AD-A008 035

A HIGH ALTITUDE INFRARED RADIANCE MODEL

Thomas C. Degges

Visidyne, Incorporated

Prepared for:

Air Force Cambridge Research Laboratories
Defense Advanced Research Projects

27 December 1974

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFCRL-TR-74-0606	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER AD-A008 035
4. TITLE (and Subtitle) A HIGH ALTITUDE INFRARED RADIANCE MODEL		5. TYPE OF REPORT & PERIOD COVERED Final, 1 June 1972 through 30 April 1974 6. PERFORMING ORG. REPORT NUMBER VI-236
7. AUTHOR(s) Thomas C. Degges		8. CONTRACT OR GRANT NUMBER(s) F19628-72-C-0330
9. PERFORMING ORGANIZATION NAME AND ADDRESS Visidyne, Inc. 19 Third Avenue, N. W. Industrial Park Burlington, Massachusetts 01803		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 8692 N/A N/A
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Cambridge Research Laboratories Hanscom AFB, Massachusetts 01731 Contract Monitor: Bertram D. Schurin/OPI		12. REPORT DATE December 27, 1974 13. NUMBER OF PAGES 346
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Reproduced by
NATIONAL TECHNICAL INFORMATION SERVICE
US Department of Commerce
Springfield, VA. 22151

PRICES SUBJECT TO CHANGE

18. SUPPLEMENTARY NOTES

This research was sponsored by Defense Advanced Research Projects Agency. ARPA Order No. 1366

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Atmosphere, Infrared, Radiance, Model

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A knowledge of the natural infrared radiance originating in the earth's upper atmosphere is of interest for systems design, military surveillance and the advancement of knowledge about physical processes in the upper atmosphere. A physical model that includes experimental data on and theoretical estimates of excitation processes that lead to emission of infrared radiation has been implemented in a

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. computer program that computes infrared radiances for an earth limb viewing geometry. The nominal spectral region of this study lies between 2.7 and 25 micrometers and emphasis is placed on radiation originating at altitudes between 70 and 500 km. The physical model is described, with emphasis on the changes required in extending its usefulness. Application of the computer program is described and estimates are given of uncertainties in results due to assumptions made in the model and lack of data on actual atmospheric composition.

TABLE OF CONTENTS

SECTION		Page No.
I	Introduction	1
II	Radiative Processes and Modelling	5
III	Atmospheric Composition	39
IV	Results from Computational Model	55
V	High Altitude Radiance Programs	77
	References	261
APPENDIX	Chemistry Program SNAPS2	273

SECTION I

INTRODUCTION

The research reported here has had as its major objective the development of a computer program to simulate the natural infrared radiance background of the earth's upper atmosphere. The nominal spectral region under study lies between 2.7 and 25 micrometers and emphasis is placed on radiation originating at altitudes between 70 and 500 kilometers. The general problem area is of interest for systems design, military surveillance and the advancement of knowledge about physical processes in the upper atmosphere. The immediate application of this work will be to aid in developing optimum infrared background measurements programs and in interpreting the results of such measurements.

This work is an extension of the study of Corbin, et al. (1969) and Degges (1972). The former investigated the natural infrared background of the earth in the 5 to 25 micrometer spectral region, with the goal of estimating earth limb viewing radiances for tangent heights from the surface to 500 km altitude. For convenience, their study divided the atmosphere into two regions with a division at 70 km. Below 70 km the atmosphere was assumed to be in thermal equilibrium. Above 70 km explicit calculations were made of processes which excite and de-excite molecular vibrational and rotational levels which are the source of infrared radiation. Their study concentrated on radiation from water vapor, carbon dioxide, ozone, nitric oxide and nitrous oxide, which are the principal radiating species in the

spectral region considered. In addition, nitric acid was included in the lower atmosphere work and estimates were made of radiation to be expected from particulate matter suspended in the atmosphere.

Corbin, et al. (1969) presented models for the lower atmosphere for a wide range of seasonal and latitudinal conditions. This was not possible for the abundances of most minor neutral species. More data has since become available, particularly for nitric oxide and the hydroxyl radical, but at present it appears that the best means of estimating abundances of important infrared emitting species is chemical rate equation integrations including molecular diffusion and eddy mixing. Even calculations involving transport properties cannot always be accepted because the values of eddy mixing coefficients are to a large extent only informed guesses and published calculations often use outdated rate coefficients. Degges (1972) reported a computer program with which to determine diurnal variations in abundances of minor species, to investigate the effects of changes in assumed eddy mixing coefficients, and to estimate the effects of new determinations of chemical rate coefficients.

A second area of study involved in improving the radiance model includes the physical processes that control the population of infrared emitting states of atmospheric molecules. Except for the pure rotational radiation from molecules such as water, the degree of excitation of vibrational levels determines the radiation from infrared emitting molecules. The most important mechanisms are collisional excitation and de-excitation and absorption and re-emission of electromagnetic radiation.

In the troposphere and lower stratosphere, collisional processes are rapid enough to control the population of vibrational levels.

Above 30 to 50 km, however, collisional excitation becomes less efficient and radiative processes become important. The combined effects of collisional and radiative processes must therefore be considered. Below an altitude of about 90 km molecular nitrogen and oxygen are the most important collision partners. Above that altitude, atomic oxygen becomes important, both in exciting nitric oxide and in determining molecular oxygen and nitrogen vibrational temperatures.

In determining the effects of radiation on the populations of vibrational levels of infrared active molecules, it is necessary to separate the radiation of a single change in vibrational quantum numbers from the rest of the radiation field. The previously reported studies did this by assuming a Doppler line shape for the individual rotational lines of a band and were able to obtain adequate numerical approximations for radiative transfer functions appropriate to single bands of linear molecules, and less accurately, for water vapor and ozone bands lying in the spectral region of interest.

The principal areas of research and program revision reported here include a further study of band radiance modeling and the effects of line shape on radiance computations. In addition to some exploratory calculations using a Voigt line profile instead of a Doppler line shape, some asymptotic analytical results have been obtained which provide an estimate of the accuracy of the numerically obtained band functions at large optical thicknesses. It is found that when a Voigt profile is used, the connection between the band models and functions for a single line is more readily seen than when a Doppler line shape is used. This material, and a discussion of changes made in the numerical methods is described in Section II.

The status of current knowledge of atmospheric composition is reviewed in Section III. Recent measurements of chemical rate coefficients have resulted in changes in estimates of abundances of some molecular species. An attempt is made to set upper and lower bounds for mixing ratios at high altitudes.

Results of calculations are given in Section IV, and estimates are made of the validity of the results.

The computer programs are described in Section V.

SECTION II

RADIATIVE PROCESSES AND MODELLING

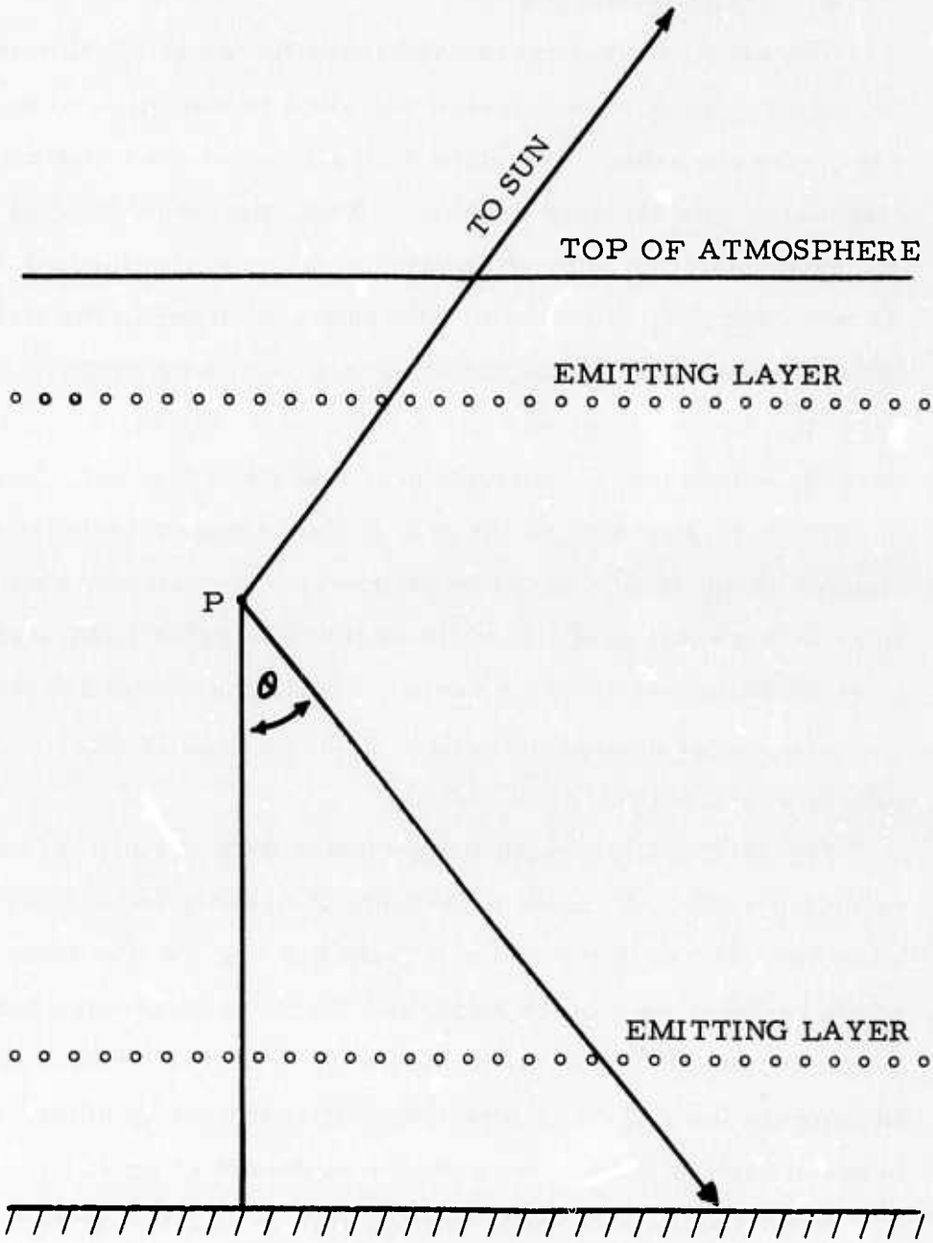
A. INTRODUCTION

The physical processes and assumptions used in developing a model for transport of infrared radiation in the upper atmosphere are discussed below, beginning with a general overview and discussion of a simpler problem. Next, the interaction of electromagnetic radiation with an isolated atom or molecule is described. An understanding of the basic processes involved is the first step in any computation of the radiation from a gas or mixture of gases. Subsequently, absorption from a continuum source, a spectrally varying source and fluorescent processes are treated. The effects of optical thickness when the path of the radiation includes a large number of molecules of the same species is considered for spectral lines with a Voigt profile, which includes Doppler (Gaussian) and Lorentz shapes as limiting cases. Finally, a model for the transport of radiation for a single vibration-rotation band is described, and the effects of temperature estimated.

Figure II-1 illustrates the geometry employed in the current modeling study. A molecule at point P absorbs radiation from three sources; (1) a collimated beam from the sun, (2) the lower boundary which radiates as a black body, and (3) other molecules between the lower and upper boundaries. Radiative transfer functions are needed to compute the effects of absorption of radiation by other molecules between each of the sources and the molecule at point P.

