ELSEIF (at least one layer was convectively adjusted) THEN
Set (convective) cloud thickness to number of layers between lowest and highest convectively adjusted layers, inclusive
If only the lowest layer was convectively adjusted, increase cloud thickness to 2 layers

FOR ALL (layers 3 through 10, inclusive)

If the layer two layers below the current layer is between the lowest and highest convectively adjusted layers (inclusive), then set the (convective) cloud amount at the top of the current layer to 0.3 divided by the cloud thickness

IF (condensation in the current layer) THEN
If the current layer is more than 2 layers above the highest convectively adjusted layer, set the (convective) cloud amount at the top of the current layer to 0.95
ENDIF

ENDIF

1 For the cloud computations, Model layers are numbered from bottom to top. The vertical index in the CLOUD array locates layer interfaces (half-levels), starting at the surface and increasing upward. Cloud amounts for layer n are assigned to the level at the top interface of the layer, which corresponds to a CLOUD subscript of n+1.

2 This does not necessarily imply that the layer was convectively adjusted, since there may be several noncontiguous groups of adjusted layers.
Figure 3.10. continued

(CLDCMP)

If the current layer is at or below the lowest convectively adjusted layer, set the (convective) cloud amount at the top of the current layer to 0.95

ENDIF

END FOR ALL

If the (convective) cloud amount at the top of layer 3 is less than the cloud amount at the top of layer 2, reset the layer 3 cloud amount to the layer 2 value

ENDIF

ENDIF

END FOR ALL

CLDOUT

Copy cloud amounts and emissivities from work arrays CLOUD and CLOUDE to Model buffer
### Figure 3.11. Cloud Parameterisation Input/Output Fields

#### Input Fields

<table>
<thead>
<tr>
<th>Internal Array Name and Dimensions</th>
<th>Internal Field Description</th>
<th>Physical Dimensions</th>
<th>COMMON Block</th>
<th>Set In</th>
<th>Used In</th>
<th>External Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLOUD(NLON, NLEV+1)</td>
<td>Cloud fraction at layer interfaces (half-levels) for the previous iteration; first level is the surface; not currently used</td>
<td>fraction /WORKSP/ CLDINP</td>
<td>—</td>
<td>BUF(NCLD)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KCOL(NLON)</td>
<td>Lowest convectively adjusted layer; set to zero if no layers adjusted</td>
<td>layer index /WORKSP/ CLDINP CLDCMP</td>
<td>BUF(NCNB)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KCHI(NLON)</td>
<td>Highest convectively adjusted layer; set to zero if no layers adjusted</td>
<td>layer index /WORKSP/ CLDINP CLDCMP</td>
<td>BUF(NCNT)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QC(NLON, NLEV)</td>
<td>Total (large-scale + convective) condensation rate</td>
<td>g/cm²/timestep /WORKSP/ CLDINP CLDCMP</td>
<td>BUF(NQC)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OCEAN(NLON)</td>
<td>Logical flag set to .TRUE. iff the grid point is open ocean</td>
<td>logical /WORKSP/ CLDINP CLDCMP</td>
<td>BUF(NORO)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Output Fields

<table>
<thead>
<tr>
<th>Internal Array Name and Dimensions</th>
<th>Internal Field Description</th>
<th>Physical Dimensions</th>
<th>COMMON Block</th>
<th>Set In</th>
<th>External Destination</th>
<th>External Use</th>
<th>History Tape Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLOUD(NLON, NLEV+1)</td>
<td>Cloud fraction at layer interfaces (half-levels), starting at the surface; surface level not carried in BUF; written to history tape</td>
<td>fraction /WORKSP/</td>
<td>CLDCMP BUF(NCLD) RADINP</td>
<td>CLOUD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLOUDE(NLON, NLEV+1)</td>
<td>Cloud emissivities; used directly by radiation parameterisation</td>
<td>fraction /CRDACL/</td>
<td>CLDCMP BUF(NCLE) RADCLW</td>
<td>CLOUDE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SECTION 4

This section catalogues and describes the various subroutines used by the model to calculate radiative fluxes and heating rates, as well as the cloud parameters used to calculate these radiation quantities. The subroutine names are given along with their arguments. A short description of the purpose of each routine is given. Definitions of the subroutine arguments are also provided. Finally, the basis of the algorithm employed by the subroutine is given. The algorithms are described in more detail in Sections 1 and 2 of this document.
RADABS

SUBROUTINE RADABS(LATIT,ICALL)

Update deck location: RADABS.3 - RADABS.552
Concordance identifier: RDABS

PURPOSE

Computes LW absorptivities.

ARGUMENTS

LATIT : [input] north to south index for current latitude (integer)
ICALL : [input] computation control flag (integer);
        = 1 - compute absorptivities between non-adjacent levels,
        = 2 - compute absorptivities between adjacent levels.

ALGORITHM

This subroutine must be called twice in order to compute all required absorptivities: first with ICALL = 1, then with ICALL = 2. For a description of the absorptivity calculations, refer to the NCAR Tech Note "Description of NCAR Community Climate Model (CCM1)," (Williamson, et al., 1987).
SUBROUTINE RADALB

Update deck location: RADALB.3 - RADALB.219
Concordance identifier: RDALB

PURPOSE
Computes ground, foliage, ocean, sea ice albedo as a function of snow cover, zenith angle, and wavelength.

ALGORITHM
The data which has been input from the external albedo dataset (and is contained in the COMMON block /CRDALB/), consists of the following fields:

\[
\begin{align*}
\alpha_{vss}(i,j) & \quad \text{visible albedo for strong zenith angle dependence} \\
\alpha_{vsw}(i,j) & \quad \text{visible albedo for weak zenith angle dependence} \\
\alpha_{nis}(i,j) & \quad \text{near infrared albedo for strong zenith angle dependence} \\
\alpha_{niw}(i,j) & \quad \text{near infrared albedo for weak zenith angle dependence} \\
f_s & \quad \text{fraction of grid box with strong zenith angle dependence}
\end{align*}
\]

All values are dimensionless (simple fractions), except for the following special codes: 1) albedos are set to $10^{-36}$ for ocean and ice surfaces, and 2) $f_s$ is set to $10^{-36}$ for ocean and $-1000$ for "permanent" land and sea ice points. Based on these time-independent fields, the following time-dependent fields are to be computed:

\[
\begin{align*}
\alpha_{v}(v, \text{dir}) & \quad \text{ALBS}(i) \quad \text{vegetation albedo for direct, visible solar radiation} \\
\alpha_{n}(v, \text{dir}) & \quad \text{ALBL}(i) \quad \text{vegetation albedo for direct, near-IR solar radiation} \\
\alpha_{v}(v, \text{dif}) & \quad \text{ALBSD}(i) \quad \text{vegetation albedo for diffuse, visible solar radiation} \\
\alpha_{n}(v, \text{dif}) & \quad \text{ALBLD}(i) \quad \text{vegetation albedo for diffuse, near-IR solar radiation} \\
\alpha_{v}(g, \text{dir}) & \quad \text{ALBGS}(i) \quad \text{ground albedo for direct, visible solar radiation} \\
\alpha_{n}(g, \text{dir}) & \quad \text{ALBGL}(i) \quad \text{ground albedo for direct, near-IR solar radiation} \\
\alpha_{v}(g, \text{dif}) & \quad \text{ALBGSD}(i) \quad \text{ground albedo for diffuse, visible solar radiation} \\
\alpha_{n}(g, \text{dif}) & \quad \text{ALBGLD}(i) \quad \text{ground albedo for diffuse, near-IR solar radiation}
\end{align*}
\]

The direct solar albedo for all wavelengths combined (array ALBG(i)) is also computed as $\alpha_v(v, \text{dir}) C(124) + \alpha_n(v, \text{dir})(1 - C(124))$, but it is not used by any part of the Model code.

Begin execution:
Set fractional vegetation cover to zero everywhere ($F(i, \text{MVEG}) = 0$) and initialize all output albedos to zero.

* LAND SURFACES NOT COVERED WITH ICE:
Compute albedos for land surfaces without ice. (Defining condition is $F(i, \text{MOCEAN}) \neq 1$ AND $F(i, \text{MSICE}) = 0$ AND $f_s > -10$. If there is snow
cover, the values are modified later in this subroutine.

\[
\begin{align*}
\alpha_{uv}(v, \text{dir}) &= \alpha_{uv}(g, \text{dir}) = R_s \alpha_{uv} f_s + R_w \alpha_{uvw}(1 - f_s) \\
\alpha_{ni}(v, \text{dir}) &= \alpha_{ni}(g, \text{dir}) = R_s \alpha_{nifs} f_s + R_w \alpha_{niw}(1 - f_s) \\
\alpha_{uv}(v, \text{dif}) &= \alpha_{uv}(g, \text{dif}) = R_s \alpha_{uv} f_s + \alpha_{uvw}(1 - f_s) \\
\alpha_{ni}(v, \text{dif}) &= \alpha_{ni}(g, \text{dif}) = \alpha_{nifs} f_s + \alpha_{niw}(1 - f_s)
\end{align*}
\]

where

\[
R_s = \frac{1.4}{1 + .8\mu}, \quad R_w = \frac{1.1}{1 + .2\mu}
\]

and \(\mu = \cos ZRS(i)\) is the cosine of the solar zenith angle.

* LAND (OR SEA) SURFACES COVERED WITH ICE

Compute albedos over land surfaces with "permanent" ice. (Defining condition is \(F(i,\text{MOCEAN}) \neq 1\) AND \(f_s \leq -10\).) If there is snow cover, the values are modified later in this subroutine.

\[
\begin{align*}
\alpha_{uv}(v, \text{dir}) &= \begin{cases} 
0.70 & T_s < 272.15 \\
0.70 - .04(T_s - 272.15) & 272.15 \leq T_s \leq 277.15 \\
0.50 & T_s > 277.15
\end{cases} \\
\alpha_{uv}(g, \text{dir}) &= \begin{cases} 
0.50 & T_s < 272.15 \\
0.50 - .06(T_s - 272.15) & 272.15 \leq T_s \leq 277.15 \\
0.20 & T_s > 277.15
\end{cases} \\
\alpha_{ni}(v, \text{dir}) &= \begin{cases} 
0.50 & T_s < 272.15 \\
0.50 - .06(T_s - 272.15) & 272.15 \leq T_s \leq 277.15 \\
0.20 & T_s > 277.15
\end{cases}
\]

where \(T_s = F(i,\text{MTS})\) is the air temperature at the surface.

* COMPUTE ALBEDO OVER SEA ICE

Compute albedos for sea ice in the same way as for "permanent" land ice as described above. (Defining condition is \(F(i,\text{MSICE}) > 0\).) These computed values are not modified by snowcover.

* ASSUME SNOW HAS 50% COVERAGE FOR ALL SURFACES EXCEPT

* SEA-ICE, FOR WHICH COVERAGE IS 0%.

Compute albedos for gridpoints with snowcover by combining a snowcover albedo with the previously computed uncovered surface albedo. The snowcover albedos for diffuse solar radiation depend on surface air temperature \(T_s\) as follows:

\[
\alpha_{uv}(s, \text{dif}) = \begin{cases} 
0.90 & T_s < 272.15 \\
0.90 - .10(T_s - 272.15) & 272.15 \leq T_s \leq 274.15 \\
0.70 & T_s > 274.15
\end{cases}
\]
and the snowcover albedos for direct solar radiation depend on the cosine of
the solar zenith angle $\mu$ as follows:

$$
\begin{align*}
\alpha_v(s, \text{dir}) &= \begin{cases} 
\alpha_v(s, \text{dir}) + (1 - \alpha_v(s, \text{dir})) 0.5 \left( \frac{3}{1+4\mu} - 1 \right) & \mu < 0.5 \\
\alpha_v(s, \text{dir}) & \mu \geq 0.5 
\end{cases} \\
\alpha_n(s, \text{dir}) &= \begin{cases} 
\alpha_n(s, \text{dir}) + (1 - \alpha_n(s, \text{dir})) 0.5 \left( \frac{3}{1+4\mu} - 1 \right) & \mu < 0.5 \\
\alpha_n(s, \text{dir}) & \mu \geq 0.5 
\end{cases}
\end{align*}
$$

Both $\alpha_v(s, \text{dir})$ and $\alpha_n(s, \text{dir})$ are limited to a maximum value of 0.90, and
these snow albedos are then combined with the previously computed albedos
as follows:

$$
\begin{align*}
\alpha_v(v, \text{dir}) &= \alpha_v(g, \text{dir}) = \alpha_v(v, \text{dir})(1 - f_{\text{snow}}) + \alpha_v(s, \text{dir})f_{\text{snow}} \\
\alpha_n(v, \text{dir}) &= \alpha_n(g, \text{dir}) = \alpha_n(v, \text{dir})(1 - f_{\text{snow}}) + \alpha_n(s, \text{dir})f_{\text{snow}} \\
\alpha_v(v, \text{dif}) &= \alpha_v(g, \text{dif}) = \alpha_v(v, \text{dif})(1 - f_{\text{snow}}) + \alpha_v(s, \text{dif})f_{\text{snow}} \\
\alpha_n(v, \text{dif}) &= \alpha_n(g, \text{dif}) = \alpha_n(v, \text{dif})(1 - f_{\text{snow}}) + \alpha_n(s, \text{dif})f_{\text{snow}}
\end{align*}
$$

where $f_{\text{snow}}$ is the fraction of the grid box covered with snow; it is set to 0.5
everywhere there is snowcover except over sea ice, where it is unconditionally
set to 0.