In the absence of collisional excitation and de-excitation, the simplest integral equation that may be written to describe radiative transport in a finite plane parallel atmosphere is

$$J(\tau) = \frac{1}{2} \int_0^{\tau_{max}} K_1(|\tau - t|)\, J(t)dt + S(\tau) \qquad (2\text{-}1)$$



Geometry assumed in deriving the radiative transfer functions

FIGURE II-1

This equation assumes that there is only one radiative transition involved and that the radiation emitted by a molecule after absorption is independent of the direction of incidence. Here τ is a measure of the optical path length in the vertical direction. $J(\tau)$ is a dimensionless measure of the population of the upper, emitting molecular state. $K_1(\tau)$ is the radiative transfer function which accounts for absorption of radiation emitted by molecules at a vertical path length τ distant from the absorbing molecule. $S(\tau)$ is an auxiliary function which includes the effects of radiation originating outside the atmosphere. If there is only a black body source located at the lower boundary from which optical path length is measured ($\tau = 0$ at the lower boundary) then

$$S(\tau) = \frac{1}{2} N_o K_2(\tau) \tag{2-2}$$

Here N_o is the radiance of the lower boundary in arbitrary units such as watts cm^{-2} - $ster^{-1}$ or the spectral radiance in units such as watts cm^{-2} cm^{-2} - $ster^{-1}$ $(\text{wavelength interval})^{-1}$.

In the gray or spectrally independent case, when equation (2-2) holds, the kernel function $K_1(\tau)$ is the first exponential integral $E_1(\tau)$ and the function describing absorption from the black body source is the second exponential integral $E_2(\tau)$. These functions are described in detail in such works as Kourganoff (1963). A complete analytic solution is available and the auxiliary functions required for a numerical solution have been tabulated by King (1956). Figure (2-2) plots the solution of equation (2-1) when equation (2-2) holds for the gray case for values of total optical thickness τ_{max} = 0, 0.1, 1.0, and 10. When the total optical thickness is small, the

Effect of Optical Thickness on Relative Vibrational Excitation of Atmosphere Illuminated From Below By Black Body Source.
[Gray Case]

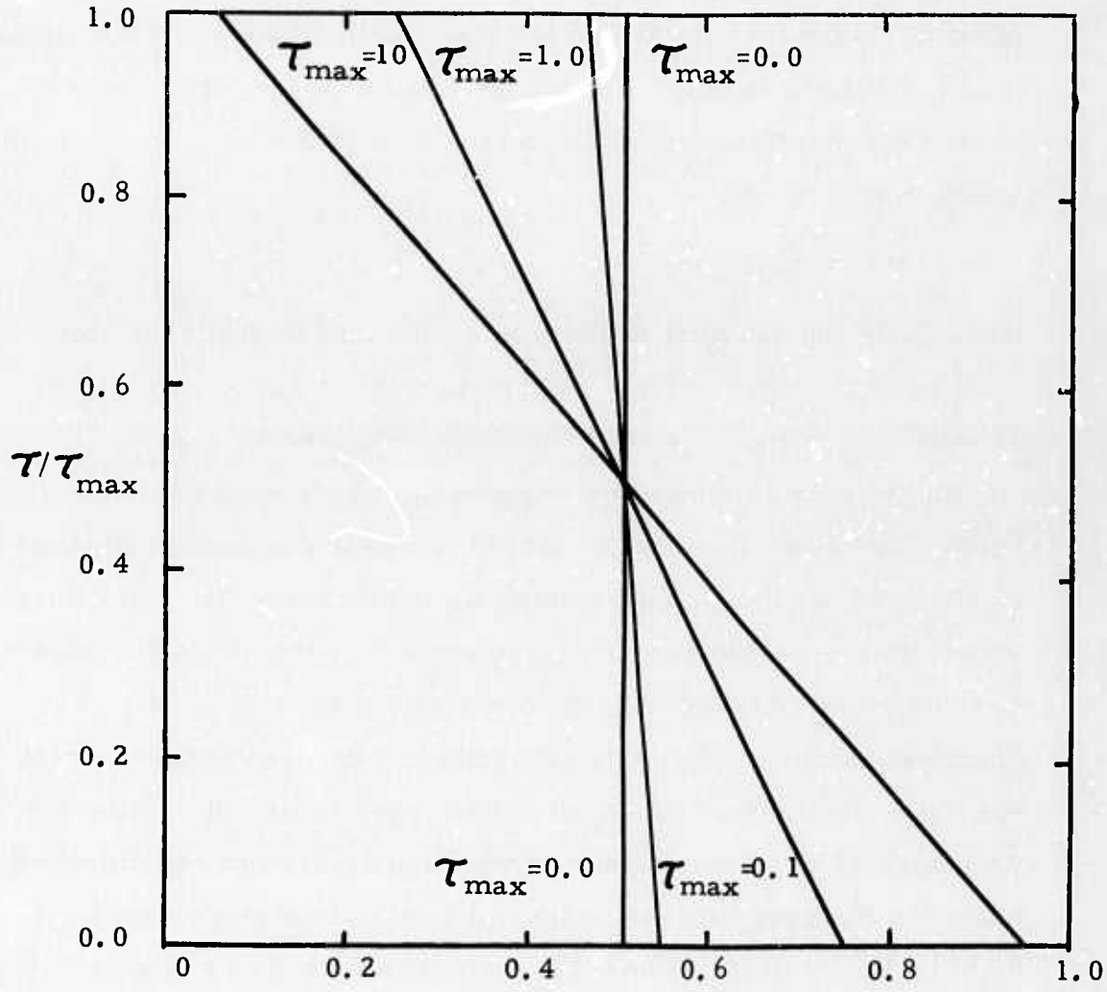


FIGURE II-2

effective excitation of the molecule at any point in the atmosphere is half that of a molecule at the (vibrational) temperature of the black body. As the total optical thickness increases, the excitation of the molecules near the black body increases, while that of molecules near the top of the atmosphere decreases. This occurs because as the optical thickness increases, molecules near the lower boundary absorb an increasing amount of radiation re-radiated from molecules above them. At the same time, less radiation penetrates to higher altitude. The physics of the problem require that if a second black body of the same temperature be placed at the top of the atmopshere, the population of radiators at all points would be uniformly that corresponding to the black body temperature. This symmetry and the linear character of the integral equation (2-1) results in the relation $J(\tau) + J(\tau_{max} - \tau) = N_o$.

When a collimated beam of light is incident on the top of the atmosphere, another term must be added to the function $S(\tau)$ in equations (2-1) and (2-2) to account for this. If the flux is incident at a zenith angle with cosine μ and has an intensity in arbitrary units such as πF watts cm^{-2} normal to its direction, the term added to $S(\tau)$ has the form $FM_1[(\tau_{max} - \tau)/\mu]$. In the gray case, $M_1(\tau)$ is simply $\exp(-\tau)$. The corresponding functions for individual spectral lines and bands will be described after a closer look at the interaction of electromagnetic radiation with a single molecule.

B. INTERACTION OF RADIATION WITH A SINGLE MOLECULE

When treating a problem in which the number of photons emitted or absorbed by an atom or molecule must be considered rather than total energy emitted, it is convenient to examine the physics of the

interaction of radiation with matter through use of the Einstein A and B coefficients. In treating a radiative problem such as the current study of high altitude infrared radiances which also involves transfer of energy between molecules through collisions, this approach is a necessity. In the following discussion, cgs units are used except where explicitly stated.

The Einstein A_{ul} coefficient is the rate at which an atom or molecule in an excited state spontaneously emits radiation. It has the units photons/sec-molecule. In an assemblage of molecules all in a state which can radiate, the average rate of radiation is given by the Einstein A_{ul} coefficient for the particular radiative transition from a state of higher energy (subscript u) to one of lower energy (subscript 1).

In addition to spontaneous emission, two other processes occur. These are absorption, characterized by the Einstein coefficient for absorption B_{lu}, and induced emission, characterized by the Einstein coefficient for induced emission B_{ul}. In a black body radiation field with spectral density, $\rho_{\nu_{lu}}$, the probability that an atom or molecule absorbs a quantum of radiation in unit time is $B_{lu}\rho_{\nu_{lu}}$. When radiation density is expressed in c. g. s. units, ergs-sec/cm^3, the coefficient B_{lu} has units cm^3/ergs-sec. Similarly, an atom or molecule in a radiation field with the above spectral density and in an excited state emits radiation at the rate $B_{ul}\rho_{\nu_{lu}}$ in addition to that characterized by the coefficient A_{ul}. The Einstein coefficients are properties of an atom or molecule. If one coefficient is known, the others may be determined by the relations

$$A_{ul} = 8\pi h B_{ul}/\lambda^3 \qquad (2\text{-}3)$$

and

$$g_1 B_{lu} = g_u B_{u\ell} \tag{2-4}$$

Thus, it is possible to treat the problem knowing only the Einstein A_{ul} coefficient.

Experimental determinations of the interaction of infrared radiation with gases are usually made by measurements of light absorption. Experimental results may be expressed as integrated absorption from a continuum source,

$$S'_{lu} = N_1 \frac{A_{ul}\lambda^2_{lu}}{8\pi} \frac{g_u}{g_1} \left[1 - \exp\left(\frac{hc}{\lambda_{lu} kT}\right)\right] \tag{2-5}$$

or as line or band strengths

$$S_{lu} = S'_{lu}/pc \tag{2-6}$$

where p is the pressure.

In the visible and ultraviolet regions of the spectrum, it is customary to use oscillator strengths to characterize the interaction of radiation and matter. In this case, the defining equation is

$$S'_{lu} = \frac{\pi e^2}{mc} N_1 f_{lu} \left[1 - \exp\left(-\frac{hc}{\lambda_{lu} kT}\right)\right] \tag{2-7}$$

The results of theoretical computations of the interaction of matter with radiation are frequently expressed as dipole moments, μ_{ul}. These are related to the Einstein A coefficient by

$$A_{ul} = \frac{64\pi^4}{3h\lambda^3} \frac{g_1}{g_u} \left|\mu_{ul}\right|^2 \tag{2-8}$$

The rate at which a single atom or molecule absorbs and

reradiates light from a parallel beam from a continuum source at a wavelength λ_o corresponding to a resonance transition is (Mitchell and Zemansky, 1961):

$$G = \frac{\pi e^2}{mc^2} I_o f_o \lambda_o^2 \quad \text{(photons/sec)}$$

$$= 8.852 \times 10^{-13} I_o f_o \lambda_o^2 \quad \text{(photons/sec)} \tag{2-9}$$

where I_o is the continuum source flux and f_o is the oscillator strength. If the incident flux is expressed in photons/cm^2-μm, and oscillator strength is replaced by the equivalent line or band strength S expressed as $cm^{-2}atm^{-1}$, then

$$G = 2.72 \times 10^{-24} I_o S \lambda^2 \tag{2-10}$$

Implicit in this expression is the convenient relation between the Einstein A_{ul} coefficient and line or band strength with wavelenth dimensions μm

$$A_{ul} = 2.804 S(cm^{-2}atm^{-1}) / \lambda^2(\mu m) \tag{2-11}$$

When the source is an extended black body surface with a spectral radiance N_λ, the G factor is obtained by integration over the solid angle subtended by the source. For radiation from a plane parallel surface of infinite extent,

$$G = 1.169 \times 10^{-23} N_\lambda S \lambda^2 \tag{2-12}$$

Another purely radiative mechnaism of importance in upper atmosphere radiance computations is fluorescence. A striking example

occurs for the weak 10.4 μm CO_2 band. Absorption of solar radiation by CO_2 4.3 μm band maintains the ν_3 vibrational mode at a vibrational temperature of about 290 K. During the day, this leads to an enhancement of emission in the 10.4 μm band by as much as two orders of magnitude when viewed along the earth's limb from outside the atmosphere.

Fluorescence occurs when the upper state of the resonance transition reradiates into other than the initial states. The relative probability of a radiative transition to a lower state, n from the upper state, u when there are k possible lower state is:

$$P_{un} = \frac{A_{un}}{\sum_{j=1}^{k} A_{uj}} \qquad (2\text{-}13)$$

where the A_{uj} are the Einstein A coefficients for the spontaneous emission probabilities for a transition from the upper state to the jth lower state.

In the case of molecular resonance-fluorescent scattering at low temperature, only the lowest vibrational level ($v'' = 0$) in the ground electronic state is populated significantly. Absorption of a photon excites the molecule to a vibrational level, v' in the same or a higher electronic state. Radiation is possible from this state to the initial level ($v'' \neq 0$) or to other vibrational levels ($v'' = 0$) in the ground electronic state. The rate of resonance fluorescent scattering in a given electronic vibrational transition may be obtained by combining Equations (2-9) and (2-13) and introducing appropriate subscripts:

$$G_{v'v''} = 8.853 \times 10^{-17} I_{v'o} f_{v'o} \lambda_{v'o}^2 \frac{A_{v'v''}}{\sum_{v''} A_{v'v''}} \qquad (2\text{-}14)$$

Here the subscript $v'v''$ refers to a transition from an upper state with vibrational level v' to a lower state with vibrational level v''. The subscript $v'o$ refers to absorption of a photon by a molecule in the lowest vibrational level of the ground state, exciting the molecule to a vibrational level v' in the same or a higher electronic state.

C. RADIATIVE TRANSFER FUNCTIONS FOR LINES AND BANDS

In developing a realistic model of the interaction of radiation with matter, it is necessary to take into account the details of absorption. Line shape must be considered because it determines the amount of radiation excaping from an optically thick layer of gas. Thus, for large optical thickness (greater than a magnitude of about 10), the total emission from an isolated spectral line with Lorentz shape is proportional to the square root of the optical thickness, while that from a line with Doppler shape is proportional to the square root of the logarithm of the optical thickness. In addition, absorption of radiation emitted at one place in the gas by a molecule at another location cannot be computed without knowledge of the line shape.

At high altitudes, the profile of the spectral absorption curve of an individual line in a vibration-rotation band is predominantly due to velocity or Doppler broadening. Kuhn and London (1969), who investigated the heat budget of the atmosphere between 30 and 110 km, report that radiative transfer calculations require the use of the Voigt profile (including both collisional and Doppler broadening) at

altitudes between 20 and 70 km. Alternatively, the Doppler profile is adequate at higher altitudes where the Doppler shape is the largest contributor to line broadening. The original computational model therefore assumed a Doppler profile.

The discussion below assumes a Voigt profile for a single line, and it will be shown that for small optical thicknesses and low pressures, the difference between calculations with Doppler and Voigt profiles is small. Asymptotic values for large optical thicknesses are quite different, however.

The radiative transfer functions required to treat transfer of resonance radiation in a single line are well known for the Voigt profile and its two limiting cases, the Doppler and Lorentz profiles. Avrett and Hummer (1965) review the properties of four of these functions, giving series expansions for small values of optical thickness and asymptotic expressions for large values of the optical thickness. The notation used for these functions varies widely. That used here is adapted from that used by Ivanov and Shcherbakov (1965 a, b) in their tabulation of functions for the Doppler profile.

The best known of these functions is probably that relating to the equivalent width of a line in absorption or to the total radiation emitted in a line from a source in thermal equilibrium at a constant temperature, the integrated absorbtance, defined as

$$L(a, \tau) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} (1 - e^{-\tau f(a, x)}) dx \qquad (2\text{-}15)$$

Here f(a, x) is used to denote the Voigt profile,

$$f(a, x) = \frac{a}{\pi} \int_{-\infty}^{\infty} \frac{dte^{-t^2}}{(t-x)^2 + a^2} \tag{2-16}$$

The parameter a is proportional to the ratio of the Lorentz half width at half maximum b_L and the Doppler half width at half maximum b_D,

$$a = \frac{b_L}{b_D} (\ln 2)^{1/2} \tag{2-17}$$

The parameter x is proportional to the difference between the spectral frequency of the line center ν_o and that at any other point in the line profile, ν

$$x = \frac{2(\nu - \nu_o)}{b_D} (\ln 2)^{1/2} \tag{2-18}$$

The quantity τ is the optical thickness at the center of a line with Doppler broading alone,

$$\tau = \frac{2\pi e^2}{mc} \frac{Nfl}{b_D} (\ln 2)^{1/2} \tag{2-19}$$

where N is the number density of ground state molecules and l is the path length. The actual optical thickness at the center of a line with Voigt profile is the product of the τ of Equation (2-19) and exp (a^2) times the co-error function of the parameter a, erfc(a). In general, Equation (2-15) must be evaluated numerically. For small values of τ, a series expansion may be obtained in the form

$$L(a, \tau) = -\sum_{n=1}^{\infty} \frac{(-\tau)^n}{n!} \Phi_n(a) \tag{2-20}$$

where

$$\Phi_n(a) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} [f(a,x)]^n dx \qquad (2\text{-}21)$$

For all values of a, $\Phi_1(a)$ is one. When $\Phi_n(a)$ is divided by erfc a, a smooth variation between a = o (Doppler profile) and a = ∞ (Lorentz profile) is obtained as shown in Figure (II-3) for small n. For large values of τ, asymptotic series are available, as given for example by van Trigt (1968).

$$L(o,\tau) \approx 2\left(\frac{\ln \tau}{\pi}\right)^{1/2} \left[1 + \frac{.2886}{\ln \tau} + \ldots\right] \qquad (2\text{-}22a)$$

$$L(a\neq o,\tau) \quad \frac{2(a\tau)^{1/2}}{(\pi)^{1/4}} \left[1 + \frac{\sqrt{\pi}}{8} \frac{3-2a^2}{a\tau} + \ldots\right] \qquad (2\text{-}22b)$$

Equation (2-10) gives the rate at which a single molecule absorbs radiation from a parallel beam. If the molecule is located within an optically thick medium, the rate of absorption of radiation is reduced by a factor which depends on the optical thickness,

$$M_1(a,\tau) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} f(a,x) e^{-\tau f(a,x)} dx \qquad (2\text{-}23)$$

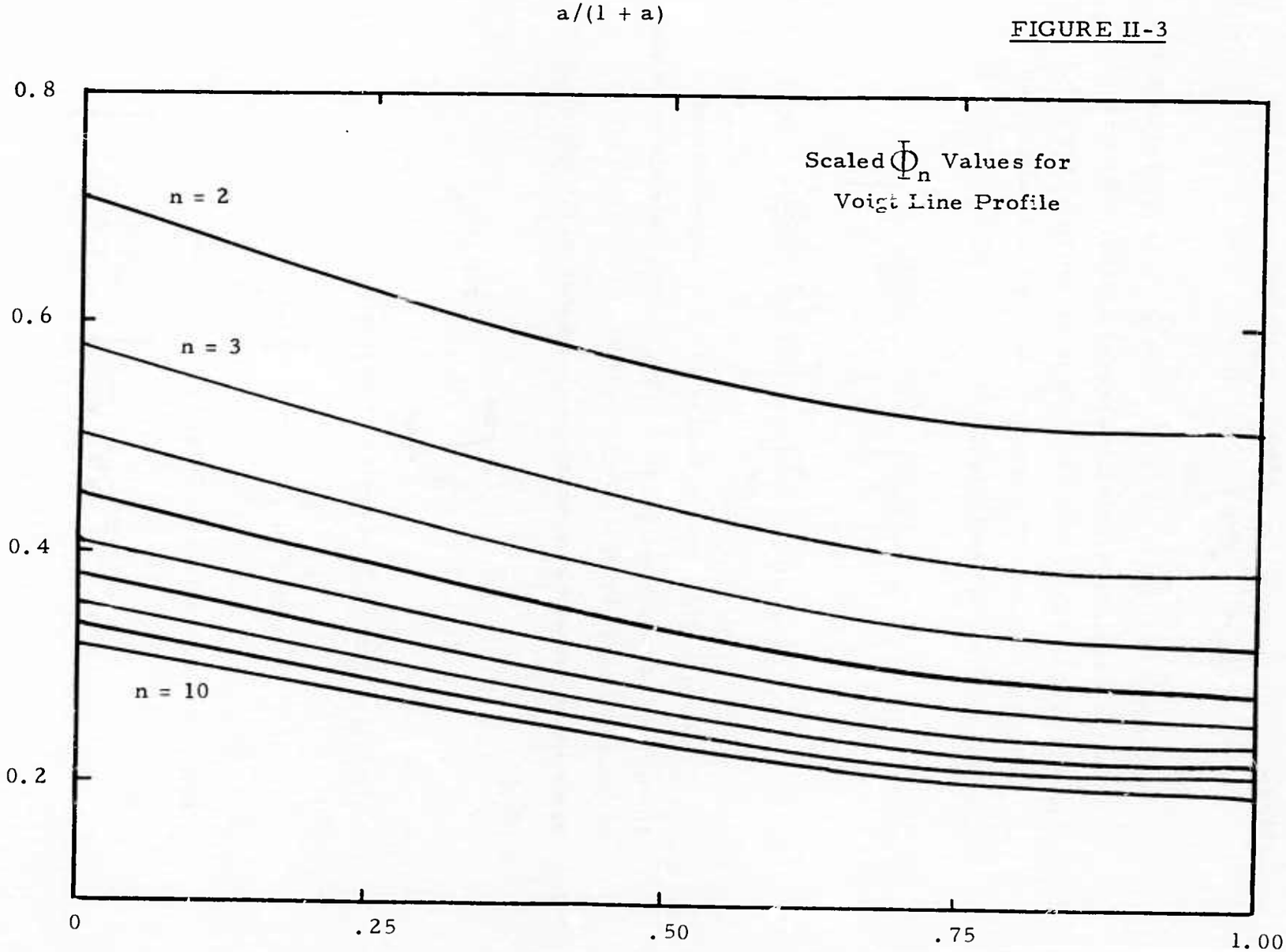
For small values of τ, a series expansion is

$$M_1(a,\tau) = \sum_{n=o}^{\infty} \frac{(-\tau)^n}{n!} \Phi_{n+1}(a) \qquad (2\text{-}24)$$

Asymptotic expressions for large values of are

$$M_1(o,\tau) \approx \frac{1}{\sqrt{\pi}\tau\sqrt{\ln \tau}} \left[1 - \frac{0.2886}{\ln \tau} + \ldots.\right] \qquad (2\text{-}25a)$$

FIGURE II-3


a/(1 + a)
Scaled Φ_n Values for
Voigt Line Profile
n = 2
n = 3
n = 10
0.8
0.6
0.4
0.2
0
.25
.50
.75
1.00

$$M_1(a\neq 0, \tau) \approx \frac{1}{\pi 1/4}\sqrt{\frac{a}{\tau}}\left[1 - \frac{\sqrt{\pi}}{8}\,\frac{3-2a^2}{a\tau} + \cdots\right] \qquad (2\text{-}25b)$$

The function M_1 is the deriviative with respect to τ of the function L. In addition to its use in modifying the G factor of Equation (2-10), it gives the probability of escape of a photon in a direction with optical thickness τ.

A third function is defined as

$$M_2(a, \tau) = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty} f(a, x)^2\, e^{-\tau f(a, x)}\, dx \qquad (2\text{-}26)$$

and has a series expansion

$$M_2(a, \tau) = \sum_{n=o}^{\infty} \frac{(-\tau)^n}{n!}\,\Phi_{n+2}(a) \qquad (2\text{-}27)$$

This function is proportional to the rate at which a molecule absorbs radiation from another molecule at a optical distance τ. This function would be used as the kernel function K_1 in Equation (2-1) if one dimensional radiative transport were being treated. It is included here because of its use in generating the kernel function for a plane parallel geometry.

When the source of radiation is an infinitely extended plane black body, the function which is used to determine the rate at which a molecule absorbs radiation is given by

$$N_{12}(a, \tau) = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty} f(a, x)\, E_2[\tau f(a, x)]\, dx \qquad (2\text{-}28a)$$

$$= \tau\int_{\tau}^{\infty} M_1(t)\,\frac{dt}{t^2} \qquad (2\text{-}28b)$$

$$= \int_0^{\pi/2} M_1\left(\frac{\tau}{\cos\theta}\right)\sin\theta\, d\theta \qquad (2\text{-}28c)$$

The series expansion for this function is given by

$$N_{12}(a,\tau) = \Phi_1(a) + (\gamma\tau - \tau + \tau \ln \tau)\Phi_2(a)$$

$$+ \tau\Phi_L(a) - \sum_{n=2}^{\infty} \frac{(-\tau)^n}{(n-1)n!}\,\Phi_{n+1} \qquad (2\text{-}29)$$

Here a new integral is defined,

$$\Phi_L(a) = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty} [f(a,x)]^2 \ln f(a,x)\,dx$$

and γ is the Euler-Mascheroni constant 0.5772... The asymptotic expansions

$$N_{12}(o,\tau) \approx \frac{1}{\sqrt{\pi}\,\tau\sqrt{\ln\tau}}\left(0.5 - \frac{0.2693}{\ln\tau} + \ldots.\right) \qquad (2\text{-}30a)$$

$$N_{12}(a\neq o,\tau) \approx \frac{1}{\pi^{1/4}}\left(\frac{a}{\tau}\right)^{1/2}\left(\frac{2}{3} + \ldots.\right) \qquad (2\text{-}30b)$$

may be obtained from Equations (2-29b) and (2-25).