* OCEAN ALBEDO DEPENDS ON ZENITH ANGLE

Compute albedos for open ocean as a function of the cosine of the solar zenith
angle $\mu$. (Defining condition is $F(i, \text{MOCEAN}) = 1$ AND $\mu \geq 0$.)

$$
\begin{align*}
\alpha_v(g, \text{dir}) &= \alpha_n(g, \text{dir}) = 0.06 \\
\alpha_v(g, \text{dif}) &= \alpha_n(g, \text{dif}) = 0.06
\end{align*}
$$

At this point only the following albedos remain at the initialized value of zero:
1) all vegetation albedos for open ocean, and
2) all ground albedos for open ocean iff $\mu < 0$. 

4-5
RADCLW

SUBROUTINE RADCLW

Update deck location: RADCLW.3 - RADCLW.412
Concordance identifier: RDCLW

PURPOSE

Computes the LW fluxes and heating rates.

ALGORITHM

Refer to the NCAR Tech Note "User's Guide to CCM1," (Bath, et al., 1987) for an outline of the control logic for this subroutine. For a mathematical description of the computations, refer to the NCAR Tech Note "Description of NCAR Community Climate Model (CCM1)," (Williamson, et al., 1987).
SUBROUTINE RADC02

Update deck location: RADC02.3 - RADC02.248
Concordance identifier: RDC02

PURPOSE
Computes the carbon dioxide emissivity.

ALGORITHM
For a mathematical description of the computations, refer to the NCAR Tech Note "Description of NCAR Community Climate Model (CCM1)," (Williamson, et al., 1987). For a description of the arrays C02EM and C02EML, see the documentation of COMMON block CRDC02.
SUBROUTINE RADCSW

Update deck location: RADCSW.3 - RADCSW.544
Concordance identifier: RADCSW

PURPOSE

Computes the SW fluxes and heating rates.

ALGORITHM

Refer to the NCAR Tech Note "User's Guide to CCM1," (Bath, et al., 1987) for an outline of the control logic for this subroutine. For a mathematical description of the computations, refer to the NCAR Tech Note "Description of NCAR Community Climate Model (CCM1)," (Williamson, et al., 1987).
SUBROUTINE RADCTL

Update deck location: RADCTL.3 - RADCTL.136
Concordance identifier: RDCTL

PURPOSE

Controls all radiation initialization and computations for the current latitude. This is the only radiation routine directly called by the Model.

ALGORITHM

Refer to the NCAR Tech Note "User's Guide to CCM1," (Bath, et al., 1987) for an outline of the control logic for this subroutine.

NOTE: This subroutine contains Cray Update block if tests using the conditional OZLONG. If OZLONG is defined, then the pressures on σ surfaces are a function of Model surface pressure; otherwise the computed pressures are independent of latitude, longitude, and time.

* BEGINNING OF RUN INITIALIZATION

If this is the first call of the job: call RADINI to initialize time-independent data.

* BEGINNING OF ITERATION INITIALIZATION

If this is the first call (i.e., the first latitude) for this model iteration:

Set full computation flags FRADLV and FRADSW.
If full SW computation, call RADSLR to compute solar parameters from Julian Day.
Print date and solar parameters if this is a full SW or LW computation.
If this is the first iteration of the run OR a full SW or LW computation in an annual cycle run, call RADOZ1 to read and initialize ozone path length data.

For OZLONG defined:

* COMPUTE TIME-DEPENDENT OZONE PATH LENGTH INTEGRALS

Call RADPLV to compute pressures on model sigma surfaces.
Call RADOZ2 to compute the ozone path length for the current latitude.

For OZLONG undefined:

* COMPUTE TIME-INDEPENDENT OZONE PATH LENGTH INTEGRALS

If this is the first call of the run, call RADPLV to compute pressures on model sigma surfaces.
If this is the first iteration of the run, OR it is a full LW or SW computation iteration, call RADOZ2 to compute the ozone path length integrals for the current latitude.

* FILL F ARRAY WITH RADIATION INPUT QUANTITIES
* AND COMPUTE ALBEDOS

Call RADINP.

* SET MINIMUM MIXING RATIOS AND COMPUTE PRESSURE-WEIGHTED.
* VERTICAL PATH LENGTHS FOR WATER VAPOR, CARBON DIOXIDE,
* AND OZONE.

Call RADPTH.

* SOLAR CALCULATIONS

Call RADCSW.

* LONGWAVE CALCULATIONS

Call RADCLW.

* COPY RADIATION OUTPUT QUANTITIES TO MODEL BUFFER

Call RADOUT.
SUBROUTINE RADEMS

Update deck location: RADEMS.3 - RADEMS.168
Concordance identifier: RDEMS

PURPOSE

Computes the water vapor emissivity.

ALGORITHM

For a mathematical description of the computations, refer to the NCAR Tech "Description of NCAR Community Climate Model (CCM1)," (Williamson, et al., 1987).
SUBROUTINE RADINI

Update deck location: RADINI.3 - RADINI.348
Concordance identifier: RDINI

PURPOSE

Initializes spatial resolution parameters and physical constants, and reads time-independent surface albedo data from an external unit.

ALGORITHM

* CHECK DATA STATEMENT INITIALIZATIONS OF ARRAYS DEPENDENT ON VERTICAL RESOLUTION

The arrays CALBR and QFIX (see COMMON block /CRDCON/) are initialized by DATA statements. Their values depend not only on the number of vertical levels in the Model, but also to some extent on where the levels are located. To help ensure that these DATA statements are changed when the vertical resolution is changed, the vertical resolution is checked against a PARAMETER associated with each of these arrays. It is therefore necessary to change these PARAMETERs as well as the DATA statements when changing the vertical resolution.

The arrays CDTH, CDRH, and SMP are also vertically oriented arrays set in DATA statements, but their values are currently vertically uniform, so they are not checked in this way.

* SET RADIATION CONSTANTS

Time independent constants in COMMON blocks /CRDCAE/ and /CRDCON/ are set here with executable statements, or in DATA statements at the beginning of this subroutine.

* READ IN SURFACE ALBEDOS AND FRACTION OF SMOOTH SURFACES.

Call RADRDA five times to read horizontal slices of ALBVSS, ALBVSW, ALBNIS, ALBNIW, and FRCTST from unit 58.

* INDICES FOR F ARRAY

Set indices locating radiation fields in the F array (see COMMON block /CRDNDX/). Compute grid latitudes and some trigonometric functions of latitude, and print the values to NOUT.
SUBROUTINE RADINP

Update deck location: RADINP.3 - RADINP.146
Concordance identifier: RDINP

PURPOSE

Provides the radiation parameterization with the required input fields computed by other parts of the Model. These fields are copied from the Model buffer to the radiation parameterization's F array, with unit conversion and vertical reordering where necessary.

ALGORITHM

* SET COS(ZENITH ANGLE) FOR DIURNAL OR DAILY AVG INSOLATION
* DETERMINE VALUES FOR FRAC AND COSZRS

Call RADZEN to set the effective mean cosine of solar zenith angle (COSZRS), and the daylight fraction (FRAC).
Print the eccentricity factor (ECCF) to NOUT if this is the first latitude of a full SW computation.
Print COSZRS and FRAC at all longitudes to NOUT.

* COMPUTE SURFACE PRESSURE

Surface pressure is copied from the Model buffer and converted from pascals to bars.

\[ \text{PRESSG}(i) = \text{BUF}(\text{NLPSM1} + i) \times 10^{-5} \]

Compute TTOP1 for the current latitude and print the value at all latitudes if the current latitude is the last one, and this is a full computation iteration.

\[ \text{TTOP1}(j) = \frac{1}{\text{nlon}} \sum_{i=1}^{\text{nlon}} \text{BUF}(\text{NTM1} + i) \]

* GET ABSORPTIVITIES FOR PARTIAL COMPUTATION FROM MODEL BUFFER

Retrieve absorptivities stored in the Model buffer, but only for a partial LW computation.

\[ F(i, \text{MABSB} + k - 1) = \text{BUF}(\text{NABSB} + \text{NLONP2}(\text{KXP} - k) + i); \quad 1 \leq k \leq \text{KXP} \]

\[ F(i, \text{MABSB1}) = \text{BUF}(\text{NABSB1} + i) \]

\[ F(i, \text{MABSB2}) = \text{BUF}(\text{NABSB2} + i) \]

\[ F(i, \text{MABSBT}) = \text{BUF}(\text{NABSBT} + i) \]
Copy specific humidity and temperature, and compute pressure at full levels (layer midpoints).

\[
\begin{align*}
F(i, MQNM + k - 1) &= BUF(NQM1 + NLONP2(NLEV - k) + i) \\
F(i, MTNM + k - 1) &= BUF(NTM1 + NLONP2(NLEV - k) + i) \\
F(i, MPBR + k - 1) &= PRESSG(i) \times SIG(NLEV - k + 1) \times 10^6 \\
\end{align*}
\]

Compute pressure at half-levels (layer interfaces).

\[
F(i, MPNM + k - 1) = \begin{cases} 
PRESSG(i) \times 10^6 & k = 1 \\
PRESSG(i) \times SIGKMH(NLEV - k + 1) \times 10^6 & 1 < k \leq KMAX \\
PRESSG(i) \times SIG(1) \times 0.5 \times 10^6 & k = KMAX 
\end{cases}
\]

* SET SURFACE TEMPERATURE

\[
F(i, MTG) = BUF(NTS + i)
\]

* SET FLAGS FOR OCEAN, SEA ICE, AND SNOW COVER

\[
\begin{align*}
F(i, MOCEAN) &= \begin{cases} 
1. & \text{if } \text{INT}(BUF(NORO + i) + .45) = 0 \\
0. & \text{otherwise}
\end{cases} \\
F(i, MSICE) &= \begin{cases} 
1. & \text{if } \text{INT}(BUF(NORO + i) + .45) = 2 \\
0. & \text{otherwise}
\end{cases} \\
F(i, MSCV) &= \begin{cases} 
+1. & \text{if } BUF(NSN + i) > 0. \\
-1. & \text{otherwise}
\end{cases}
\end{align*}
\]

* COPY CLOUDS

Shift clouds one level so that the lowest level in the F array is the surface, which never has clouds.

\[
F(i, MCLD + k) = \begin{cases} 
BUF(NCLD + NLONP2(NLEV - k) + i) & 1 \leq k \leq KMAX \\
0. & k = 0
\end{cases}
\]

* COMPUTE ALBEDOS

Call RADALB.
SUBROUTINE RAD03E(LATIT)

Update deck location: RAD03E.3 - RAD03E.69
Concordance identifier: RD03E

PURPOSE
Calculates ozone band absorption and emissivity.

ARGUMENT
LATIT : [input] north to south index for current latitude (integer)

ALGORITHM
NOTE: This subroutine contains Cray Update block if tests using the conditional OZLONG. If OZLONG is defined, then the ozone path length integrals are a function of longitude; otherwise they are zonally uniform.

* CALCULATE OZONE BAND ABSORPTION BETWEEN LEVEL K AND INFINITY
  Fill arrays DBVT and H2OTr as described in the documentation for COMMON block /CRDS03/.

* CALCULATE OZONE EMISSIVITY
  Fill arrays O3EMS, DBVTIT, and DBVTLY as described in the documentation for COMMON block /CRDS03/.
RADOUT

SUBROUTINE RADOUT

Update deck location: RADOUT.3 - RADOUT.142
Concordance identifier: RDOUT

PURPOSE

Moves the radiation parameterization's output fields from the F array to the Model's buffer, with unit conversion and vertical reordering where necessary. Some fields are accumulated by adding to (instead of overwriting) the contents of the Model buffer.