The kernel function for a plane parallel geometry is given by

$$N_{21}(a,\tau) = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty} [f(a,x)]^2\, E_1[\tau f(a,x)]\,dx \qquad (2\text{-}31a)$$

$$= \int_{\tau}^{\infty} M_2(t)\,\frac{dt}{t} \tag{2-31b}$$

$$= \int_{o}^{\pi/2} M_2 \left(\frac{\tau}{\cos\theta}\right) \frac{\sin\theta\, d\theta}{\cos\theta} \tag{2-31c}$$

The series expansion useful for small τ is

$$N_{21}(a,\tau) = -(\gamma + \ln\tau)\Phi_2 - \Phi_L - \sum_{n=1}^{\infty} \frac{(-\tau)^n}{n \cdot n!}\Phi_{n+2} \tag{2-32}$$

and the asymptotic expansions are

$$N_{21}(o,\tau) \approx \frac{1}{\sqrt{\pi}\,\tau^2\sqrt{\ln\tau}}\left(0.5 - \frac{0.0193}{\ln\tau} + \ldots\right) \tag{2-33a}$$

$$N_{21}(a\neq o,\tau) \approx \frac{a^{1/2}}{\pi^{1/4}\tau^{3/2}}\left(\frac{1}{3} + \ldots\right) \tag{2-33b}$$

The recurrence relation

$$N_{kn}(a,\tau) = \frac{1}{n-1}\left[M_k(a,\tau) - \tau N_{k+1,\,n-1}(a,\tau)\right] \tag{2-34}$$

carries over into the corresponding band functions and provides a partial check on numerical calculations.

In extending this treatment of radiative transfer to a vibration-rotation band, in addition to assuming that each line has the same Voigt profile shape, the following assumptions are made:

(1) Rotational levels within a vibrational level remain in

equilibrium with the translational temperature. This is valid up to at least 150 km according to Goody (1964).

(2) There is complete redistribution of frequency within a single line following absorption of radiation.

(3) There is complete redistribution of absorbed energy among all rotational levels.

(4) Line strengths are used as if each line were at the band center, neglecting wavelength variation across a band.

(5) There is no overlap of adjacent lines.

(6) Temperature variations at different levels do not affect the radiative transfer.

Let g_J be the fraction of total band strength associated with a single line, so that

$$\sum_J g_J = 1 \tag{2-35}$$

and the sum over J may involve complete summation over rotational quantum number for two or more branches. The optical thickness at the center of a single line will be given by τg_J, where τ is the optical thickness that would result if the entire band strength were in a single line. Then the radiative transfer functions for a band may be defined as functions of radiative transfer functions for a single line. First,

$$L^b(a, \tau) = \sum_J L(a, \tau g_J) \tag{2-36}$$

where the superscript b refers to an entire band.

In order that the correct derivative relations

$$M_1^{\,b}(a,\tau) = \frac{d}{d\tau}\, L^{b}(a,\tau) \tag{2-37a}$$

$$M_2^{\,b}(a,\tau) = -\frac{d}{d\tau}\, M_1^{\,b}(a,\tau) \tag{2-37b}$$

and the integral relations of Equations (2-28b) and (2-31b) hold,

$$M_1^{\,b}(a,\tau) = \sum_J g_J\, M_1(a,\tau g_J) \tag{2-38a}$$

$$M_2^{\,b}(a,\tau) = \sum_J g_J^2\, M_2(a,\tau g_J) \tag{2-38b}$$

$$= N_{12}^{\,b}(a,\tau) = \sum_J g_J\, N_{12}(a,\tau g_J) \tag{2-38c}$$

$$N_{21}(a,\tau) = \sum_J g_J^2\, N_{21}(a,\tau g_J) \tag{2-38d}$$

While these functions must be evaluated numerically, it is possible to perform this evaluation in two different ways for small values of τ. Making the definitions

$$G_n = \sum_J g_J \tag{2-39a}$$

$$G_L = \sum_J g_J^2 \ln g_J \tag{2-39b}$$

The following series expansions are obtained:

$$L^b(a,\tau) = -\sum_{n=1}^{\infty} \frac{(-\tau)^n}{n!} G_n \Phi_n(a) \tag{2-40a}$$

$$M_1^b(a,\tau) = \sum_{n=1}^{\infty} \frac{(-\tau)^n}{n!} G_{n+1} \Phi_{n+1}(a) \tag{2-40b}$$

$$M_2^b(a,\tau) = \sum_{n=o}^{\infty} \frac{(-\tau)^n}{n!} G_{n+2} \Phi_{n+2}(a) \tag{2-40c}$$

$$L_3^b(a,\tau) = G_1 \Phi_1(a)\tau + \frac{1}{2}\left[\left(G_2 \ln\tau + G_L + \gamma - \frac{3}{2}\right)\Phi_2(a) + G_2 \Phi_L(a)\right]\tau^2 + \sum_{n=3}^{\infty} \frac{(-\tau)^n}{(n-2)n!} G_n \Phi_n(a) \tag{2-40d}$$

$$N_{12}^b(a,\tau) = G_1 \Phi_1(a) + G_2 \Phi_2(a)\tau \ln\tau + \left[G_L \Phi_2(a) + (\gamma-1)G_2 \Phi_2(a) + G_L \Phi_L(a)\right]\tau - \sum_{n=2}^{\infty} \frac{(-\tau)^n}{(n-1)n!} G_{n+1} \Phi_{n+1}(a) \tag{2-40e}$$

$$N_{21}(a,\tau) = -\left[\gamma\Phi_2(a) + \Phi_L(a)\right] G_2 - G_2\Phi_2(a)\ln\tau$$

$$- G_L\Phi_2(a) - \sum_{n=1}^{\infty} \frac{(-\tau)^n}{n \cdot n!} G_{n+2}\Phi_{n+2}(a) \qquad (2\text{-}40f)$$

For large values of the optical thickness parameter τ, there is a sharp distinction between the asymptotic expansions for Doppler line shape and Voigt line shape. When Doppler line shape is assumed, the asymptotic forms for the functions discussed above differ from those for a single line by a factor proportional to $(\ln\tau)^{1/2}$ in the case of bands for a linear molecule. The numerical results suggest that the asymptotic expansions for an asymmetric rotor are multiplied by an additional factor proportional to $\ln\tau$. In the case of a Voigt profile, there is no such change in the asymptotic form. The L functions retain their $\tau^{1/2}$ dependence, the M_1 and N_{12} functions retain the $\tau^{-1/2}$ dependence, and the M_2 and N_{21} functions retain the $\tau^{-3/2}$ dependence. The chief complication with the functions for Voigt profiles is that for intermediate values of τ the behavior is similar to that for a Doppler profile, while at some value of τ dependent on the parameter a, the functions begin to tend toward the asymptotic form. Similar behavior has been found by Malkmus (1967, 1968) for curves of growth for various random band models.

A strict derivation of the asymptotic expansions for the Doppler profile functions has not been obtained. In the high rotational temperature limit, the first term in the expansions for parallel and perpendicular bands of linear molecules appear to be

	Parallel	Perpendicular	
$L^b(o,\tau) \sim$	$\sqrt{\pi\epsilon}\,Q_R \ln\tau$	$\frac{3}{2}\sqrt{\pi\epsilon}\,Q_R \ln\tau$	(2-41a)
$M_1^b(o,\tau) \sim$	$\sqrt{\pi\epsilon}\,Q_R/\tau$	$\frac{3}{2}\sqrt{\pi\epsilon}\,Q_R/\tau$	(2-41b)
$M_2^b(o,\tau) \sim$	$\sqrt{\pi\epsilon}\,Q_R/\tau^2$	$\frac{3}{2}\sqrt{\pi\epsilon}\,Q_R/\tau^2$	(2-41c)
$L_3^b(o,\tau) \sim$	$\frac{1}{2}\sqrt{\pi\epsilon}\,Q_R \ln\tau$	$\frac{3}{4}\sqrt{\pi\epsilon}\,Q_R \ln\tau$	(2-41d)
$N_{12}^b(o,\tau) \sim$	$\frac{1}{2}\sqrt{\pi\epsilon}\,Q_R/\tau$	$\frac{3}{4}\sqrt{\pi\epsilon}\,Q_R/\tau$	(2-41e)
$N_{21}^b(o,\tau) \sim$	$\frac{1}{2}\sqrt{\pi\epsilon}\,Q_R/\tau^2$	$\frac{3}{4}\sqrt{\pi\epsilon}\,Q_R/\tau^2$	(2-41f)

Here Q_R is the rotational partition function and ϵ is hcB_v/kT, the dimensionless ratio of rotational quantum energy to kinetic temperature.

The first term in the asymptotic expansions for Voigt profile band functions is easily obtained, as is illustrated for the $L^b(a,\tau)$ function. At large optical paths, $L(a,\tau)$ for a single line is from Equation (2-21b) approximately $2\pi^{1/4}(a\tau)^{1/2}$. Since the optical thickness at the center of a line in a band is simply $g_J\tau$, then

$$L^b(a,\tau) \sim 2\pi^{-1/4}\, a^{1/2} \sum_J (\tau g_J)^{1/2}$$

or, denoting $\sum_J g_J{}^{1/2}$ by $G_{1/2}$

$$L^b(a,\tau) \sim \frac{2}{\pi^{1/4}} (a\tau)^{1/2} G_{1/2} \tag{2-42a}$$

and

$$M_1^b(a,\tau) \sim \frac{(a/\tau)^{1/2}}{\pi^{1/4}} G_{1/2} \tag{2-42b}$$

$$M_2^b(a,\tau) \sim \frac{a^{1/2}}{2\pi^{1/4}\tau^{3/2}} G_{1/2} \tag{2-42c}$$

$$L_3^b(a,\tau) \sim \frac{4(a\tau)^{1/2}}{3\pi^{1/4}} G_{1/2} \tag{2-42d}$$

$$N_{12}^b(a,\tau) \sim \frac{2}{3\pi^{1/4}} \left(\frac{a}{\tau}\right)^{1/2} G_{1/2} \tag{2-42e}$$

$$N_{21}^b(a,\tau) \sim \frac{a^{1/2}}{3\pi^{1/4}\tau^{3/2}} G_{1/2} \tag{2-42f}$$

In order to simplify programming, a single set of functions is used for all parallel bands of linear molecules and a single set is used for perpendicular bands. These functions vary with the rotational temperature and in the program, in order to partially account for temperature variation, the parameter τ is not used, but rather a derived parameter τ_o, which is the optical thickness at the center of the strongest line in the band. This is either an R branch line for a parallel band or a Q branch line for a perpendicular band.

The effects of temperature variation will be illustrated by the

results of calculations for the $L^b(o,\tau)$ function for a perpendicular band of a linear molecule. The discussion assumes the Doppler profile because it is possible to obtain a series expansion for the high temperature limit in the variable τ_o. Use of double precision computation permits accurate calculation into the region where the logarithmic approximation to $L^b(o,\tau)$ becomes good.

The band used as an example is a "perpendicular" transition with upper state a $^1\pi$ level and lower state a $^1\Sigma$ level. It is assumed that the molecule is not symmetric and there are no nuclear spin effects. The relative strength of each line is the product of the exponential part of the Boltzmann factor of the rotational energy level of the lower state and the appropriated Hoenl-London transition probability or factor. For the Jth rotational level, the Boltzmann factor is $\exp[-\epsilon J(J+1)]$ where $\epsilon = hcB_v/kT$ and h is Planck's constant, B_v is the rotational energy constant for the lower vibrational state, and k is Boltzmann's constant. The Hoenl-London factors for the P, Q, and R branches are

$$S_J^P = \frac{J-1}{2} \tag{2-43a}$$

$$S_J^Q = \frac{2J+1}{2} \tag{2-43b}$$

$$S_J^R = \frac{J+2}{2} \tag{2-43c}$$

and have been normalized so that the sume for a given value of J is 2J + 1.