ALGORITHM

* STORE DOWNWARD RADIATION AND EMISSIVITIES

Unconditionally copy the total net downward radiative flux at the surface into the Model buffer.

\[ \text{BUF}(\text{NDRP}1 + i) = F(i,\text{MSLWD}) \]

For full LW computations only, store absorptivities and copy LW heating rates.

\[ \text{BUF}(\text{NABSB} + \text{NLONP2}(\text{KMXP} - k) + i) = F(i,\text{MABSB} + k - 1) \quad 1 \leq k \leq \text{KMXP} \]

\[ \text{BUF}(\text{NABSB} + i) = F(i,\text{MABSB}1) \]

\[ \text{BUF}(\text{NABSB}2 + i) = F(i,\text{MABSB}2) \]

\[ \text{BUF}(\text{NABSBT} + i) = F(i,\text{MABSB}T) \]

\[ \text{BUF}(\text{NQRL} + \text{NLONP2}(\text{KMAX} - k) + i) = F(i,\text{MQRL} + k - 1) \quad 1 \leq k \leq \text{KMAX} \]

For full SW computations only, copy SW heating rates.

\[ \text{BUF}(\text{NQRS} + \text{NLONP2}(\text{KMAX} - k) + i) = F(i,\text{MQRS} + k - 1) \quad 1 \leq k \leq \text{KMAX} \]

* ACCUMULATED QUANTITIES FOR HISTORY TAPE

* ACCUMULATE SOLAR STATISTICS

For full SW computations only, add SW fields to the Model buffer, changing units from \( \text{erg} \cdot \text{cm}^{-2} \cdot \text{sec}^{-1} \) to \( \text{watt} \cdot \text{m}^{-2} \).

\[ \text{BUF}(\text{NFRSA} + i) = \text{BUF}(\text{NFRSA} + i) + F(i,\text{MFRS}) \times 10^{-3} \]

\[ \text{BUF}(\text{NSABTP} + i) = \text{BUF}(\text{NSABTP} + i) + F(i,\text{MSABTP}) \times 10^{-3} \]

\[ \text{BUF}(\text{MALBT} + i) = \text{BUF}(\text{MALBT} + i) + F(i,\text{MALBT}) \times 10^{-3} \]

\[ \text{BUF}(\text{MCLRSS} + i) = \text{BUF}(\text{MCLRSS} + i) + F(i,\text{MCLRSS}) \times 10^{-3} \]

\[ \text{BUF}(\text{MCLRST} + i) = \text{BUF}(\text{MCLRST} + i) + F(i,\text{MCLRST}) \times 10^{-3} \]

\[ \text{BUF}(\text{MSOLIN} + i) = \text{BUF}(\text{MSOLIN} + i) + F(i,\text{MSOLIN}) \times 10^{-3} \]

Increment accumulation counter \( \text{NACSW} \) (once per iteration).
* ACCUMULATE LW STATISTICS

For full LW computations only, add LW fields to the Model buffer, changing units from erg\cdot cm^{-2}\cdot sec^{-1} to watt\cdot m^{-2}.

\[
\begin{align*}
\text{BUF(NFRLA + i)} &= \text{BUF(NFRLA + i)} + F(i, MFRL) \times 10^{-3} \\
\text{BUF(NFIRTP + i)} &= \text{BUF(NFIRTP + i)} + F(i, MFIRTP) \times 10^{-3} \\
\text{BUF(NCLRRLS + i)} &= \text{BUF(NCLRRLS + i)} + F(i, MCLRRLS) \times 10^{-3} \\
\text{BUF(NCLRRLT + i)} &= \text{BUF(NCLRRLT + i)} + F(i, MCLRRLT) \times 10^{-3}
\end{align*}
\]

Increment accumulation counter \text{NACLW}(once per iteration).
SUBROUTINE RADOZ1

Update deck location: RADOZ1.3 - RADOZ1.147
Concordance identifier: RDOZ1

PURPOSE

Determines ozone path length integrals on pressure surfaces (arrays PLOSO and PLOLO) for the current Model time.

ALGORITHM

NOTE: All vertical indices in this subroutine increase from top to bottom.

* SET CONSTANTS ON FIRST CALL

Execute an immediate RETURN unless this is the first entry into this subroutine, or the annual cycle option is set.

If this is the first entry, set some time-independent constants used in the path length integral computations.

* GET APPROPRIATE OZONE DATA

Two monthly averages for the ozone data are required in order to interpolate to the current Model date and time. If the correct two monthly averages are currently being held in memory, skip input of ozone mixing ratios and proceed directly to computation of path lengths (see below). If this is the first entry into this subroutine, read the previous month’s data from unit NOZONE into array OZMIXN. (This is a latitude-height array since the ozone data are zonally averaged.) Also read the pressures locating the data levels into array POZ, and print these pressure levels, along with the MONTH and the number of pressure levels (KOZ), to unit NOUT.

Set the layer interface pressures.

\[ PIN(k) = \begin{cases} 
0.0 & k = 1 \\
0.5(POZ(k - 1) + POZ(k)) & 2 \leq k \leq KOZ \\
POZ(KOZ) & k = KOZ + 1 
\end{cases} \]

* COPY OLD NEXT MONTH MIXING RATIOS TO LAST MONTH

Copy OZMIXN to OZMIXL.

* READ THE NEXT MONTH’S OZONE DATA.

Read the monthly averaged ozone data into OZMIXN.

* FIND THE PATH LENGTHS FOR THIS DAY

Linearly interpolate to the current Model date and time using the monthly averages contained in OZMIXL and OZMIXN; place the result in array OZMIX.

4-18
Compute the ozone path length integrals at each latitude $j$, counting from north to south.

\[ \text{PLOSO}(j, k) = \begin{cases} 0.0 & k = 1 \\ \text{CPLOS} \sum_{l=2}^{k} \text{OZMIX}(j, l - 1)(p_l - p_{l-1}) & 2 \leq k \leq \text{KOZ} + 1 \end{cases} \]

\[ \text{PLOLO}(j, k) = \begin{cases} 0.0 & k = 1 \\ \text{CPLOL} \sum_{l=2}^{k} \text{OZMIX}(j, l - 1)(p_l^2 - p_{l-1}^2) & 2 \leq k \leq \text{KOZ} + 1 \end{cases} \]

where

\[ \text{CPLOS} = 100 \frac{V_0}{gM_d} \quad \text{CPLOL} = 100 \frac{V_0}{2gM_dp_0} \quad p_k = \text{PIN}(k) \]

and

$V_0$ is the volume of a gas at STP ($22.4136$ m$^3$·kmol$^{-1}$),
$g$ is the acceleration due to gravity ($9.80616$ m·s$^{-2}$),
$M_d$ is the effective molecular weight of dry air ($28.9644$ kg·kmol$^{-1}$), and
$p_0$ is standard atmospheric pressure ($1.01325 \times 10^5$ pascals).

The factor of 100 in the expressions for CPLOS and CPLOL converts the path length integrals to the cgs unit system used by the radiation code.

Print ozone update message to NOUT.
RADOZ2

SUBROUTINE RADOZ2(J)

Update deck location: RADOZ2.3 - RADOZ2.110
Concordance identifier: RDOZ2

PURPOSE

Interpolates the ozone path length integrals from pressure surfaces (arrays PLOSO and PLOLO) to Model sigma surfaces (arrays PLOS and PLOL) for the current Model time.

ARGUMENT

J : [input] north to south index for current latitude (integer)

ALGORITHM

NOTE: This subroutine contains Cray Update block if tests using the conditional OZLONG. If OZLONG is defined, then the ozone path length integrals are a function of longitude; otherwise they are zonally uniform.

* INTERPOLATE TO MODEL SURFACES

For each Model layer interface k, locate the bracketing ozone pressure surfaces $k_1$ and $k_1 + 1$. Note that $k$ increases upward and refers to levels of PLOS, PLOL, and the Model layer interface (half level) pressures PRSI, while $k_1$ increases downward and refers to levels of PLOSO, PLOLO, and the ozone mixing ratio's layer interface pressures PIN. OZMIX at level $k_1$ is the ozone mixing ratio for the pressure layer between levels $k_1$ and $k_1 + 1$ of PIN.

\[
\begin{align*}
\sigma \text{ half level} & : k+1 & \text{ PLOS, PLOL, PRSI} \\
\text{pressure level} & : k_1 & \text{ PLOSO, PLOLO, PIN} \\
\sigma \text{ half level} & : k & \text{ PLOS, PLOL, PRSI} \\
\text{pressure level} & : k_1 + 1 & \text{ PLOSO, PLOLO, PIN}
\end{align*}
\]

Compute the ozone path length integrals on $\sigma$ half levels. For OZLONG undefined, computations are done for each level $k$ at the current latitude $j$:

\[
\begin{align*}
PLOS(j,k) &= PLOSO(j,k_1) + CPLOS \times OZMIX(j,k_1)(PRSI(k) - PIN(k_1)) \\
PLOL(j,k) &= PLOLO(j,k_1) + CPLOL \times OZMIX(j,k_1)(PRSI^2(k) - PIN^2(k_1))
\end{align*}
\]

For OZLONG defined, computations are done for each level $k$ and longitude $i$, for the current latitude:

\[
\begin{align*}
PLOS(i,k) &= PLOSO(i,k_1) + CPLOS \times OZMIX(i,k_1)(PRSI(i,k) - PIN(k_1)) \\
PLOL(i,k) &= PLOLO(i,k_1) + CPLOL \times OZMIX(i,k_1)(PRSI^2(i,k) - PIN^2(k_1))
\end{align*}
\]

If the lowest available ozone mixing ratio is above ground level, that mixing ratio is assumed to extend down to the ground. CPLOS and CPLOL are defined as in RADOZ1.
SUBROUTINE RADPLV(PS,SIG,SIGKMH)

Update deck location: RADPLV.3 - RADPLV.61
Concordance identifier: RDPLV

PURPOSE
Computes pressures on Model $\sigma$ layer interfaces (half levels), to be used only in computing ozone path length integrals.

ARGUMENTS

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PS(PLON)</td>
<td>[input] surface pressure in pascals (real)</td>
</tr>
<tr>
<td>SIG(PLEV)</td>
<td>[input] full $\sigma$ values for layer midpoints, top to bottom (as in /COMMAP/) (real)</td>
</tr>
<tr>
<td>SIGKMH(PLEV)</td>
<td>[input] half $\sigma$ values for layer interfaces, top to bottom (as in /COMMAP/) (real)</td>
</tr>
</tbody>
</table>

ALGORITHM

NOTE: This subroutine contains Cray Update block if tests using the conditional OZLONG. If OZLONG is defined, then the pressures on $\sigma$ surfaces are a function of Model surface pressure; otherwise the computed pressures are independent of latitude, longitude, and time.

Compute the interface pressures on $\sigma$ half levels. Note that the vertical index for $\sigma$ increases downwards, while the vertical index for PRSI increases upwards. For OZLONG undefined, PRSI is computed only once at each level $k$:

$$
PRSI(k) = \begin{cases} 
\sigma_{NLEV+1-k} \times 10^5 & \text{for } 1 \leq k \leq NLEV \\
0.5\sigma_1 \times 10^5 & \text{for } k = NLEV + 1 
\end{cases}
$$

For OZLONG defined, PRSI is computed at all longitudes $i$ and levels $k$ for the current latitude:

$$
PRSI(i,k) = \begin{cases} 
\sigma_{NLEV+1-k}PS(i) & \text{for } 1 \leq k \leq NLEV \\
0.5\sigma_1PS(i) & \text{for } k = NLEV + 1 
\end{cases}
$$

where $\sigma_{NLEV+1-k} = SIGKMH(NLEV + 2 - k)$ and $\sigma_1 = SIG(1)$. SIGKMH and SIG are passed to RADPLV as arguments, but they are actually the arrays of the same name in the COMMON block /COMMAP/.
SUBROUTINE RADPTH

Update deck location: RADPTH.3- RADPTH.72
Concordance identifier: RDPTH

PURPOSE

Sets mixing ratios to a minimum value, computes vertical path length integrals for water vapor and carbon dioxide, and computes clear sky fractions at all levels.

ALGORITHM

* SET MINIMUM MIXING RATIOS.
* FOR THIS PURPOSE, KLQL DEFINES THE BOUNDARY TROPOSPHERE AND THE STRATOSPHERE.

Modify mixing ratios in the F array at all longitudes (i) and levels (k) for the current latitude. Model buffer values are unaffected.

\[
q_k = \begin{cases} 
\max(q_k, Q\text{FIX}(k)) & \text{for } k < KLQL \\
\max(q_k, 10^{-7}) & \text{for } k \geq KLQL 
\end{cases}
\]

where \( q_k \) refers to \( F(i,NQNM + k - 1) \), Q\text{FIX} is the vertical array of constants in /CRDCON/, and KLQL is a PARAMETER set to 7 in Common Deck PRDRES.

* CALCULATE:
* PLH20 - PATH LENGTH OF WATER VAPOR (VERTICAL, PRES-WEIGHTED)
* PLC02 - PATH LENGTH OF CARBON DIOXIDE (VERTICAL, P-WEIGHTED)
* TCLRSF - PRODUCT OF THE CLEAR-SKY FRACTIONS FROM THE TOP OF THE ATMOSPHERE TO LEVEL K
* SET TOP LAYER H2O MIXING RATIO TO Q\text{FIX}(K\text{MAXP1})

Compute the arrays PLH20, PLC02, and TCLRSF as described in the documentation for COMMON block /CRDPHT/. 

4–22
SUBROUTINE RADRDA(NUNIT, A, IMX, JMX, CONV)

Update deck location: RADRDA.3 - RADRDA.66
Concordance identifier: RDRDA

PURPOSE

Reads the next field from the external albedo dataset.

ARGUMENTS

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUNIT</td>
<td>logical unit number for external albedo dataset (integer)</td>
</tr>
<tr>
<td>A(IMX, JMX)</td>
<td>horizontal slice of albedo field read from NUNIT, multiplied by CONV (real)</td>
</tr>
<tr>
<td>IMX</td>
<td>number of unique longitudes in the Model (integer)</td>
</tr>
<tr>
<td>JMX</td>
<td>number of latitudes in the Model (integer)</td>
</tr>
<tr>
<td>CONV</td>
<td>conversion factor (field read from external unit is multiplied by this value) (real)</td>
</tr>
</tbody>
</table>

ALGORITHM

Read 20-word header from NUNIT and check for horizontal resolution consistency. Abort with error message if check fails.

Read horizontal slice into A as JMX records, each of length IMX. Longitudes are assumed to be ordered west to east, starting at Greenwich, and the latitude records are assumed to be ordered north to south.

Multiply A by CONV, except where the value is $\geq 10^{30}$, which indicates missing data.
RADSLR

SUBROUTINE RADSLR(JD,FJD,R,DLT,ALP,SLAG,N,HANG,TAUDA,COSZ,ALAT)

Update deck location: RADSLR.3 - RADSLR.104
Concordance identifier: RDSLR

PURPOSE
Computes solar position parameters for the current date.