The rotational partition function is the sum over all rotational levels of the product of the rotational degeneracy factor $2J + 1$ and the Boltzmann factor for each level. At high temperatures, an asymptotic approximation is available for the partition function.

$$Q_R = \sum_{J=0}^{\infty} (2J + 1) \exp [- \epsilon J(J + 1)] \tag{2-44a}$$

$$\simeq \frac{1}{\epsilon} \left(1 + \frac{\epsilon}{3} + \frac{\epsilon^2}{15} + \frac{4\epsilon^3}{315} + \ldots.\right) \tag{2-44b}$$

The J value of the strongest line in the Q branch may be found by setting

$$\frac{d}{dJ} \frac{1}{2} (2J + 1) \exp[- \epsilon J(J + 1)] = 0 \tag{2-45}$$

and solving for J. The result is

$$J_{max} = \frac{1}{2}\sqrt{\frac{2}{\epsilon}} - \frac{1}{2} \tag{2-46}$$

Adopting $Q_R \simeq 1/\epsilon$, the fraction of band strength in the strongest line is $\sqrt{\epsilon/2} \exp(- 1/2 + \epsilon/4)$ and the optical thickness at the center of this (fictitious) line is

$$\tau_o = \tau\sqrt{\epsilon/2} \exp (- 1/2 + \epsilon/4) \tag{2-47}$$

using $k = \tau_o \sqrt{\epsilon/2} \exp (1/2 - \epsilon/4)$, the optical thickness at the center of any line in any branch is

$$\tau_J^P = k_o(J - 1) \exp [- \epsilon J(J +1)] \tag{2-47a}$$

$$\tau_J^Q = k\,(2J+1)\exp\left[-\epsilon J(J+1)\right] \tag{2-47b}$$

$$\tau_J^R = k(J+2)\exp\left[-\epsilon J(J+1)\right] \tag{2-47c}$$

These expressions are then substituted in Equations (2-36) and (2-15) to obtain

$$L^b(o,\tau;\ \epsilon\to o) = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty}\sum_{J;P,Q,R}\left(1-e^{-k\tau_J^{(P,Q,R)}e^{-x^2}}\right)dx \tag{2-48}$$

where $f(o,\ x) = e^{-x^2}$ is used for the Doppler profile. The sum over J may be replaced by an integral over $z = J(J+1)$ and the summands $1 - \exp[-kf(z,\ x)]$ expanded in series and integrated over x and z. After simplification,

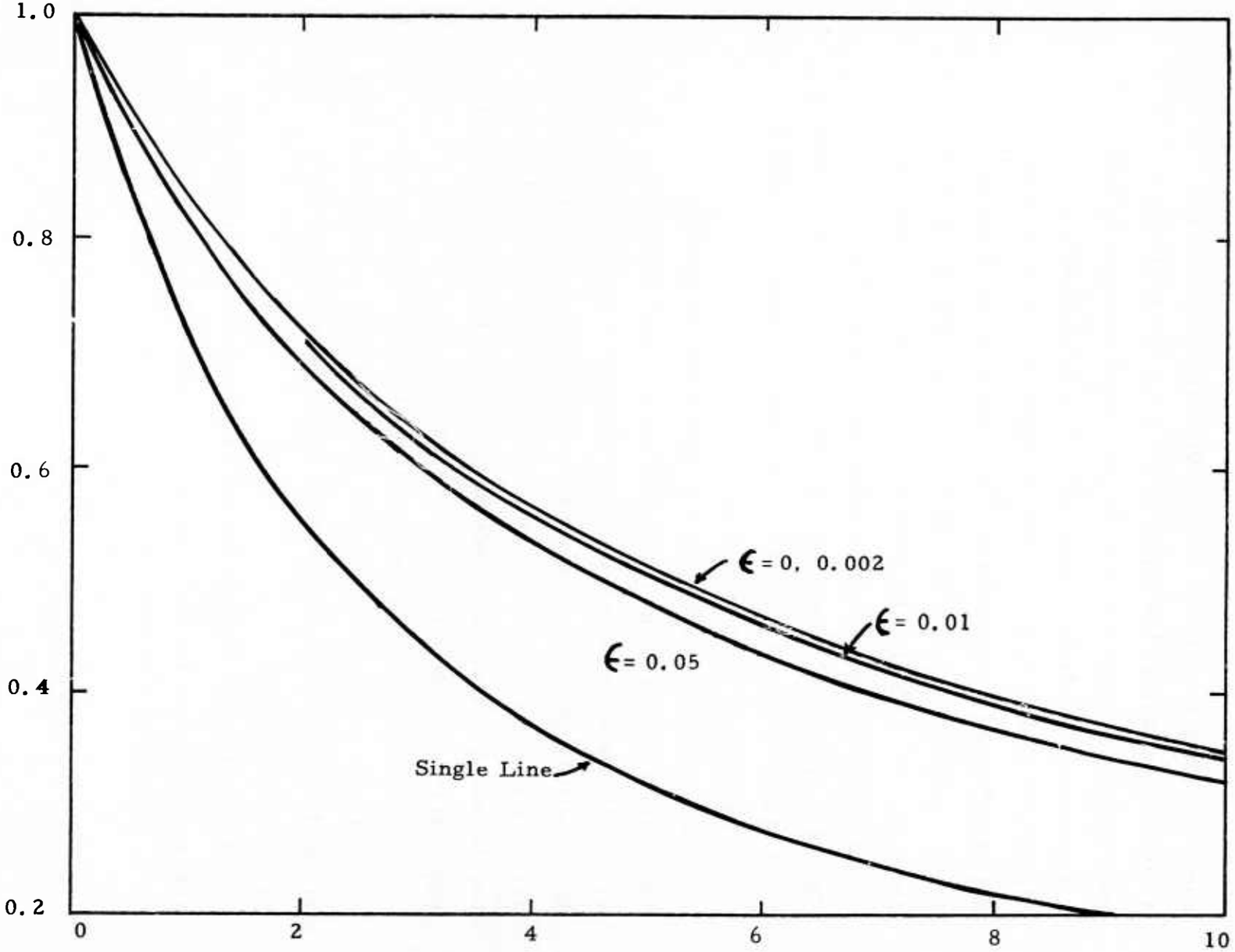
$$L^b(o,\tau;\epsilon\to o) = \tau\sum_{n=1}^{\infty}\frac{(-\tau_o)^{n-1}e^{(n-1)/2}}{n;\sqrt{n}}\,(2^{n-1}+1)(2n)^{-(n+1)/2}\,\Gamma\left(\frac{n+1}{2}\right) \tag{2-49}$$

Use of double precision arithmetic on the CDC 6600 permits evaluation of Equation (2-49) for values of τ_o up to about 50.

Define the function

$$S(\tau) = \frac{1}{\tau}\ L^b\ (o,\tau) \tag{2-50}$$

This function is plotted in Figure (II-4) for values of τ between zero and ten, using the series of Equation (2-49) for $\epsilon = 0$ and



Comparison of Values of $S(\tau_o)$ for Perpendicular Bands at Different Reduced Rotational Temperatures.

Note the more rapid decrease for a single line.

numerical integration for $\epsilon = 0.002$, 0.01 and 0.05. On the scale of this drawing, the curves of $\epsilon = 0$ and $\epsilon = 0.002$ are indistinguishable, the values at $\tau = 10$ being 0.3501 and 0.3483 respectively. For larger values of τ, it is convenient to plot $\tau S(\tau)/\ln\tau$ versus $(\ln\tau)^{-1}$. This is shown in Figure (II-5) for values of τ between 10 and 50 for $\epsilon = 0$, and between 10 and 10^{100} for $\epsilon = 0.002$, 0.01 and 0.05. At $\tau = 50$, the value for $\epsilon = 0$ is about one percent higher than that for $\epsilon = 0.002$. The value of $\epsilon = 0.05$ is about eight percent lower than that for $\epsilon = 0.002$. The finite values of ϵ used here correspond to rotational temperatures of about 280K, 56K, and 1.1K for carbon dioxide and 1400 K, 280K, and 56K, for carbon monoxide.

The asymptotic behavior of the other radiative transfer functions is similar in the high temperature limit, taking into account the differences in the leading term of the expansion for large optical depth. In the upper atmosphere, the region in which radiative transfer is the most significant factor affecting vibrational populations is at such an altitude that the Doppler shape is the main contributor to the effects of radiative transfer.

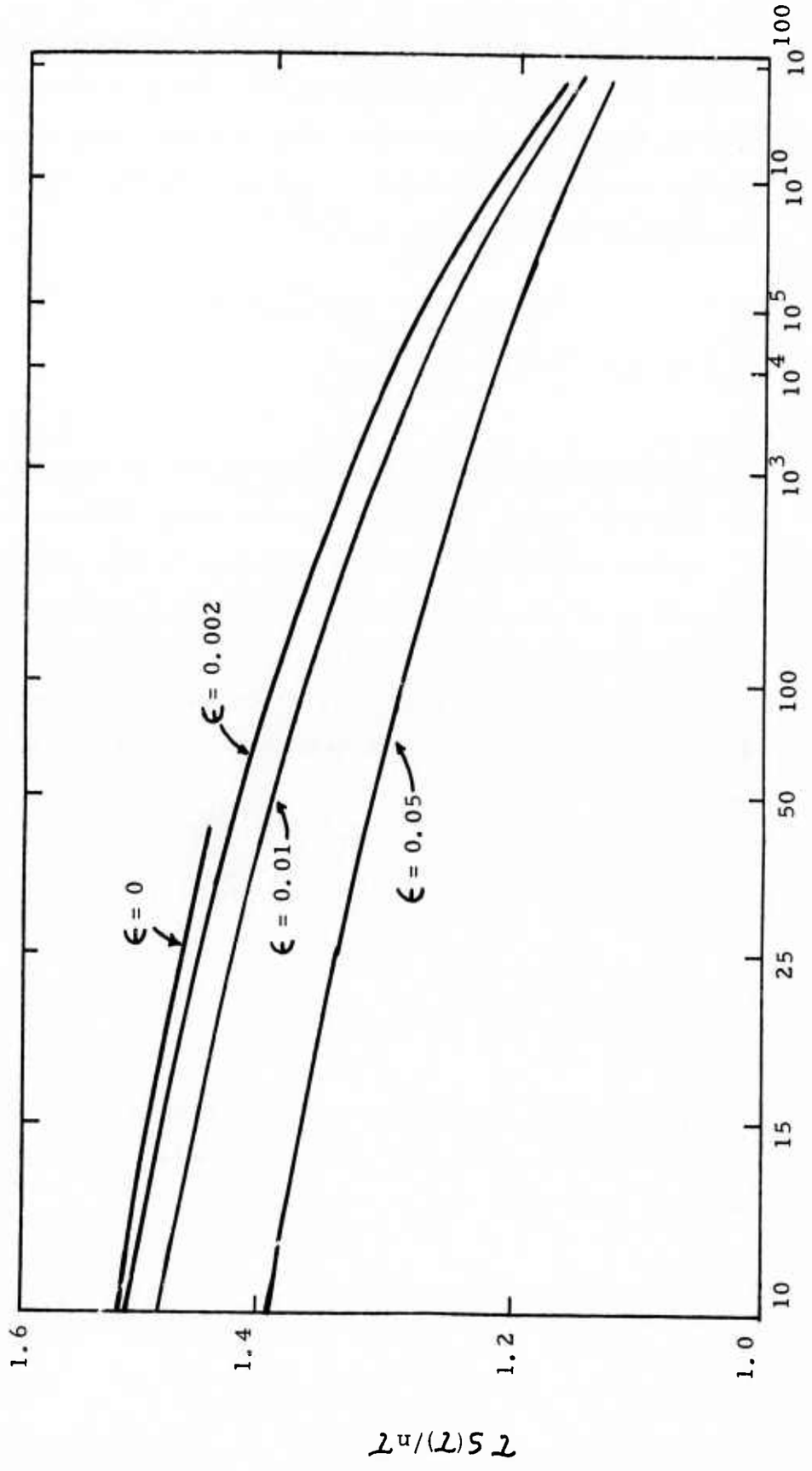
The next step is to fit these radiative transfer functions into the context of obtaining a solution to Equation (2-1) and the extensions of Equation (2-1) when the effects of collisional excitation and de-excitation are added.