ARGUMENTS

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JD</td>
<td>[input]</td>
<td>integral part of current julian day (integer)</td>
</tr>
<tr>
<td>FJD</td>
<td>[input]</td>
<td>fractional part of current julian day (real)</td>
</tr>
<tr>
<td>R</td>
<td>[output]</td>
<td>radius of earth's orbit normalized by the semi-major axis of the orbit (real)</td>
</tr>
<tr>
<td>DLT</td>
<td>[output]</td>
<td>earth's declination (real)</td>
</tr>
<tr>
<td>ALP</td>
<td>[output]</td>
<td>not used (real)</td>
</tr>
<tr>
<td>SLAG</td>
<td>[output]</td>
<td>not used (real)</td>
</tr>
<tr>
<td>N</td>
<td>[input]</td>
<td>number of latitudes (integer)</td>
</tr>
<tr>
<td>HANG(N)</td>
<td>[output]</td>
<td>half-day length in degrees (not used) (real)</td>
</tr>
<tr>
<td>TAUDA(N)</td>
<td>[output]</td>
<td>half-day length in radians (not used) (real)</td>
</tr>
<tr>
<td>COSZ(N)</td>
<td>[output]</td>
<td>cosine of the zenith angle (real)</td>
</tr>
<tr>
<td>ALAT(N)</td>
<td>[output]</td>
<td>latitude in radians (real)</td>
</tr>
</tbody>
</table>
SUBROUTINE RADTPL

Update deck location: RADTPL.3 - RADTPL.113
Concordance identifier: RDTPL

PURPOSE
Fills some arrays used in the LW computations, including those related to Planck function evaluation.

ALGORITHM

* CHANGE UNITS OF CO2 AMOUNT FROM CM-ATM TO GM.CM-2
  Multiply all elements of PLC02 by $1.9642 \times 10^{-3}$. See documentation for COMMON block /CRDPTH/.

* SET THE TOP AND BOTTOM INTERMEDIATE LEVEL TEMPERATURES, TOP LEVEL
* PLANCK TEMPERATURE AND TOP LAYER TEMP**4.
* TINT IS LOWER INTERFACE TEMPERATURE
* (NOT AVAILABLE FOR BOTTOM LAYER, SO USE GROUND TEMPERATURE)

* INTERMEDIATE LEVEL TEMPERATURES ARE CALCULATED BASED ON THE TEMP
* AT THE FULL LEVEL BELOW LESS DY*DELTA T (BETWEEN THE FULL LEVELS).

* NOW SET THE LAYER TEMP=FULL LEVEL TEMPERATURES AND ESTABLISH A
* PLANCK TEMPERATURE FOR ABSORPTION (TPLNKA) WHICH IS THE AVERAGE OF
* THE INTERMEDIATE LEVEL TEMPERATURES. NOTE THAT TPLNKA IS NOT
* EQUAL TO THE FULL LEVEL TEMPERATURES.

* NOW CALCULATE TPLANK FOR EMISSIVITY CALCULATION
* ASSUME ISOThERMAL TPLNKE I.E. ALL LEVELS=TTOP
  Fill the following temperature related arrays (see the documentation for the appropriate COMMON block for details): TLAYR, TLAYR4, TINT, and TINT4 in /CRDTMP/; TPLNKA in /CRDABS/; and TPLNKE in /CRDEMS/.

* NOW COMPUTE PLH20 * TLAYER = S2T(I,K)
  Fill the path length related arrays S2C, S2T, and W. See the documentation for COMMON block /CRDEMS/ for details. STEMP and SCTEMP are local arrays used as accumulators in the vertical integration for S2T and S2C, respectively.
SUBROUTINE RADZEN(FJD,R,DLT,SLAG,XLAT,HA,DHR,NLNG,COSZRS,FRAC)

Update deck location: RADZEN.3 - RADZEN.83
Concordance identifier: RDZEN

PURPOSE
Computes effective mean cosine of solar zenith angle and daylight fraction from solar position parameters.

ARGUMENTS

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FJD</td>
<td>[input]</td>
<td>fractional part of current julian day (real)</td>
</tr>
<tr>
<td>R</td>
<td>[input]</td>
<td>radius of earth's orbit normalized by the semi-major axis of the orbit (real)</td>
</tr>
<tr>
<td>DLT</td>
<td>[input]</td>
<td>earth's declination (real)</td>
</tr>
<tr>
<td>SLAG</td>
<td>[input]</td>
<td>not used (real)</td>
</tr>
<tr>
<td>XLAT</td>
<td>[input]</td>
<td>latitude in radians (real)</td>
</tr>
<tr>
<td>HA</td>
<td>[input]</td>
<td>half-day length in degrees (not used) (real)</td>
</tr>
<tr>
<td>DHR</td>
<td>[input]</td>
<td>not used (real)</td>
</tr>
<tr>
<td>NLNG</td>
<td>[input]</td>
<td>number of unique longitudes (integer)</td>
</tr>
<tr>
<td>COSZRS(NLNG)</td>
<td>[output]</td>
<td>cosine of the zenith angle (real)</td>
</tr>
<tr>
<td>FRAC(NLNG)</td>
<td>[output]</td>
<td>fractional amount of daytime (real)</td>
</tr>
</tbody>
</table>
SUBROUTINE CLDCMP

Update deck location: CLDCMP.3 - CLDCMP.124
Concordance identifier: CLDCM

PURPOSE
Computes clouds and their emissivities for the current latitude.

ALGORITHM
NOTE: For the cloud computations, Model layers are numbered from bottom to top. The vertical index in the CLOUD array locates layer interfaces (half-levels), starting at the surface and increasing upward. Cloud amounts for layer $n$ are assigned to the level at the top interface of the layer, which corresponds to a CLOUD subscript of $n+1$. Clouds are allowed to form only for cloud indices $k$ where $KLDL \leq k \leq KLDH$, and the parameters $KLDL$ and $KLDH$ are set to 3 and 10, respectively, in Common Deck PRDRES.

* INITIALIZE CLOUD EMISSIVITIES TO 1.
  Initialize array CLOUDE(i, k) to 1. everywhere.

* CLOUDS NOT CALCULATED OVER OCEANS IF PARTIAL CALCULATION
  * IN RADIATION

* FULL COMPUTATION OR NOT OVER OCEAN
The rest of this subroutine is skipped for gridpoints over open ocean during a partial radiation computation.
Initialize array CLOUD(i, k) to 0. everywhere.
The remainder of this subroutine is divided into two parts: one for stratiform clouds, and another for convective clouds. Each vertical column may have only one of the two types of clouds.

* STRATIFORM CLOUD CALCULATIONS
Stratiform clouds are formed at level $k$ ($KLDL \leq k \leq KLDH$) if, and only if, the following two conditions are met:
1) no levels have been convectively adjusted (condition is $KLO(i) \leq 0$ AND $KCHI(i) \leq 0$), and
2) there is condensation in the layer below level $k$ (condition is $QC(i, k-1) > 0$).

For the lowest two levels at which clouds are allowed to form ($KLDL \leq k \leq KLDL+1$), condition (1) above still applies, but condition (2) is relaxed somewhat, and clouds are formed at both levels ($KLDL$ and $KLDL+1$) if there is condensation in either of the two layers $KLDL$ or $KLDL+1$ (condition is $QC(i, KLDL - 1) > 0$ OR $QC(i, KLDL) > 0$). If the appropriate cloud-forming conditions are satisfied, then the
stratiform cloud amount and emissivity are computed as follows:

$$\text{CLOUD}(i, k) = \text{CMAX} \quad \text{for KLDL} \leq k \leq \text{KLDH}$$

$$\text{CLOUDE}(i, k) = \frac{1 - \exp\left(-\min(1000 \cdot QC(i, k - 1), 2.5)\right)}{1 - \exp(-2.5)} \quad \text{for KLDL} + 2 \leq k \leq \text{KLDH}$$

where CMAX = 0.95, and other levels remain at their initialized values.

If there was no convective adjustment at any level (see condition 1 above), then the rest of this subroutine is skipped.

* **CONVECTIVE CLOUD LOOP**

Set convective cloud thickness CLTHK to the number of layers between the lowest and highest convectively adjusted layers, inclusive.

$$\text{CLTHK} = KCHI(i) - \text{KCLO}(i) + 1$$

* **SET MINIMUM LOW LEVEL CLOUD THICKNESS TO 2**

If the lowest convectively adjusted layer is exactly two layers below the layer of lowest permissible clouds and the cloud thickness is only 1. (condition is $\text{KCLO}(i) + 2 = \text{KLDL}$ AND CLTHK = 1.), then increase cloud thickness: CLTHK = 2.

* **SET MINIMUM THICKNESS AT ANY OTHER LEVEL TO 1**

$$\text{CLTHK} = \max(\text{CLTHK}, 1.)$$

* **SET MAXIMUM CUMULUS ACCUMULATION FOR EACH LEVEL**

Set cumulus cloud amount for all permissible cloud levels $k$ ($\text{KLDL} \leq k \leq \text{KLDH}$):

$$\text{CLOUD}(i, k) = \frac{\text{CMNBMX}}{\text{CLTHK}} \quad \text{for } \text{KCLO}(i) + 2 \leq k \leq \text{KCHI}(i) + 2$$

and if there is condensation in the layer ($QC(i, k - 1) > 0$),

$$\text{CLOUD}(i, k) = \text{CMAX} \quad \text{for } k > \text{KCHI}(i) + 2 \quad \text{OR} \quad k < \text{KCLO}(i) + 1$$

where CMNBMX = 0.3 and CMAX = 0.95 as for the stratiform clouds.

* **CHECK TO SEE IF LEVEL 3 CLOUD IS GREATER THAN LEVEL 4 CLOUD AND**

* **IF SO SET LEVEL 4 AMOUNT TO LEVEL 3 AMOUNT**

$$\text{CLOUD}(i, \text{KLDL} + 1) = \max(\text{CLOUD}(i, \text{KLDL} + 1), \text{CLOUD}(i, \text{KLDL}))$$
SUBROUTINE CLDCTL

Update deck location: CLDCTL.3 - CLDCTL.23
Concordance identifier: CLDCTL

PURPOSE
 Controls all cloud initialization and computations for the current latitude. This is the only cloud routine directly called by the Model.

ALGORITHM

* COPY INPUT QUANTITIES TO TEMPORARY WORKSPACE
  Call CLDINP.

* COMPUTE CLOUDS
  Call CLDCMP.

* COPY OUTPUT QUANTITIES TO MODEL BUFFER
  Call CLDOUT.
SUBROUTINE CLDINP

Update deck location: CLDINP.3 - CLDINP.50
Concordance identifier: CLDIN

PURPOSE

Provides the cloud parameterization with the required input fields computed by other parts of the Model. These fields are copied from the Model buffer to the cloud parameterization's version of the COMMON block /WORKSP/, with modification and vertical reordering where necessary.

ALGORITHM

* SET CONVECTIVE ADJUSTMENT LEVEL FLAGS

Transform the values so that the level count is from bottom to top, setting both levels to zero if there was no convective adjustment.

\[
KCL0(i) = \begin{cases} 
NLEV + 1 - BUF(\text{NCNB} + i) & \text{BUF(\text{NCNB} + i)} \neq 0 \\
0 & \text{BUF(\text{NCNB} + i)} = 0
\end{cases}
\]

\[
KCHI(i) = \begin{cases} 
NLEV + 1 - BUF(\text{NCNT} + i) & \text{BUF(\text{NCNT} + i)} \neq 0 \\
0 & \text{BUF(\text{NCNT} + i)} = 0
\end{cases}
\]

* SET CONVECTIVE ADJUSTMENT CONDENSATION RATE AND CLOUDS

Convert the condensation rate from \(\text{kg-m}^{-2}\cdot\text{s}^{-2}\) to \(\text{g}\cdot\text{cm}^{-2}\cdot(\text{time step})^{-1}\) using the value of \(2\Delta t\) (TWODT) from /COMSIDS/.

\[
QC(i, k) = BUF(NQC + \text{NLONP2}(NLEV - k) + i)(0.1)(\text{TWODT})(0.5)
\]

where \(1 \leq k \leq \text{NLEV}\).

Copy clouds for \(1 \leq k \leq \text{NLEV}\), shifting one level so that the lowest level in the CLOUD array is the surface, which never has clouds and is not explicitly set.

\[
\text{CLOUD}(i, k + 1) = BUF(NCLD + \text{NLONP2}(NLEV - k) + i)
\]

* SET OCEAN FLAG

Set the logical array OCEAN to .TRUE. iff the gridpoint is open ocean.

\[
\text{OCEAN}(i) = \begin{cases} 
\text{.TRUE.} & \text{if } \text{INT}(BUF(\text{NORO} + i) + .45) = 0 \\
\text{.FALSE.} & \text{otherwise}
\end{cases}
\]
SUBROUTINE CLDOUT

Update deck location: CLDOUT.3 - CLDOUT.35
Concordance identifier: CLDOT

PURPOSE

Moves the cloud parameterization's output fields from the COMMON block /WORKSP/ to the Model buffer, with vertical reordering where necessary.

ALGORITHM

* COPY CLOUDS AND CLOUDE

Copy clouds and cloud emissivity to Model buffer for $1 \leq k \leq \text{NLEV}$, inverting the cloud levels and eliminating the unused surface level.

\[
\begin{align*}
\text{BUF}(\text{NCLD} + \text{NLONP2}((\text{NLEV} - k) + i)) &= \text{CLOUD}(i, k + 1) \\
\text{BUF}(\text{NCLE} + \text{NLONP2}((\text{NLEV} - k) + i)) &= \text{CLOUDE}(i, k + 1)
\end{align*}
\]
SECTION 5

Many of the radiation routines use common decks to transfer information from one subroutine to another. This section describes these common decks in detail. The name of the common deck is given along with a description of what subroutines use these common blocks. The variables contained in the common decks along with a description of each of these variables is also provided.
$COMDECK CRDABS
C
CL          LW ABSORPTIVITIES FOR INDIVIDUAL GASES
PLU   SOME RELATED QUANTITIES
C

All of the arrays in this COMMON block are set in subroutine RADABS with the exception of TPLNKA, which is set in RADTPL. For additional computational details, refer to the NCAR Technical Note "Description of the NCAR Community Climate Model (CCM1)" (Williamson, et al., 1987).

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>H2OABS(IMX,KMXP,KMXP)</td>
<td>Total LW absorptivities between half levels (H₂O + CO₂ + O₃)</td>
</tr>
<tr>
<td>TPATHA(IMX)</td>
<td>Scratch array used only in RADABS</td>
</tr>
<tr>
<td>TPLNKA(IMX,KMXP)</td>
<td>TPLNKA(i,k) =</td>
</tr>
<tr>
<td></td>
<td>({ 0.5 \left( \text{TINT}(i, k) + \text{TINT}(i, k + 1) \right) ), (1 \leq k \leq \text{KMAX})</td>
</tr>
<tr>
<td></td>
<td>(\text{T}_{\text{KMAX}}), (k = \text{KMAX} + 1)</td>
</tr>
<tr>
<td></td>
<td>where (\text{TINT}(k)) is an array in /CRDTMP/</td>
</tr>
<tr>
<td>H2OAB1(IMX,KMX,4)</td>
<td>LW absorptivities between adjacent levels for H₂O</td>
</tr>
<tr>
<td>CO2AB1(IMX,KMX,4)</td>
<td>LW absorptivities between adjacent levels for CO₂</td>
</tr>
<tr>
<td>O3AB1(IMX,KMX,4)</td>
<td>LW absorptivities between adjacent levels for O₃</td>
</tr>
</tbody>
</table>
$COMDECK CRDACL
C
CL CLOUD EMISSIVITY FOR LW COMPUTATIONS
C

This COMMON block contains only array CLOUDE. These emissivities are computed in subroutine CLDCMP and used in subroutines CLDOUT and RADCLW. See description of subroutine CLDCMP for computational details.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Variable Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLOUDE(PLON,PLEVP)</td>
<td>Real</td>
<td>Cloud emissivities</td>
</tr>
</tbody>
</table>

5-3
All of the arrays in this COMMON block are set in subroutine RADINI via calls to RADRDA, which reads the data from input unit 58.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALBVSS(IMX, JMX)</td>
<td>Visible albedo for strong zenith angle dependence</td>
</tr>
<tr>
<td>ALBVSW(IMX, JMX)</td>
<td>Visible albedo for weak zenith angle dependence</td>
</tr>
<tr>
<td>ALBNIS(IMX, JMX)</td>
<td>Near infrared albedo for strong zenith angle dependence</td>
</tr>
<tr>
<td>ALBNIW(IMX, JMX)</td>
<td>Near infrared albedo for weak zenith angle dependence</td>
</tr>
<tr>
<td>FRCTST(IMX, JMX)</td>
<td>Fraction of grid box with strong zenith angle dependence for albedo</td>
</tr>
</tbody>
</table>
C
CL WATER VAPOR NARROW BAND CONSTANTS FOR LW COMPUTATIONS
C

All arrays in this COMMON block are set in subroutine RADINI with a DATA statement. FREQ is set, but never used.

All arrays are dimensioned with a length of 2: the first value is for the 800–1000 cm\(^{-1}\) band, and the second is for the 1000–1200 cm\(^{-1}\) band. Each array is a different parameter used in the computation of the LW emissivities and absorptivities for water vapor; see Ramanathan, V., and Downey, P., 1986: A nonisothermal emissivity and absorptivity formulation for water vapor. *J. Geophys. Res.*, 91, 8649–8666.

The following table gives approximate values for all of the arrays; the code defines the values to 14 decimal digits.

<table>
<thead>
<tr>
<th></th>
<th>band 1</th>
<th>band 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>FREQ</td>
<td>900</td>
<td>1100</td>
</tr>
<tr>
<td>REALK</td>
<td>(1.8967 \times 10^{-5})</td>
<td>(7.0172 \times 10^{-5})</td>
</tr>
<tr>
<td>ST</td>
<td>(3.1930 \times 10^{-4})</td>
<td>(9.7907 \times 10^{-4})</td>
</tr>
<tr>
<td>A1</td>
<td>(2.8775 \times 10^{-2})</td>
<td>(2.3237 \times 10^{-2})</td>
</tr>
<tr>
<td>A2</td>
<td>(-5.7967 \times 10^{-5})</td>
<td>(-9.5105 \times 10^{-5})</td>
</tr>
<tr>
<td>B1</td>
<td>(2.9928 \times 10^{-2})</td>
<td>(2.1737 \times 10^{-2})</td>
</tr>
<tr>
<td>B2</td>
<td>(-8.6322 \times 10^{-5})</td>
<td>(-7.8544 \times 10^{-5})</td>
</tr>
</tbody>
</table>
Except for CON1 and CON2, all of the variables in this COMMON block are set to numeric constants in subroutine RADINI, primarily with DATA statements. For details on the usage and values of these variables, refer to a code listing and the description of the absorptivity and emissivity formulations; also see Ramanathan, V., and Downey, P., 1986: A nonisothermal emissivity and absorptivity formulation for water vapor. J. Geophys. Res., 91, 8649–8666. CON1 and CON2 are also set in RADINI, as described below. See the description of COMMON block /CRDINT/ for full definitions of UINPL and WINPL.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CON1(2,1,k) = 1</td>
<td>for 1 ≤ k ≤ KMAX + 1</td>
</tr>
<tr>
<td>CON1(2,2,k) = UINPL(2,k)</td>
<td>for 1 ≤ k ≤ KMAX</td>
</tr>
<tr>
<td>CON1(2,3,k) = UINPL(3,k)</td>
<td>for 1 ≤ k ≤ KMAX</td>
</tr>
<tr>
<td>CON1(2,4,k) = UINPL(4,k)</td>
<td>for 1 ≤ k ≤ KMAX</td>
</tr>
<tr>
<td>CON2(2,1,k) = WINPL(1,k)</td>
<td>for 1 ≤ k ≤ KMAX</td>
</tr>
<tr>
<td>CON2(2,2,k) = WINPL(2,k)</td>
<td>for 1 ≤ k ≤ KMAX</td>
</tr>
<tr>
<td>CON2(2,3,k) = WINPL(3,k)</td>
<td>for 1 ≤ k ≤ KMAX</td>
</tr>
<tr>
<td>CON2(2,4,k) = WINPL(4,k)</td>
<td>for 1 ≤ k ≤ KMAX</td>
</tr>
</tbody>
</table>

All other values are undefined.
All of the arrays in this COMMON block are set in subroutine RADCO2.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO2EMS(IMX,KMXP)</td>
<td>CO₂ emissivity. See the NCAR Technical Note “Description of the NCAR Community Climate Model (CCM1)” (Williamson, et al., 1987) for details.</td>
</tr>
<tr>
<td>CO2EM(IMX,KMXP)</td>
<td>Derivative of the Planck function with respect to temperature at layer interfaces (half levels), at the center of the CO₂ band (667 cm⁻¹), normalized by 4σT³.</td>
</tr>
<tr>
<td>CO2EML(IMX,KMX)</td>
<td>Derivative of the Planck function with respect to temperature at layer midpoints (full levels), at the center of the CO₂ band (667 cm⁻¹), normalized by 4σT³.</td>
</tr>
</tbody>
</table>

\[
CO2EM(i, k) = 1.2 \times 10^{11} \frac{\exp(960/T_{int})}{T_{int}^5 (\exp(960/T_{int}) - 1)^2}
\]

where \( T_{int} = TINT(i, k) \) from /CRDTMP/

\[
CO2EML(i, k) = 1.2 \times 10^{11} \frac{\exp(960/T_k)}{T_k^5 (\exp(960/T_k) - 1)^2}
\]

where \( T_k = TLAYR(i, k) \) from /CRDTMP/
**Variable Name** | **Set** | **Description**
--- | --- | ---
IMAX | RADINI | Longitude dimension (number of unique points); set to PLON from deck PARAMS
KMAX | RADINI | Number of Model layers; set to PLEV from deck PARAMS
KMAXP1 | RADINI | KMAX+1
C(200) | RADINI | Array of scalar constants used by the radiation code; only the following 9 elements are currently used:
C(54)=1.E2*GRAVIT; acceleration due to gravity in \( \text{cm} \cdot \text{s}^{-2} \); based on the SI units constant in /COMCON/
C(55)=1.E-2*RGA; inverse of acceleration due to gravity in \( \text{s}^2 \cdot \text{cm}^{-1} \); based on the SI units constant in /COMCON/
C(58)=1.E4*CPAIR; specific heat of air at constant pressure in \( \text{erg} \cdot \text{K}^{-1} \cdot \text{g}^{-1} \); based on the SI units constant in /COMCON/
C(75)=EPSILO; ratio of the molecular weights of water vapor and dry air; based on the SI units constant in /COMCON/
C(81)=1.013250E6; standard sea level pressure in bars
C(83)=1.E3*STEBOL; Stefan-Boltzmann constant in \( \text{erg} \cdot \text{cm}^{-2} \cdot \text{s}^{-1} \cdot \text{K}^{-4} \); based on the SI units constant in /COMCON/
C(85)=0.5/(C(54)*C(81))
C(123)=3.3E-4; carbon dioxide volume mixing ratio
C(124)=0.647; fraction of solar flux for wavelengths shorter than 0.9 microns
PLOL(JMX,KMXP) | RADOZ2 | Ozone path lengths (in the form needed for LW computations) with the dimensions used for zonally uniform ozone; when ozone is allowed to vary with longitude, the dimensions are PLOL(PLON,PLEVP)
<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set In</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLOS(JMX,KMXP)</td>
<td>RADOZ2</td>
<td>Ozone path lengths (in the form used primarily for SW computations) with the dimensions used for zonally uniform ozone; when ozone is allowed to vary with longitude, the dimensions are PLOS(PLON,PLEVP)</td>
</tr>
<tr>
<td>ZDIF(KMXP)</td>
<td>—</td>
<td>Not used</td>
</tr>
<tr>
<td>CSX(JMX)</td>
<td>RADINI</td>
<td>Cosine of latitude; used only in printout</td>
</tr>
<tr>
<td>SN(JMX)</td>
<td>RADINI</td>
<td>Sine of latitude; used only in printout</td>
</tr>
<tr>
<td>TTOP1(JMX)</td>
<td>RADINP</td>
<td>Vertically averaged temperature; used only in printout</td>
</tr>
<tr>
<td>CDTH(KMX)</td>
<td>RADINI</td>
<td>Cloud thickness; set to 1. for all layers</td>
</tr>
<tr>
<td>CDRH(KMX)</td>
<td>RADINI</td>
<td>Cloud relative humidity; set to 0.8 for all layers</td>
</tr>
<tr>
<td>SMP(KMX)</td>
<td>RADINI</td>
<td>Cloud multiple scattering factor; set to 10. for all layers</td>
</tr>
<tr>
<td>CABDR(KMX)</td>
<td>RADCSW</td>
<td>Cloud albedo for direct solar radiation</td>
</tr>
<tr>
<td>CABDF(KMX)</td>
<td>RADCSW</td>
<td>Cloud albedo for diffuse solar radiation</td>
</tr>
<tr>
<td>COVABS(KMX)</td>
<td>RADCSW</td>
<td>Absorption for overlapped clouds</td>
</tr>
<tr>
<td>ALBS(IMXP)</td>
<td>RADALB</td>
<td>Vegetation albedo for direct, visible solar radiation</td>
</tr>
<tr>
<td>ALBL(IMXP)</td>
<td>RADALB</td>
<td>Vegetation albedo for direct, near-IR solar radiation</td>
</tr>
<tr>
<td>ALBSD(IMXP)</td>
<td>RADALB</td>
<td>Vegetation albedo for diffuse, visible solar radiation</td>
</tr>
<tr>
<td>ALBLD(IMXP)</td>
<td>RADALB</td>
<td>Vegetation albedo for diffuse, near-IR solar radiation</td>
</tr>
<tr>
<td>ALBG(IMXP)</td>
<td>RADALB</td>
<td>Ground albedo for direct, visible solar radiation</td>
</tr>
<tr>
<td>ALBGL(IMXP)</td>
<td>RADALB</td>
<td>Ground albedo for direct, near-IR solar radiation</td>
</tr>
<tr>
<td>ALBGSD(IMXP)</td>
<td>RADALB</td>
<td>Ground albedo for diffuse, visible solar radiation</td>
</tr>
<tr>
<td>ALBGGLD(IMXP)</td>
<td>RADALB</td>
<td>Ground albedo for diffuse, near-IR solar radiation</td>
</tr>
<tr>
<td>ALBG(IMXP)</td>
<td>RADALB</td>
<td>Vegetation albedo for direct solar radiation (all wavelengths combined); not used in any computations</td>
</tr>
<tr>
<td>QFIX(KMXP)</td>
<td>RADINI</td>
<td>Minimum water vapor mixing ratio for each cloud level (values depend on vertical resolution); values for the standard 12-layer Model are given below, numbering levels from bottom to top:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>level</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 9</td>
<td>$1.5 \times 10^{-6}$</td>
</tr>
<tr>
<td>10</td>
<td>$2.0 \times 10^{-6}$</td>
</tr>
<tr>
<td>11</td>
<td>$2.5 \times 10^{-6}$</td>
</tr>
<tr>
<td>12</td>
<td>$3.0 \times 10^{-6}$</td>
</tr>
<tr>
<td>13</td>
<td>$3.5 \times 10^{-6}$</td>
</tr>
</tbody>
</table>
### Description