A definite lower limit to the source function at the far boundary may be obtained by assuming the source function to be linear in τ. In the gray case, this leads to

$$J(\tau) = \frac{1 - 2\epsilon_3(\tau_o)}{2(1 + \tau_o - e^{-\tau_o})} \qquad (2\text{-}51)$$

FIGURE II-5

Comparison of Values of $\tau_o S(\tau_o)/\ln \tau_o$ for Perpendicular Bands at Different Reduced Rotational Temperatures and Large Values of the Optical Thickness τ



where τ_o again refers to the total optical path length through a finite plane layer. This expression is never in error by more than 13.4%, when compared with King's (1956) exact solution. Making the same approximation for line or band radiative transport functions gives the result

$$J(\tau_o) = \frac{2L_3(\tau_o)}{\tau_o + L(\tau_o)} \tag{2-52}$$

This becomes increasingly inaccurate as the optical path through the layer increases, because of the long range effects of line shape.

Ivanov (1972) has obtained an symptotic solution to two radiative transfer problems involving line radiation with arbitrary line shape. The first is that of a plane layer of finite optical thickness τ_o in which one boundary layer is in radiative quilibrium and is the sole source of radiative flux to the medium. The corresponding integral equation is

$$\Phi(\tau, \tau_o) = \frac{\lambda}{2}\int_o^{\tau_o} \Phi(t, \tau_o) N_{21}(|t - \tau|)\, dt$$

$$+ \frac{\lambda}{2} N_{21}(\tau) \tag{2-53}$$

The parameter λ allows for loss of radiation by "true" absorption or collisional de-excitation. The asymptotic solution to this equation when τ_o is very large is

$$\Phi^a(\tau,\tau_o) = \frac{\frac{\lambda}{2} N_{21}(\tau)}{[1-\lambda+\lambda N_{12}(\tau)]^{3/2}} \left(\frac{1-\lambda+\lambda N_{12}(\tau_o)}{1-\lambda+\lambda N_{12}(\tau_o-\tau)} \right) \quad (2\text{-}54)$$

The second problem is that of a plane layer with a constant excitation mechanism and no external source of radiation. The integral equation is

$$S(\tau) = \frac{\lambda}{2} \int_o^{\tau_o} N_{21}(\tau - t)\, S(t)dt + 1 \quad (2\text{-}55)$$

and the asymptotic solution is

$$S^a(\tau) = [1-\lambda+\lambda N_{12}(\tau)]^{-1/2} [1-\lambda+\lambda N_{12}(\tau_o-\tau)]^{-1/2} \quad (2\text{-}56)$$

This last result may then be used to provide an asymptotic solution to the earthshine problem with no collisions, in which case λ is equal to one. A lower limit to the excitation from below is $\frac{1}{2} N_{12}(\tau_o)$. From symmetry,

$$\Phi(\tau_o/2, \tau_o) = \frac{1}{2} \quad (2\text{-}57a)$$

$$\Phi(\tau, \tau_o) = 1 - \Phi(\tau_o - \tau, \tau_o) \quad (2\text{-}57b)$$

A lower (in the asymptotic case of large τ_o) limit to the source function for τ greater than $\tau_o/2$ is then

$$\Phi^a(\tau,\tau_o) = \frac{1}{2} \frac{N_{12}(\tau_o)}{[N_{12}(\tau)\, N_{12}(\tau_o-\tau)]}, \tau \geq \tau_o/2 \quad (2\text{-}58)$$

This gives low values in the interior, and a bridging formula which gives a good initial solution which can be refined by iteration is

$$\Phi^{a}(\tau, \tau_{o}) = \frac{1}{2}\left[N_{12}(\tau)/N_{12}(\tau_{o} - \tau)\right]^{1/2}, \quad \tau \geq \tau_{o}/2 \tag{2-59}$$

This can be used with Equation (2-57b) to cover the entire range of τ from zero to τ_{o}.

In practice, rather than formally integrate a modified Equation (2-1) including a τ dependent λ to account for collisional de-excitation, and an additional source term to account for collisional excitation, the computer program sets an initial vibrational population at each altitude and computes the total radiative excitation at each level which results from the sum of radiation from the earth and lower atmosphere, the sun and other levels in the atmosphere.

The band functions are used to determine the effects of absorption between a molecule at a given altitude and the various sources of radiation affecting a given band in the following way. The vertical optical paths to the lower boundary and to each level in the atmosphere are computed and, for daytime conditions, the optical path to the assumed top of the atmosphere in the direction of the sun using the secant of the solar zenith angle as a multiple. The effective radiation from the lower boundary is determined by multiplying the surface radiancy by the appropriate N_{12}^{b} function. The effective radiation from the sun is determined by use of the proper M_{1}^{b} function. The effective radiation from the other levels of the atmosphere may be obtained by integrating the product of emission at each level and the band function N_{21}^{b}. This is no

longer done directly, however, as it was discovered that the earlier integration subroutine gave incorrect results. Instead, the assumption is now made that the relative vibrational population between two points varies linearly with optical thickness. Then the integral relations between the band functions are used in the form

$$\int_{\tau_1}^{\tau_2} J(t)\, N_{21}^{b}(a,t)\,dt = \int_{\tau_1}^{\tau_2} (A + Bt)\, N_{21}(a,t)\,dt$$

$$= J(\tau_1)\, N_{12}^{b}(a,\tau_1) - J(\tau_2)\, N_{12}^{b}(A,\tau_2)$$

$$+ \frac{J(\tau_2) - J(\tau_1)}{\tau_2 - \tau_1} \left[L_3^{b}(A,\tau_L) - L_3^{b}(a,\tau_1) \right] \qquad (2\text{-}60)$$

Vibrational populations are determined by balancing collisional and radiative excitation and de-excitation rates. Total band intensities at an observation point inside or outside the atmosphere are computed by determining the optical path from each level to the observation point and integrating the product of local emission rate and the band function $M_1^{b}(a,\tau)$. The spectral distribution of radiation is determined by finding a mean effective temperature and line of sight column density for the radiating molecules, computing intensity for each line in the band with the line function $L(a,\tau)$, summing the line emission rates and scaling each line emission rate so that the total band intensity is obtained.

The above analysis has been presented for the general Voigt profile. Actual use of the radiative transfer functions in the programs has been restricted to the Doppler profile. This restriction is adequate for the water vapor and ozone bands which in the concentrations used in the program have a maximum vertical optical thickness at line center less than 3. For limb viewing of carbon dioxide bands the Voigt profile should be used. Efforts to find a general way of setting up the radiative transfer functions to use a Voigt profile that varies with altitude have not been successful. Exploratory calculations show that the computed radiance values for strongest bands at a tangent height of 60 kilometers may be in error by 20 -- 50 %. In view of the uncertainties in chemical abundances, this error is not felt to be serious.

The same radiative transfer functions have been used for the non-linear molecules as were used for the linear molecules. The match between radiative transfer functions has been made by determining a single "effective" rotational constant for the non-linear tri-atomic molecules. This was done by computing actual radiative transfer functions for a limited range of optical thicknesses adequate to cover that needed in the model. The computed functions were compared with those for the linear molecules and the effective rotational constant needed to produce an approximate match at optical thickness at line center of about 2 was used.

SECTION III

ATMOSPHERIC COMPOSITION

There have been relatively few measurements of the variation with altitude of infrared emitting species at altitudes above 40 km, the limit of direct sampling by balloons. While ozone measurements by different workers agree reasonably well with each other and with theoretical computations at altitudes up to 100 km, and nitric oxide profiles appear reasonably well known between about 80 and 150 km at middle lattitudes, corresponding data are lacking for other species which are important. There is in some cases indirect data available which when combined with computational models can yield limits to concentrations. Thus, Anderson (1971 a) has been able to put an upper limit on total hydroxyl radical abundance above 60 km from measurement of resonance fluorescence and with this result computes possible water vapor profiles between 50 and 100 km using a photochemical model that includes vertical transport by eddy diffusion. In a later paper, Anderson (1971 b) gives a measured altitude profile for hydroxyl with error limits $\pm$ 120 per cent.

At present, then, estimates of chemical abundances of minor species required to predict infrared radiation require the use of photochemical computations. Even with the inclusion of best estimates of the effects of such transport processes as eddy mixing, these computations must be used with caution, even when they appear to agree with the limited concentration determinations that are available. Wofsy and McElroy (1973) give a discussion of methane concentrations in the stratosphere which illustrates the interplay of experiment and

modelling studies, showing how estimates of eddy mixing coefficients may be obtained.

A knowledge of the rates of the appropriate chemical reactions is necessary in order to estimate the abundance of trace constituents at high altitudes in the earth's atmosphere. A set of reaction rates, with estimates of probable error in the rates was listed in the final report on the previous research(Degges, 1972). This list was necessarily incomplete and several of the rates quoted were only estimates. Several reactions have been measured only at room temperature and others only at flame temperatures. In the latter case, only upper or lower bounds may be known.

Since that report was completed, a series of evaluations of chemical reaction rate coefficients has been published by workers at the National Bureau of Standards as a part of the Department of Transportation Climatic Impact Assessment Program. The most recent of these compilations, edited by Garvin and Hampson (1974) provides a thorough summary and evaluation of the accuracy of rate coefficients for neutral chemical relations and photochemistry, vibrational energy transfer processes, high temperature air reactions, and ion-molecule reactions. These reports are the main source of data used in updating the chemistry program.

Among the most important new measurements which have been made are those of Davis, Herron and Huie at temperatures between 200 and 350 K for the reactions

$$O + NO_2 \longrightarrow NO + O_2 \qquad (3\text{-}1)$$

and

$$O + O_2 + Ar \longrightarrow O_3 + Ar \qquad (3\text{-}2)$$

Reaction (3-1) is found to be temperature independent with a rate constant 9.1×10^{-12} cm^3/molec-sec in contrast to the exp (-300/T) temperature dependence of the analysis of Baulch, et al (1970) which we have used previously. Reaction (3-2) is found to have a temperature dependence of exp (500/T) rather than the exp (1060/T) given by Johnston (1968) and is slower at low temperatures than previously believed. This implies slower ozone formation and slightly greater atomic oxygen concentrations at high altitudes.

Many reaction rates remain poorly determined, and the compilation of Garvin and Hampson points these out clearly. One such reaction is

$$OH + HO_2 \longrightarrow H_2O + O_2 \qquad (3\text{-}3)$$

where measurements of the rate range between 10^{-11} and 2×10^{-12} cm^3 $molecules^{-1}$ s^{-1}. They suggest that model calculations should include the extreme ranges.