CALBR(KMXP) is a RADINI parameter used in the computation of cloud albedos for solar radiation (values depend on vertical resolution); values for the standard 12-layer Model are given below, numbering levels from bottom to top:

<table>
<thead>
<tr>
<th>level</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 4</td>
<td>0.6</td>
</tr>
<tr>
<td>5, 6</td>
<td>0.3</td>
</tr>
<tr>
<td>7 - 13</td>
<td>0.15</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable</th>
<th>Set</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCON</td>
<td>RADINI</td>
<td>SCON=1.37E6; mean solar constant in erg·cm⁻²·s⁻¹</td>
</tr>
<tr>
<td>DIFF</td>
<td>RADINI</td>
<td>DIFF=1.8; diffusivity factor for reflected solar radiation</td>
</tr>
</tbody>
</table>
$COMDECK CRDCTL
C
CL RADIATION CONTROL VARIABLES
C

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FRADSW</td>
<td>Logical flag; true iff full SW computations are being done for the current iteration</td>
</tr>
<tr>
<td>FRADLW</td>
<td>Logical flag; true iff full LW computations are being done for the current iteration</td>
</tr>
<tr>
<td>IRADSW</td>
<td>Frequency of full SW computations, in iterations</td>
</tr>
<tr>
<td>IRADLW</td>
<td>Frequency of full LW computations, in iterations</td>
</tr>
<tr>
<td>NACLW</td>
<td>Accumulation counter for averaged LW history tape fields</td>
</tr>
<tr>
<td>NACSW</td>
<td>Accumulation counter for averaged SW history tape fields</td>
</tr>
<tr>
<td>FNLW</td>
<td>Normalization factor for averaged LW history tape fields</td>
</tr>
<tr>
<td>FNSW</td>
<td>Normalization factor for averaged SW history tape fields</td>
</tr>
</tbody>
</table>

FRADSW and FRADLW are set in RADCTL; IRADSW and IRADLW are set to their default values in subroutine PRESET, then potentially modified by NAMELIST $NEWRUN in subroutine DATA. Due to implicit assumptions imbedded in various parts of the radiation computation code, IRADSW and IRADLW must be set to the same value. This is assured by a test in subroutine DATA at the time these parameters are read in to the Model. The accumulation counters NACLW and NACSW are initialized in subroutine LINEMS, incremented in RADOUT, and used in LINEMS to compute the normalization factors FNLW and FNSW. Averaged fields are normalized in WSHIST just before they are written.
### Arrays Used in LW Emissivity Computations

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>H2OEMS(IMX,KMXP)</td>
<td>RADEMS</td>
<td>Water vapor LW emissivity</td>
</tr>
<tr>
<td>TPATH(IMX)</td>
<td>RADEMS</td>
<td>Scratch array used only in RADEMS</td>
</tr>
<tr>
<td>TPLNKE(IMX)</td>
<td>RADTPL</td>
<td>Radiative temperature of the atmosphere above the top of the Model; set to highest full level temperature</td>
</tr>
</tbody>
</table>
| S2T(IMX,KMXP)          | RADTPL    | $S2T(i,k) = \begin{cases} 
\frac{1}{2g\rho_0} \sum_{l=k}^{KMAX} q_l T_l (p_l^2 - p_{l+\frac{1}{2}}^2) \\
+ S2T(i,KMAX + 1), & 1 \leq k \leq KMAX \\
\frac{1}{2g\rho_0} QFIX(KMAX + 1) T_{k-1} - 1 p_{k-\frac{1}{2}}^2, & k = KMAX + 1 
\end{cases}$ |
| UC(IMX)                | RADEMS    | Scratch array used independently in RADABS and RADEMS                       |
| S2C(IMX,KMXP)          | RADTPL    | $S2C(i,k) = \begin{cases} 
\frac{1}{2g\rho_0} \sum_{l=k}^{KMAX} [q_l^2 (p_l^2 - p_{l+\frac{1}{2}}^2) \\
\times \exp[1800(\frac{1}{1.4} - \frac{1}{2.96})]] \\
+ S2C(i,KMAX + 1), & 1 \leq k \leq KMAX \\
\frac{1}{2g\rho_0} [QFIX(KMAX + 1)]^2 p_{k-\frac{1}{2}}^2, & k = KMAX + 1 
\end{cases}$ |
| W(IMX,KMXP)            | RADTPL    | $W(i,k) = \begin{cases} 
\frac{1}{g\rho_0} \sum_{l=k}^{KMAX} q_l (p_l - p_{l+\frac{1}{2}}) \\
+ W(i,KMAX + 1), & 1 \leq k \leq KMAX \\
\frac{1}{g\rho_0} QFIX(KMAX + 1) p_{k-\frac{1}{2}}, & k = KMAX + 1 
\end{cases}$ |

Where

- $g = C(54)$ from /CRDCON/, acceleration due to gravity,
- $p_0 = C(81)$ from /CRDCON/, standard sea level pressure,
- $\epsilon = C(75)$ from /CRDCON/, ratio of molecular weights of water vapor and dry air,
- $q_k$ is water vapor mixing ratio at layer $k$ (full level),
- QFIX is the array of minimum mixing ratios from /CRDCON/,
- $T_k$ is temperature at layer $k$ (full level), and
- $p_{k+\frac{1}{2}}$ is pressure at the top interface (half level) of layer $k$. 

5-12
$COMDECK CRDFLX
C
CL RADIATIVE FLUXES
C

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set</th>
<th>In</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FS(KMXP)</td>
<td>RADCSW</td>
<td></td>
<td>Net direct downward clearsky SW flux; used only in RADCSW</td>
</tr>
<tr>
<td>FSU(KMXP)</td>
<td>RADCLW</td>
<td></td>
<td>Net upward clearsky LW flux; used only in RADCLW</td>
</tr>
<tr>
<td>FSD(KMXP)</td>
<td>RADCLW</td>
<td></td>
<td>Net downward clearsky LW or SW flux; used independently in RADCLW</td>
</tr>
<tr>
<td></td>
<td>RADCSW</td>
<td></td>
<td>in RADCLW and RADCSW</td>
</tr>
<tr>
<td>FU(KMXP)</td>
<td>RADCLW</td>
<td></td>
<td>Net upward LW or SW flux; used independently in RADCLW</td>
</tr>
<tr>
<td></td>
<td>RADCSW</td>
<td></td>
<td>in RADCLW and RADCSW</td>
</tr>
<tr>
<td>FD(KMXP)</td>
<td>RADCLW</td>
<td></td>
<td>Net downward LW or SW flux; used independently in RADCLW</td>
</tr>
<tr>
<td></td>
<td>RADCSW</td>
<td></td>
<td>and RADCSW</td>
</tr>
</tbody>
</table>
All arrays in this COMMON block are set in subroutine RADINI. In the following table, \( \sigma \) is the Model vertical coordinate and \( k \) is the level index, counting from bottom to top (integral for full levels, fractional for half levels). For the purpose of these definitions, \( \sigma_{\frac{1}{2}} = 1 \) and \( \sigma_{\text{KMAX}+rac{1}{2}} = 0.5\sigma_{\text{KMAX}} \).

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAF3(KMXP)</td>
<td>( DAF3(k) = \frac{1}{\sigma_{k-\frac{1}{2}} - \sigma_{k+\frac{1}{2}}} ); defined only for ( 1 \leq k \leq \text{KMAX} )</td>
</tr>
<tr>
<td>DAF4(KMXP)</td>
<td>( DAF4(k) = \frac{1}{\sigma_{k-\frac{1}{2}} - \sigma_{k+\frac{1}{2}}} ); defined only for ( 1 \leq k \leq \text{KMAX} )</td>
</tr>
<tr>
<td>UINPL(4,KMXP)</td>
<td>Constants for interpolation proportional to ((\text{pressure})^2). All definitions apply only for ( 1 \leq k \leq \text{KMAX} ); array values are undefined at ( k = \text{KMAX}+1 )</td>
</tr>
<tr>
<td></td>
<td>( \text{UINPL}(1,k) = DAF3(k) \times \left[ \frac{\sigma^2_{k-\frac{1}{2}} - \left( \frac{\sigma_{k+\frac{1}{2}}}{2} \right)^2}{2} \right] )</td>
</tr>
<tr>
<td></td>
<td>( \text{UINPL}(2,k) = DAF3(k) \times \left[ \left( \frac{\sigma_{k+\frac{1}{2}}}{2} \right)^2 - \sigma^2_{k+\frac{1}{2}} \right] )</td>
</tr>
<tr>
<td></td>
<td>( \text{UINPL}(3,k) = DAF3(k) \times \left[ \left( \frac{\sigma_{k+\frac{1}{2}}}{2} \right)^2 - \sigma^2_{k+\frac{1}{2}} \right] )</td>
</tr>
<tr>
<td></td>
<td>( \text{UINPL}(4,k) = DAF3(k) \times \left[ \frac{\sigma^2_{k-\frac{1}{2}} - \left( \frac{\sigma_{k+\frac{1}{2}}}{2} \right)^2}{2} \right] )</td>
</tr>
<tr>
<td>WINPL(4,KMXP)</td>
<td>Constants for interpolation proportional to pressure. All definitions apply only for ( 1 \leq k \leq \text{KMAX} ); array values are undefined at ( k = \text{KMAX}+1 )</td>
</tr>
<tr>
<td></td>
<td>( \text{WINPL}(1,k) = DAF4(k) \times \left( \frac{\sigma_{k-\frac{1}{2}} - \sigma_{k}}{2} \right) )</td>
</tr>
<tr>
<td></td>
<td>( \text{WINPL}(2,k) = DAF4(k) \times \left( \frac{\sigma_{k} - \sigma_{k+\frac{1}{2}}}{2} \right) )</td>
</tr>
<tr>
<td></td>
<td>( \text{WINPL}(3,k) = DAF4(k) \times \left[ \left( \frac{\sigma_{k+\frac{1}{2}}}{2} \right)^2 - \sigma_{k+\frac{1}{2}} \right] )</td>
</tr>
<tr>
<td></td>
<td>( \text{WINPL}(4,k) = DAF4(k) \times \left[ \sigma_{k-\frac{1}{2}} - \left( \frac{\sigma_{k+\frac{1}{2}} + \sigma_{k}}{2} \right) \right] )</td>
</tr>
<tr>
<td>Variable Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ZINPL(4,KMXP)</td>
<td>Constants for interpolation proportional to height. All definitions apply only for $1 \leq k \leq KMAX$; at $k = KMAX+1$, all array values are undefined. [ ZINPL(1,k) = \frac{1}{2} \left( \ln \frac{\sigma_{k+\frac{1}{2}}}{\sigma_k} \right) / \left( \ln \frac{\sigma_{k-\frac{1}{2}}}{\sigma_{k+\frac{1}{2}}} \right) ] [ ZINPL(2,k) = \frac{1}{2} \left( \ln \frac{\sigma_{k+\frac{1}{2}}}{\sigma_k} \right) / \left( \ln \frac{\sigma_{k-\frac{1}{2}}}{\sigma_{k+\frac{1}{2}}} \right) ] [ ZINPL(3,k) = ZINPL(1,k) + 2.*ZINPL(2,k) ] [ ZINPL(4,k) = ZINPL(2,k) + 2.*ZINPL(1,k) ]</td>
</tr>
<tr>
<td>PINPL(4,KMXP)</td>
<td>Mean sigma for the half layer for pressure broadening for ozone (method B for mean pressure interpolation). All definitions apply only for $1 \leq k \leq KMAX$; array values are undefined at $k = KMAX+1$. [ PINPL(1,k) = \frac{1}{2} \left( \frac{\sigma_k + \sigma_{k-\frac{1}{2}}}{2} + \sigma_{k-\frac{1}{2}} \right) ] [ PINPL(2,k) = \frac{1}{2} \left( \frac{\sigma_k + \sigma_{k+\frac{1}{2}}}{2} + \sigma_{k+\frac{1}{2}} \right) ] [ PINPL(3,k) = \frac{1}{2} \left( \frac{\sigma_k + \sigma_{k-\frac{1}{2}}}{2} + \sigma_{k-\frac{1}{2}} \right) ] [ PINPL(4,k) = \frac{1}{2} \left( \frac{\sigma_k + \sigma_{k+\frac{1}{2}}}{2} + \sigma_{k+\frac{1}{2}} \right) ]</td>
</tr>
<tr>
<td>DY(KMXP)</td>
<td>[ DY(1) = 0 ] [ DY(KMAX+1) = \left( \ln \frac{\sigma_{KMAX+\frac{1}{2}}}{\sigma_{KMAX}} \right) / \left( \ln \frac{\sigma_{KMAX}}{\sigma_{KMAX-\frac{1}{2}}} \right) ] [ DY(k) = \left( \ln \frac{\sigma_{k-\frac{1}{2}}}{\sigma_{k-\frac{1}{2}}} \right) / \left( \ln \frac{\sigma_{k}}{\sigma_{k-1}} \right); \text{ for } 2 \leq k \leq KMAX ]</td>
</tr>
</tbody>
</table>
Contains the current latitude index.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Variable Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>J</td>
<td>Integer</td>
<td>Current latitude index, counting from north to south. Set in subroutine RADINP.</td>
</tr>
</tbody>
</table>
INDICES FOR RADIATION F ARRAY

All of the variables in this COMMON block are pointers in the F array workspace COMMON block /WORKSP/ (Update Common Deck CRDWRK). When used as the second subscript in F, each pointer locates the first (lowest) level in the longitude-height slice for a particular field, at the current latitude. The descriptions in the following table identify the field associated with the pointer. All of these fields are either input or output (with respect to the radiation parameterization); additional details are given in the “Users’ Guide to NCAR CCM1” (Bath, et al., 1987). The values of the pointers are given for the standard 12-layer Model; they are independent of horizontal resolution.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Pointer Value</th>
<th>No. Levels</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCLD</td>
<td>13</td>
<td>NLEV+1</td>
<td>Cloud fraction at layer interfaces (half-levels), starting at the surface; multiplied by cloud emissivity in RADCLW</td>
</tr>
<tr>
<td>MTNM</td>
<td>1</td>
<td>NLEV</td>
<td>Temperature at layer midpoints (full levels)</td>
</tr>
<tr>
<td>MQNM</td>
<td>26</td>
<td>NLEV</td>
<td>Water vapor mixing ratio</td>
</tr>
<tr>
<td>MPNM</td>
<td>38</td>
<td>NLEV+1</td>
<td>Pressure at layer interfaces (half-levels)</td>
</tr>
<tr>
<td>MPBR</td>
<td>51</td>
<td>NLEV</td>
<td>Pressure at layer midpoints (full levels); not currently used</td>
</tr>
<tr>
<td>MVEG</td>
<td>63</td>
<td>1</td>
<td>Fractional vegetation cover used for ground albedo computation; currently set to zero everywhere</td>
</tr>
<tr>
<td>MFRS</td>
<td>64</td>
<td>1</td>
<td>Downward SW flux at the surface</td>
</tr>
<tr>
<td>MQRS</td>
<td>65</td>
<td>NLEV</td>
<td>Net SW heating rate for each layer; converted (within the F array) from a net flux into each layer at the end of RADCSW</td>
</tr>
<tr>
<td>MQRL</td>
<td>77</td>
<td>NLEV</td>
<td>Net LW heating rate for each layer</td>
</tr>
<tr>
<td>MTG</td>
<td>89</td>
<td>1</td>
<td>Surface (ground) temperature</td>
</tr>
<tr>
<td>MTS</td>
<td>1</td>
<td>1</td>
<td>Lowest free atmosphere temperature</td>
</tr>
<tr>
<td>MABSB</td>
<td>90</td>
<td>NLEV+1</td>
<td>Total LW absorptivity (H₂O, CO₂, O₃) between the lower interface of each layer and the surface</td>
</tr>
<tr>
<td>MABSBT</td>
<td>103</td>
<td>1</td>
<td>Boundary term used to compute the downward LW flux at the surface</td>
</tr>
<tr>
<td>MABSB1</td>
<td>104</td>
<td>1</td>
<td>Total LW absorptivity (H₂O, CO₂, O₃) for the lowest 1/4 of the lowest layer</td>
</tr>
</tbody>
</table>

5-17
<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Pointer Value</th>
<th>No. Levels</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MABSB2</td>
<td>105</td>
<td>1</td>
<td>Total LW absorptivity (H₂O, CO₂, O₃) for the lowest 3/4 of the lowest layer</td>
</tr>
<tr>
<td>MFRL</td>
<td>106</td>
<td>1</td>
<td>Net upward LW flux at the surface</td>
</tr>
<tr>
<td>MSABTP</td>
<td>107</td>
<td>1</td>
<td>Solar flux absorbed by all Model layers plus the surface</td>
</tr>
<tr>
<td>MALBT</td>
<td>108</td>
<td>1</td>
<td>SW flux absorbed by the atmosphere above the top of the Model</td>
</tr>
<tr>
<td>MFIRTP</td>
<td>109</td>
<td>1</td>
<td>Net upward LW flux at the top of the Model</td>
</tr>
<tr>
<td>MOCEAN</td>
<td>110</td>
<td>1</td>
<td>Flag set to 1. for open ocean gridpoints, 0. otherwise</td>
</tr>
<tr>
<td>MSICE</td>
<td>111</td>
<td>1</td>
<td>Flag set to 1. for sea ice gridpoints, 0. otherwise</td>
</tr>
<tr>
<td>MSCV</td>
<td>112</td>
<td>1</td>
<td>Flag set to 1. for gridpoints with snowcover, -1. otherwise</td>
</tr>
<tr>
<td>MCLRLT</td>
<td>113</td>
<td>1</td>
<td>Net upward clearsky LW flux at the top of the atmosphere</td>
</tr>
<tr>
<td>MCLRST</td>
<td>114</td>
<td>1</td>
<td>Net downward clearsky SW flux at the top of the atmosphere</td>
</tr>
<tr>
<td>MCLRLS</td>
<td>115</td>
<td>1</td>
<td>Net upward clearsky LW flux at the surface</td>
</tr>
<tr>
<td>MCLRSS</td>
<td>116</td>
<td>1</td>
<td>Net downward clearsky SW flux at the surface</td>
</tr>
<tr>
<td>MSLWD</td>
<td>117</td>
<td>1</td>
<td>Total (LW + SW) net downward radiative flux at the surface</td>
</tr>
<tr>
<td>MSOLIN</td>
<td>118</td>
<td>1</td>
<td>Downward SW flux (solar input) at the top of the atmosphere</td>
</tr>
</tbody>
</table>
PRESSURES USED FOR OZONE COMPUTATIONS

IDEALIZED SIGMA LEVEL PRESSURES BASED ON SURFACE

PRESSURE OF 1.E5 PASCALS WHEN UPDATE CONDITIONAL OZLONG IS UNDEFINED

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRSI(PLEVP)</td>
<td>Pressure (in Pascals) of half sigma levels (layer interfaces) used only for interpolating input ozone mixing ratios to Model levels</td>
</tr>
</tbody>
</table>

When the option making ozone a function of longitude is used, the dimensions become PRSI(PLON,PLEVP), and the Model surface pressure is used (instead of the constant 1.E5) to compute PRSI.
$COMDECK CRDPTH
C CL VERTICAL PATH LENGTHS FOR WATER VAPOR AND CARBON CL DIOXIDE, PLUS CLEAR SKY FRACTION C

All three arrays in this COMMON block are initially set in subroutine RADPTH. The units of PLC02 are changed in RADTPL, and TCLRSF is reset in RADCLW to incorporate cloud emissivity.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLH20(IMXP,KMXP)</td>
<td>Water vapor path lengths between each half level and the top of the atmosphere;</td>
</tr>
</tbody>
</table>
|                        | \[ \begin{array}{l}
|                        | \frac{1}{2gpo} \sum_{i=k}^{KMAX} q_i (p_{i-\frac{1}{2}} - p_{i+\frac{1}{2}}) \\
|                        | + PLH20(i,KMAX+1), \quad 1 \leq k \leq KMAX \\
|                        | \frac{1}{2gpo} QFIX(KMAX+1)p_{KMAX+\frac{1}{2}}^2, \quad k = KMAX+1 \end{array} \] |
| PLC02(IMXP,KMXP)       | Carbon dioxide path lengths between each half level and the top of the atmosphere; the following expression is used for the computation in RADPTH and results in path lengths with units of cm-atm, as used by the SW computations; all values are multiplied by 1.9642 \times 10^{-3} in RADTPL, changing the units to g cm\(^{-2}\) for the LW computations; |
|                        | \[ PLC02(i,k) = \frac{RT_o}{2gpo} \omega_i p_{i-\frac{1}{2}}^2, \quad 1 \leq k \leq KMAX+1 \] |
|                        | where \( \frac{RT_o}{2gpo} = 3.8917273 \times 10^{-7} \) |
| TCLRSF(IMXP,KMXP)      | Product of all clear sky fractions for the layers above the current layer; reset in RADCLW after clouds are multiplied by cloud emissivity; |
|                        | \[ TCLRSF(i,k) = \begin{array}{l}
|                        | \prod_{l=k}^{KMAX} (1 - C_k), \quad 1 \leq k \leq KMAX \\
|                        | 1, \quad k = KMAX+1 \end{array} \] |

where

- \( g = C(54) \) from /CRDCON/, acceleration due to gravity,
- \( p_0 = C(81) \) from /CRDCON/, standard sea level pressure,
- \( q_k \) is water vapor mixing ratio at layer \( k \) (full level),
- \( \omega_i = C(123) \) from /CRDCON/, \( \text{CO}_2 \) volume mixing ratio,
- \( Q\text{FIX} \) is the array of minimum mixing ratios from /CRDCON/, \( P_{k+\frac{1}{2}} \) is pressure at the top interface (half level) of layer \( k \), and \( C_k \) is the cloud fraction for layer \( k \).
$COMDECK CRDRDT
C
CL SCRATCH ARRAYS USED IN LW COMPUTATIONS
C

Contains scratch arrays used locally by subroutines RADABS and/or RADEMS. Also contains array TROC02(IMX,KMXP), water vapor overlap factor computed in RADEMS and used in RADC02.
SCRATCH ARRAY USED IN LW COMPUTATIONS

Contains surface pressure array set in RADCLW and referenced in RADABS.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Variable Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TARRAY(PLON)</td>
<td>Real</td>
<td>Surface pressure</td>
</tr>
</tbody>
</table>
Contains solar radiation parameters used in computing solar zenith angle.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Variable Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JD</td>
<td>Integer</td>
<td>Integral part of current julian day</td>
</tr>
<tr>
<td>FJD</td>
<td>Real</td>
<td>Fractional part of current julian day</td>
</tr>
<tr>
<td>FCSTDA</td>
<td>Real</td>
<td>Julian day and fraction of day</td>
</tr>
<tr>
<td>R1</td>
<td>Real</td>
<td>Radius of earth's orbit, normalized by semi-major axis</td>
</tr>
<tr>
<td>DLT</td>
<td>Real</td>
<td>Solar declination</td>
</tr>
<tr>
<td>SLAG</td>
<td>Real</td>
<td>Not used</td>
</tr>
</tbody>
</table>

The following variable is set by subroutine CALDYI

The following are set by subroutine RADSLR, called from RADCTL, and are used by RADZEN to compute solar zenith angle.
DATE-RELATED SOLAR RADIATION PARAMETERS

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set In</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAZ(12)</td>
<td>CALDYI</td>
<td>Day of year for each month of year</td>
</tr>
<tr>
<td>TAUDAM(JG2)</td>
<td>RADCTL</td>
<td>Solar parameter set via a CALL to RADSLR; not used except in printout</td>
</tr>
<tr>
<td>COSZM(JG2)</td>
<td>RADCTL</td>
<td>Solar parameter set via a CALL to RADSLR; not used except in printout</td>
</tr>
<tr>
<td>HA(JG2)</td>
<td>RADCTL</td>
<td>Solar parameter set via a CALL to RADSLR; used to compute zenith angle in RADZEN</td>
</tr>
<tr>
<td>COSZRS(JF)</td>
<td>RADINP</td>
<td>Effective mean cosine of zenith angle; set via a CALL to RADZEN</td>
</tr>
<tr>
<td>FRAC(JF)</td>
<td>RADINP</td>
<td>Daylight fraction; set via a CALL to RADZEN</td>
</tr>
<tr>
<td>ALAT(JG2)</td>
<td>RADINI</td>
<td>Model latitudes in radians, north to south</td>
</tr>
</tbody>
</table>
All of the arrays in this COMMON block are set in subroutine RAD03E.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBVTIT(IMX,KMXP)</td>
<td>Statement function DBVT(T) (described below) evaluated at layer interface</td>
</tr>
<tr>
<td></td>
<td>(half level) temperatures (array TINT from /CRDTMP/)</td>
</tr>
<tr>
<td>DBVILY(IMX,KMX)</td>
<td>Statement function DBVT(T) (described below) evaluated at layer midpoint</td>
</tr>
<tr>
<td></td>
<td>(full level) temperatures (array TLAYR from /CRDTMP/)</td>
</tr>
<tr>
<td>DBVTT(IMX)</td>
<td>Statement function DBVT(T) (described below) evaluated at the temperature</td>
</tr>
<tr>
<td></td>
<td>of the isothermal layer above the top of the Model (array TPLNKE from</td>
</tr>
<tr>
<td></td>
<td>/CRDEMS/)</td>
</tr>
<tr>
<td>H20TR(IMX,KMXP)</td>
<td>Water vapor transmittance in the ozone absorption band</td>
</tr>
</tbody>
</table>

\[
H20TR(i, k) = \exp (-12 \cdot S2C(i, k))
\]

where \(S2C\) is the array from /CRDEMS/.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>03EMS(IMX,KMXP)</td>
<td>Ozone emissivity, computed as follows:</td>
</tr>
</tbody>
</table>

\[
03EMS(i, k) = DBVT(i) \cdot H20TR(i, k) \cdot 03BNDI \cdot \frac{TPLNKE(i)}{375}
\]

where TPLNKE is the array from /CRDEMS/, and

\[
03BNDI = 74T_e \ln \left( 1 + \frac{u_1}{\sqrt{4 + u_1(1 + \nu)}} + \frac{u_2}{\sqrt{4 + u_2(1 + \nu)}} \right)
\]

where

\[
T_e = (C02T(i, k)/293)^7
\]

\[
u_1 = 18.29 \cdot PLOS/T_e
\]

\[
u_2 = 0.5649 \cdot PLOS/T_e
\]

\[
u = 0.3205T_e \cdot PLOS/PLOL,\] where

\(C02T\) is the array from /CRDTMP/, and \(PLOS\) and \(PLOL\) are the appropriate elements of the arrays from /CRDCON/.
The statement function $DBVT(T)$ is used to evaluate both the Planck function and its derivative in the ozone absorption band. Analytically, the function is defined as

$$DBVT(T) = \frac{2.8101 \times 10^{11} \exp\left(\frac{-1500.5}{T}\right)}{T^5}$$

but it is approximated with the rational polynomial

$$DBVT(T) = \frac{1.131 \times 10^{-10} T^2 + 2.377 \times 10^{-6} T - 2.891 \times 10^{-4}}{1.555 \times 10^{-5} T^2 - 6.136 \times 10^{-3} T + 1}.$$