Water vapor abundances are not well known, even in the stratosphere. Balloon measurements are limited to altitudes below 40 km and are subject to considerable uncertainty as to the amount of water vapor carried to high altitude by the balloon. Direct measurements put lower and upper limits of the volume mixing ratio at two and ten parts per million in the upper stratosphere. A recent optical measurement from a balloon has been made by Goldman, et al (1973) who observed thermal emission from the rotational spectrum near 25 micrometers. Near and below 30 km, the volume mixing ratio is nearly three parts per million. In the region between 30 and 36 km, above the peak altitude reached by the balloon, the mixing ratio appears to be two or three times as large. Scholz, et al (1970)

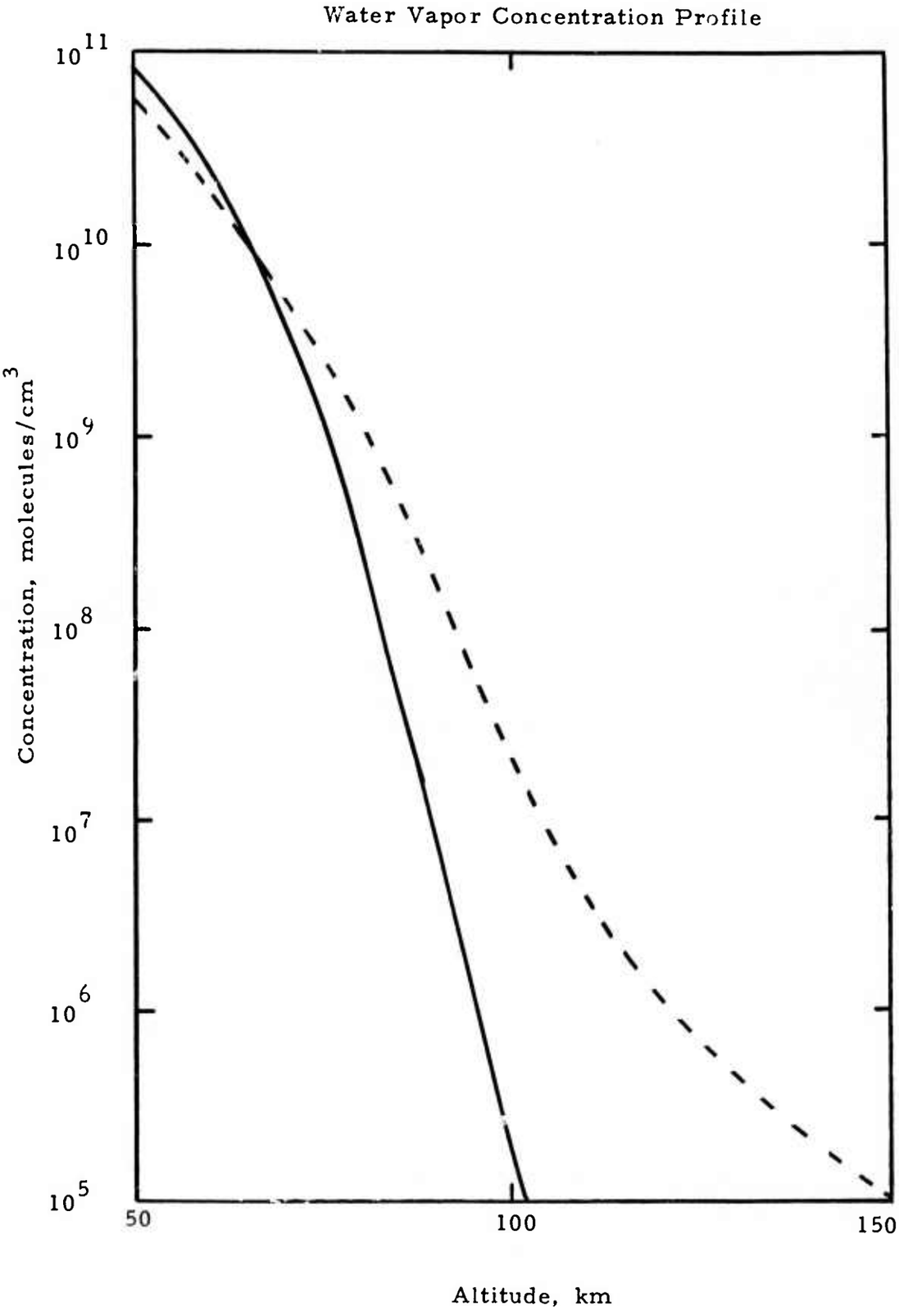
obtained comparable results in a direct sampling from a rocket near 50 km. Hunten and Stroebel (1974) have reviewed measurements and computations. They present a model which assumes a mixing ratio of four parts per million at 30 km. This includes the water producer by oxidation of methane in the stratosphere. Figure III-1 shows the results of their computation (solid line) and the water vapor profile used by Degges (1972), (dashed line). Hunten and Strobel (1974) show that their results are relatively insensitive to variations in the eddy mixing processes. Except for very large changes in the mixing ratio at the 30 km level, water vapor concentrations at all altitudes will vary directly with the low altitude mixing ratio. They find that better agreement can be obtained with Anderson's (1971 a, b) measurements if half as much water is assumed to be present in the atmosphere.

Figure III-2 shows a typical nitric oxide concentration curve, based on mid-latitude measurements.

While ozone abundances below about 50 km must take into account lateral transport as well as vertical transport by eddy mixing and molecular diffusion, photochemical models are adequate above that altitude, and agree with observations obtained by measuring absorption of solar and stellar ultraviolet radiation and those deduced from measurements of the molecular oxygen 1.27 μm infrared atmospheric band. The exact profile depends on assumed eddy diffusion coefficients, however, principally through atomic hydrogen and oxygen concentration. O_3 abundances used here are shown in Figure III-3.

Carbon dioxide has a lifetime of about 10 days at the top of the atmosphere, where it is dissociated by solar radiation. It is well shielded by molecular oxygen below about 120 km, however, and should remain well mixed below 100 km. The only good measurement

FIGURE III-1


Water Vapor Concentration Profile
10^{11}
10^{10}
10^{9}
10^{8}
10^{7}
10^{6}
10^{5}
Concentration, molecules/cm^3
50
100
150
Altitude, km

Nitric Oxide Concentration Profiles

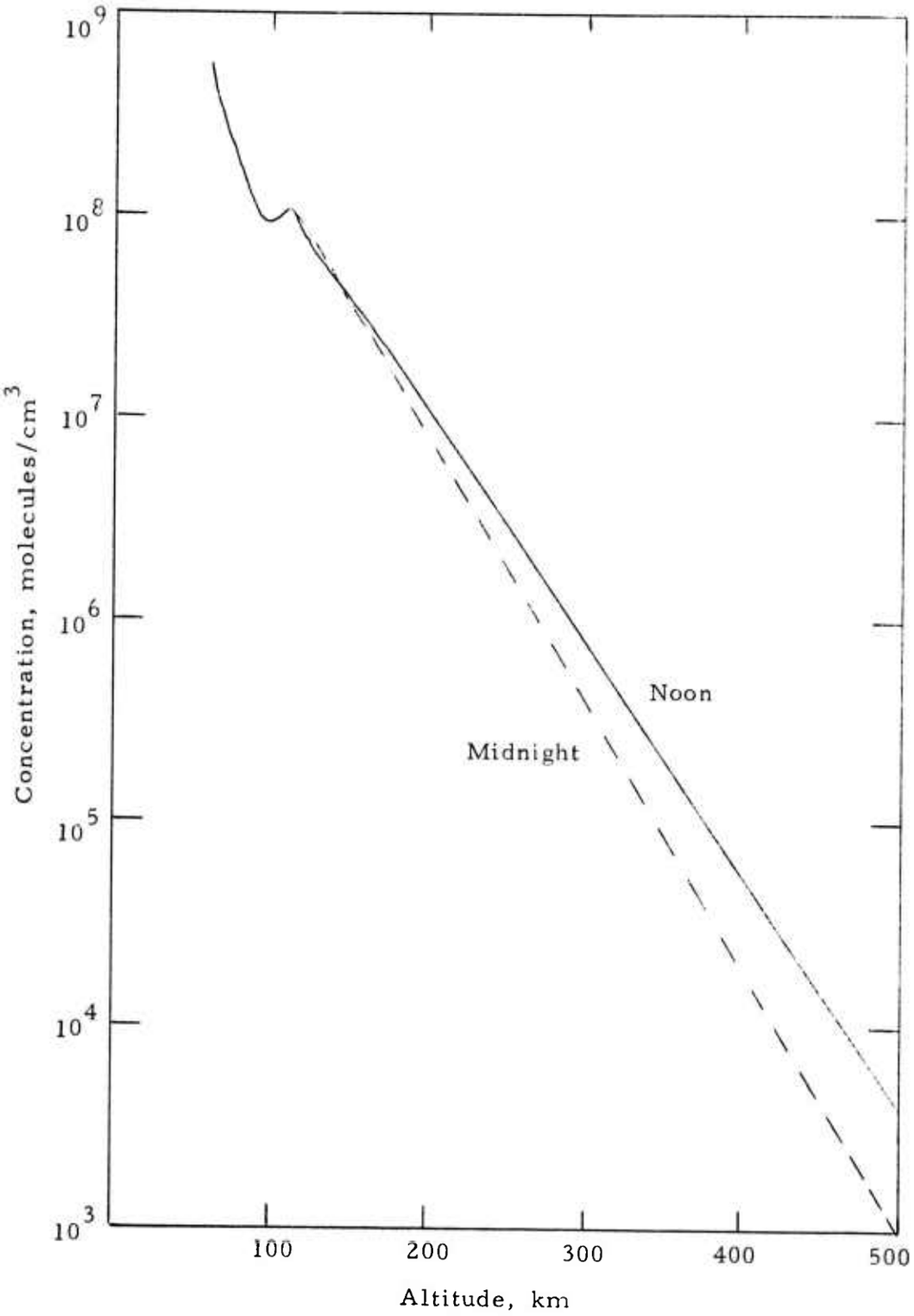


FIGURE III-2

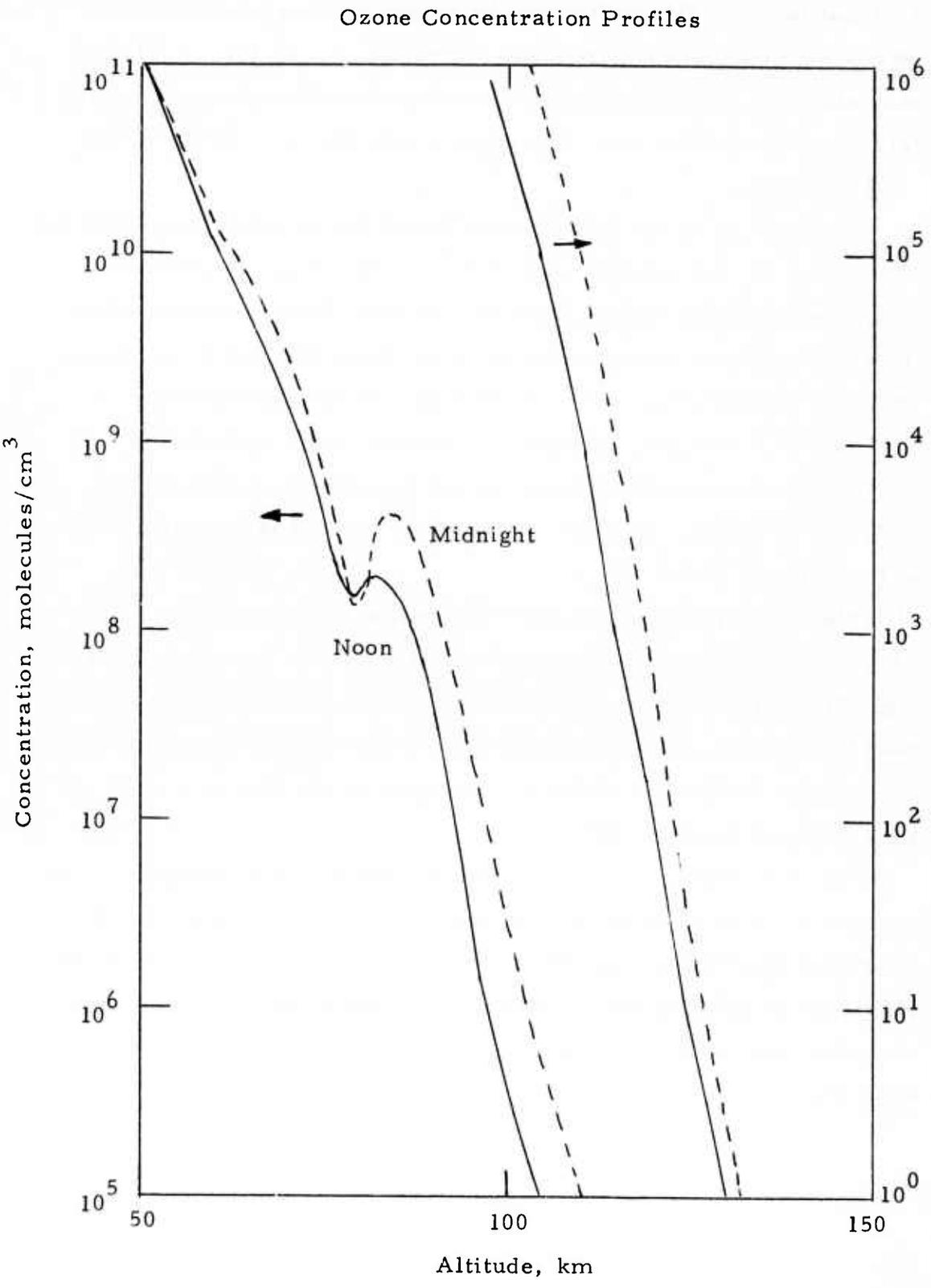

Ozone Concentration Profiles
Concentration, molecules/cm^3
10^{11}
10^{10}
10^9
10^8
10^7
10^6
10^5
10^6
10^5
10^4
10^3
10^2
10^1
10^0
Midnight
Noon
50
100
150
Altitude, km

FIGURE III-3

above 100 km remains that of Offerman and von Zahn (1971), who find that between 120 and 140 km the mixing ratio is not more than 50 per cent less than the tropospheric value. Above 140 km, molecular diffusion is rapid enough compared to photodissociation to maintain diffusive equilibrium. Figure III-4 is a typical estimate of CO_2 concentrations.