The code in RAD03E specifies the polynomial coefficients to 11 decimal digits. The polynomial was fit to the interval 160K to 360K with a maximum relative error of 0.21%.
All of the arrays in this COMMON block are set in subroutine RADTPL with the exception of C02T, which is set in RADCO2.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| TINT(IMX,KMXP) | \[ T_{g} = F(i,MTG), \]
| | \[ T_{k-1} - DY(k) \times (T_{k-1} - T_{k}), \quad 2 \leq k \leq \text{KMAX} \]
| | \[ T_{\text{KMAX}}, \quad k = \text{KMAX} + 1 \]
| TINT4(IMX,KMXP) | \\
| TLAYR4(IMX,KMXP) | \[ TLAYR4(i,k) = (TLAYR(i,k))^4 \]
| | \[ 1 \leq k \leq \text{KMAX} + 1 \]
| TINT4(i,k) | \[ (TINT(i,k))^4 \]
| | \[ 1 \leq k \leq \text{KMAX} + 1 \]
| TLAYR(IMX,KMXP) | \[ TLAYR(i,k) = \begin{cases} T_{k}, & 1 \leq k \leq \text{KMAX} \\ TINT(i,k), & k = \text{KMAX} + 1 \end{cases} \]
| C02T(IMX,KMXP) | \[ C02T(i,k) = \begin{cases} TPLNKE(i) + \sum_{l=k}^{\text{KMAX}} TLAYR(i,l) \frac{P_{l+\frac{1}{2}} - P_{l-\frac{1}{2}}}{P_{l-\frac{1}{2}}}, & 1 \leq k \leq \text{KMAX} \\ TPLNKE(i), & k = \text{KMAX} + 1 \end{cases} \]

where

- \( T_{k} \) is temperature at layer \( k \) (full level), and
- \( P_{k+\frac{1}{2}} \) is pressure at the top interface (half level) of layer \( k \).
$COMDECK CRDTRN
C CL FARWING CORRECTION CONSTANTS FOR NARROW-BAND EMISSIVITY
CL MODEL
C INTRODUCE FARWING CORRECTION TO ACCOUNT FOR THE
C DEFICIENCIES IN NARROW-BAND MODEL USED TO DERIVE THE
C EMISSIVITY. TUNED WITH ARKINGS LINE CALCULATIONS.
C
The four variables in this COMMON block are scalars. They are set in subroutine
RADEMS to the following values:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FWCOEF</td>
<td>0.1</td>
</tr>
<tr>
<td>FWC1</td>
<td>0.3</td>
</tr>
<tr>
<td>FWC2</td>
<td>4.5</td>
</tr>
<tr>
<td>FC1</td>
<td>2.6</td>
</tr>
</tbody>
</table>
$COMDECK CRDWRK
C
CL F ARRAY FOR RADIATION PACKAGE
C

Contains the “F” array used as the main data buffer by the radiation parameterization. This COMMON block is equivalenced to array WORK in Common Deck WORKSP, as is the COMMON block CLDWRK for the cloud parameterization. Note that the array WORK, which is equivalenced to array F and actually determines the length of this COMMON block, is set via the PARAMETER LENWRK to be the length of the F array, LENWRK=PLON*(8*PLEV+22)), as this is the longer of the two spaces required by the radiation and cloud packages.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Variable Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F(PLON,8*PLEV+22)</td>
<td>Real</td>
<td>Work space equivalenced to array WORK in Common Deck WORKSP. Used as the main data buffer for storing Model data fields as input and output to the radiation package</td>
</tr>
</tbody>
</table>
SECTION 6

Column Model Structure

Since the Column Model is based on CCM1 Radiation Code, its program structure is similar to the CCM1. In particular, a single call to the Column Model code would compute radiative quantities over a number of longitude points along a single latitude band. Each longitude point has several levels in the vertical (see Figure 1.1). Thus, the Column Model program structure requires specifying all input data for all levels and longitude points for each computation (or latitude band).

Communication between the user's code and the Column Model code is achieved through common blocks. The user's task then becomes to (1) transfer the appropriate input data, transforming units as necessary, into the input variables within specified common blocks, (2) calling the radiation routines, and (3) transferring the desired radiation quantities from the appropriate output common block into the user's code.

Column Model Input

The Column Model code consists of Fortran code and common decks (groups of common/and or parameter definitions to be inserted within the Fortran code). Within the Fortran code are commands (starting in column 1) such as "$CALL CLMSIN". In this example, the common deck "CLMSIN" is to be inserted within the Fortran code at that point before compilation.

There are four common decks: 'CLMSIN' (Column Model Storage for Input); 'CLMSOUT' (Column Model Storage for Output); 'GASES' (Allows turning off the appropriate gas if desired); and 'CLMDRS' (Column Model Data Resolution parameters).

The last common deck mentioned contains three resolution parameters, the number of longitudes 'nlon', the number of latitudes 'nlat', and the number of levels 'nlev'. (Note that the number of latitudes is not actually used in the physics of the computation; it is necessary only to dimension diagnostic arrays containing latitudes and other CCM internal information). The number of longitudes and levels must be set at this point to be consistent with those used in the user's model.

The common deck 'CLMSIN' contains input data for the Column Model. The next two pages contain tabular descriptions of the input data. These data must be set in the routine 'DATAIN' as described in the comment cards in that routine.
<table>
<thead>
<tr>
<th>Quantity</th>
<th>Description</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_{sp}$</td>
<td>level scaled pressure $= p/p_s$</td>
<td>nlev</td>
</tr>
<tr>
<td>$\theta$</td>
<td>latitude (°); +90=North Pole</td>
<td>nlat</td>
</tr>
<tr>
<td>$S_0$</td>
<td>solar constant $(\text{Wm}^{-2})$</td>
<td>1</td>
</tr>
<tr>
<td>$d_s$</td>
<td>earth-sun distance factor (unitless)</td>
<td>1</td>
</tr>
<tr>
<td>$\zeta$</td>
<td>solar zenith angle $(0^\circ \text{ to } 90^\circ)$</td>
<td>1</td>
</tr>
<tr>
<td>$f_d$</td>
<td>fraction of solar day (unitless)</td>
<td>1</td>
</tr>
<tr>
<td>$P_s^3$</td>
<td>surface pressure (mb)</td>
<td>1</td>
</tr>
<tr>
<td>$T$</td>
<td>temperature (K)</td>
<td>nlev</td>
</tr>
<tr>
<td>$T_s^4$</td>
<td>surface (anemometer) air temperature (K)</td>
<td>1</td>
</tr>
<tr>
<td>H2OMMR</td>
<td>H$_2$O mass mixing ratio (g/g)</td>
<td>nlev + 1</td>
</tr>
<tr>
<td>O3MMR</td>
<td>Ozone mass mixing ratio (g/g)</td>
<td>nlev + 1</td>
</tr>
<tr>
<td>CO2VMR</td>
<td>Carbon Dioxide volume mixing ratio (ppmv)</td>
<td>1</td>
</tr>
<tr>
<td>CCV</td>
<td>Fractional Cloud Cover (unitless)</td>
<td>nlev</td>
</tr>
<tr>
<td>CLW</td>
<td>Cloud Level Liquid Water $(\text{g cm}^{-3})$</td>
<td>nlev</td>
</tr>
</tbody>
</table>

1nlev is the number of levels; quantities with this number of levels refer to *level* quantities (Figure 1.1); nlev+1 refers to *interface* quantities, while 1 implies either a surface, top-of-atmosphere, or uniformly mixed quantity. The only exceptions to this condition are H2OMMR and O3MMR, the water vapor and ozone mass mixing ratios respectively, which appear to be interface quantities (nlev+1), but which are actually level quantities with an extra top layer value for the upper boundary condition needed in the radiative transfer calculations.

2$S_0 \times d_s \times f_d \times \cos(\zeta)$ is the actual solar irradiance incident on the model atmosphere at zenith angle $\zeta$.

3Actual pressures at each level can be found from $p_s \times \sigma_{sp}$, where $\sigma_{sp}$ is the scaled pressure.

4Temperature of the atmosphere in contact with the earth’s surface, independent of the precise type (ocean, land, ice) of surface. This temperature can be considered the ‘skin’ temperature of the atmosphere, or the anemometer height temperature. It is not necessarily equal to the ground temperature, which is the temperature of the physical surface in contact with the atmosphere.
<table>
<thead>
<tr>
<th>Quantity</th>
<th>Description</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Land Flag 1</td>
<td>Surface Type (land), unitless</td>
<td>1</td>
</tr>
<tr>
<td>Ocean Flag 2</td>
<td>Surface Type (ocean), unitless</td>
<td>1</td>
</tr>
<tr>
<td>Sea Ice Flag 2</td>
<td>Surface Type (ice), unitless</td>
<td>1</td>
</tr>
<tr>
<td>$T_g^3$</td>
<td>ground temperature (K)</td>
<td>1</td>
</tr>
<tr>
<td>Snowcv 4</td>
<td>Snow cover amount (frc)</td>
<td>1</td>
</tr>
<tr>
<td>$a_s^{\text{(vis)}}$</td>
<td>Surf albedo for vis and strong zen/ang dep (frc)</td>
<td>1</td>
</tr>
<tr>
<td>$a_s^{\text{(nir)}}$</td>
<td>Surf albedo for nir and strong zen/ang dep (frc)</td>
<td>1</td>
</tr>
<tr>
<td>$a_w^{\text{(vis)}}$</td>
<td>Surf albedo for vis and weak zen/ang dep (frc)</td>
<td>1</td>
</tr>
<tr>
<td>$a_w^{\text{(nir)}}$</td>
<td>Surf albedo for nir and weak zen/ang dep (frc)</td>
<td>1</td>
</tr>
</tbody>
</table>

1 For land surfaces (as determined by ‘Ocean’), if the surface is not covered with permanent ice (such as the Greenland Icecap), yields the fraction of the surface that has strong zenith angle dependence for albedo to direct solar radiation.

2 Surface flags from CCM1. ‘Ocean’ controls whether ocean water is present; ‘Sea Ice’ specifies whether the ocean is covered with sea ice.

3 This is the temperature of the physical surface in contact with the atmosphere; it is the radiating emission temperature. If the physical surface is ocean, this would be the temperature of the top meter or so of water. If land or ice, this would be the temperature of the top few centimeters of surface. This temperature is not necessarily equal to the temperature of the air in contact with the surface.

4 For the current CCM1 code, this quantity is essentially an on-off flag; i.e. either there is snow cover present or not.

5 vis (visible) refers to the wavelength interval of $0.2 - 0.9\mu\text{m}$, and nir (near-infrared) refers to the wavelength interval of $0.9 - 4.0\mu\text{m}$. These albedos have either strong or weak zenith angle dependence for the direct solar beam.
Column Model Output

The following table describes the radiative quantities output by the Column Model.

**Column Model Output: Computed Radiation**

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Description</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>QRS</td>
<td>solar heating rate (K/day)</td>
<td>nlev</td>
</tr>
<tr>
<td>QRL</td>
<td>longwave cooling rate (K/day)</td>
<td>nlev</td>
</tr>
<tr>
<td>SABTP</td>
<td>column solar absorbed flux (Wm(^{-2}))</td>
<td>1</td>
</tr>
<tr>
<td>FRS</td>
<td>surface absorbed flux (Wm(^{-2}))</td>
<td>1</td>
</tr>
<tr>
<td>FLWUP</td>
<td>longwave up flux (Wm(^{-2}))</td>
<td>nlev+1</td>
</tr>
<tr>
<td>FLWDN</td>
<td>longwave down flux (Wm(^{-2}))</td>
<td>nlev+1</td>
</tr>
<tr>
<td>FSWCLT</td>
<td>clear-sky absorbed shortwave flux (Wm(^{-2}))</td>
<td>1</td>
</tr>
<tr>
<td>FSWCLS</td>
<td>clear-sky surface abs shortwave flux (Wm(^{-2}))</td>
<td>1</td>
</tr>
<tr>
<td>FLWCLT</td>
<td>clear-sky outgoing longwave flux (Wm(^{-2}))</td>
<td>1</td>
</tr>
<tr>
<td>FLWCLS</td>
<td>clear-sky net surface longwave flux (Wm(^{-2}))</td>
<td>1</td>
</tr>
<tr>
<td>ALBTOA</td>
<td>shortwave top-of-atmosphere albedo (frc)</td>
<td>1</td>
</tr>
</tbody>
</table>
REFERENCES