Recent work by Liu and Donahue (1974) has resulted in a model for H_2O, CH_4, H_2 and odd hydrogen that properly relates the measured mixing ratios in the stratosphere to escape of atomic hydrogen from the atmosphere by the mechanisms of the Jeans thermal flux, charge exchange and the polar wind. Their water vapor concentrations derived for a 1000 K exospheric temperature differ from that of Figure III-1 by less than a factor of two for altitudes below 100 km. At higher altitudes, their values become somewhat larger than those of the drawing.

The implication is that, since the chemistry program used here does not include the proper boundary conditions for hydrogen escape at high altitudes, long time calculations for hydrogen containing species may be in error. Future modifications to the chemistry program should include a provision for adding the hydrogen escape flux as part of the high altitude boundary conditions.

The chemistry program is described and listed in Appendix A, in almost the same form as used by Degges (1972). In addition to modification of the boundary conditions, it may be possible to improve the efficiency of the integration method. The remainder of this section describes the results of an investigation of a modification of the basic algorithm.

Carbon Dioxide Concentration Profile

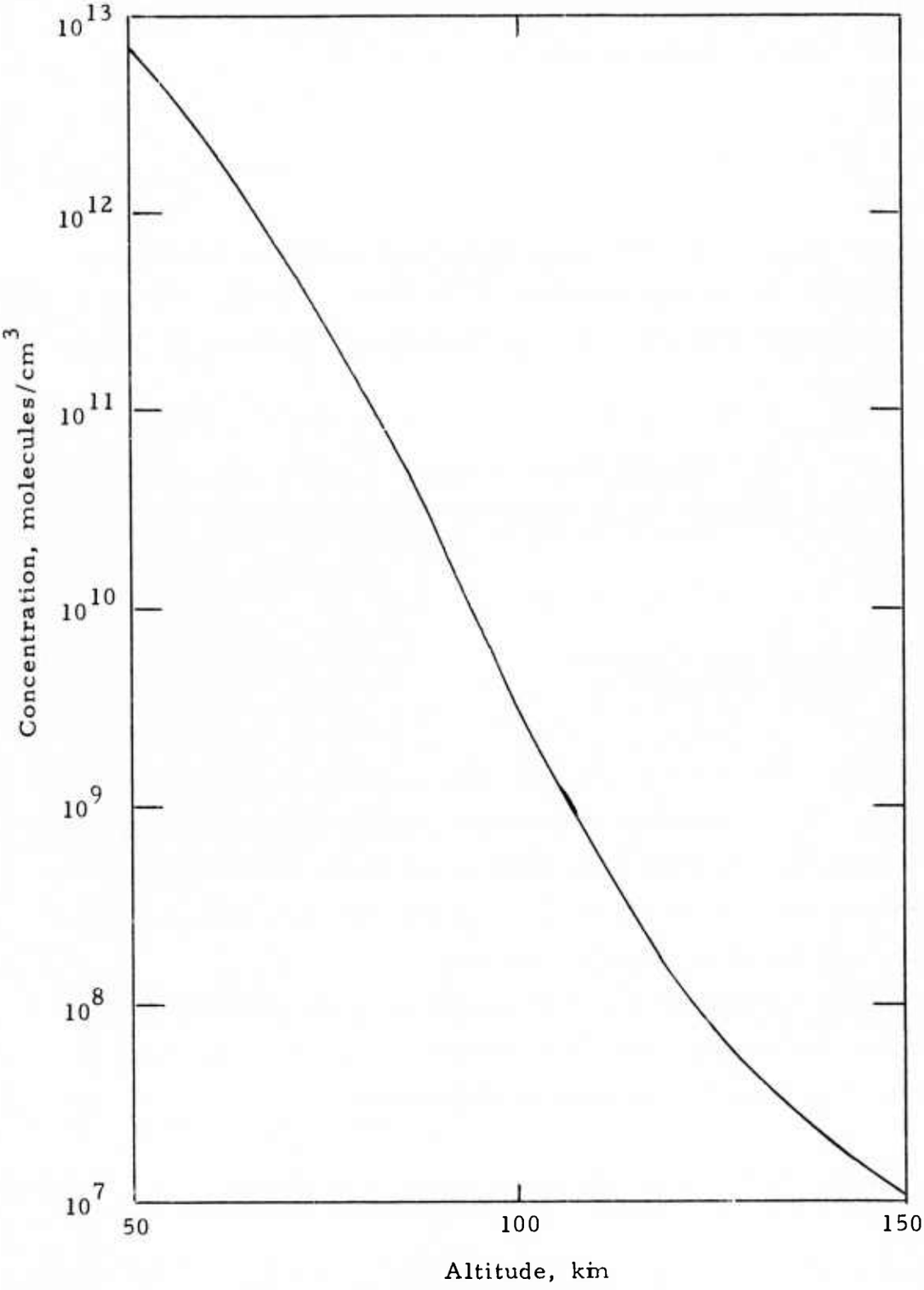


FIGURE III-4

The numerical method used to integrate the chemical rate equations in the chemistry program assumes that the rate equation for a single molecular species may be written

$$\frac{dx}{dt} = q - \beta x - \alpha x^2 \tag{3-4}$$

where q, β, and α are time dependent functions of the rate coefficients and the concentrations of the other molecular species in the system. The analytical solution of equation (3-4) for q, β, and α constant,

$$x(t) = \frac{[(\beta + \sigma)\exp(-\sigma t) + (\sigma - \beta)]x_o + 2q[1 - \exp(-\sigma t)]}{(2\alpha x_o + \beta)[1 - \exp(-\sigma t)] + \sigma[1 + \exp(-\sigma t)]} \tag{3-5}$$

where

$$\sigma = (\beta^2 + 4\alpha q)^{1/2}$$

is applied within the framework of the second order improved tangent Runge - Kutta method as described by Degges (1972). This algorithm is stable for any value of the time t and avoids the necessity of assumptions such as that some reactants with extremely short time constants are in chemical equilibrium.

When the product σt is large and any of the quantities q, β, and α are changing with time, however, true time dependence is lost. For large σt, equation (3-5) becomes

$$x(t) = \frac{\sigma - \beta}{2\alpha} = \frac{2q}{\sigma + \beta}, \quad \sigma t \text{ large} \tag{3-6}$$

and the value computed for the time t actually corresponds to time $t/2$ when Equation (3-5) is embedded in any second order Runge - Kutta method. In effect, the basic step is only first order for molecular species for which σt is large. The extrapolation algorithm does converge to a value for the concentrations of such components, but the process may require more steps than if the basic step were truly second order, as it is for small values of σt, or when q, β, and are constant.

Modifications of the algorithm has been attempted as follows. Consider the differential equation

$$\frac{dx}{dt} = q - \beta x \tag{3-7}$$

which is Equation (3-4) with the x^2 term absent. The formal solution for q and β constant is

$$x(t) = x \exp(-\beta t) + (q/\beta)[1 - \exp(-\beta t)] \tag{3-8}$$

If q and β are functions of time,

$$q = q_o + q_1 t + q_2 t^2 + \ldots\ldots \tag{3-9a}$$

$$\beta = \beta_o + \beta_1 t + \beta_2 t^2 + \ldots\ldots \tag{3-9b}$$

The improved tangent method applied over an interval h results in:

$$x(h/2) = x_o + (h/2)(q_o - \beta_o x_o) \tag{3-10}$$

$$\frac{dx}{dt} = q_o + (h/2)q_1 + (h/2)\, q_2 + \ldots\ldots$$

$$- [\beta_o + (h/2)\beta_1 + (h/2)^2\beta_2 + \ldots][x_o + (h/2)(q_o + \beta_o x_o)] \qquad (3\text{-}11)$$

and

$$x(h) = x_o + h \left.\frac{dx}{dt}\right|_{h/2}$$

$$= x_o + hq_o + (h^2/2)q_1 - h\beta_o x_o - (h^2/2)\; {}_1x_o$$

$$- (h^2/2)\; {}_o{}^2x_o + \text{terms in } h^3,\ h^4,\ \ldots \qquad (3\text{-}12)$$

which is correct through in terms in h^2.

When Equation (3-8) is embedded in the improved tangent method, the result is:

$$x(h/2) = x_o \exp(-\beta_o t) + (q_o/\beta_o)[1 - \exp(-\beta_o t)] \qquad (3\text{-}13)$$

$$\left.\frac{dx}{dt}\right|_{h/2} = q_o + (h/2)q_1 + (h/2)^2 q^2 + \ldots\ldots \qquad (3\text{-}14)$$

$$- [\beta_o + (h/2)\beta_1 + (h/2)^2\beta_2 + \ldots.] \times (h/2)$$

and

$$x(h) = x_o \exp[-h\beta_o - (h/2)\beta_1 - \ldots.] \qquad (3\text{-}15)$$

$$+ \frac{q_o + (h/2)q_1 + \ldots.}{\beta_o + (h/2)\beta_1 + ..}\; 1 - \exp[-h\beta_o(h/2)\beta_1 - \ldots]$$

A series expansion of Equation (3-15) is identical to Equation (3-12) through terms in second order. The higher order terms are different, but the method is still second order. For large βh, however, Equation (3-15) becomes

$$x(h) \simeq \frac{q_o + (h/2)q_1 + (h/2)^2 q_2 + \ldots}{\beta_o + (h/2)\beta_1 + (h/2)^2 \beta_2 + \ldots} \tag{3-16}$$

and not the correct result

$$x(h) \simeq \frac{q_o + hq_1 + h^2 q_2 + \ldots .}{\beta_o + h\beta_1 + h^2 \beta_2 + \ldots .} \tag{3-17}$$

When βh is large, the algorithm may be altered as follows to retain second order accuracy and obtain a better approximation to the endpoint value. Let:

$$x = q/\beta + \phi$$

Then

$$\frac{d\phi}{dt} = -\frac{d}{dt}\left(\frac{q}{b}\right) - \beta\phi \tag{3-18}$$

$$= \phi - \beta\phi$$

and

$$\phi = \phi_o \exp(-\beta t) + (\phi/\beta)[1 - \exp(-\beta t)] \tag{3-19}$$

further, let

$$\phi = \phi/\beta + \Psi$$

then

$$\frac{d\Psi}{dt} = -\frac{d}{dt}\left(\frac{\phi}{\beta}\right) - \beta\Psi \qquad (3\text{-}20)$$

$$= R - \beta\Psi$$

and

$$\Psi = \Psi_o \exp(-\beta t) + (R/\beta)[1 - \exp(-\beta t)] \qquad (3\text{-}21)$$

Only terms in q_o, q_1, β_o, and β_1 enter into the second order terms in the solution of Equation (3-7) given in Equation (3-12). If q and β are assumed linear in the interval of integration h, $q_o = q(t = 0)$, $\beta_o = \beta(t = 0)$, $q_1 = 2[q(h/2) - q_o]/h$, $\beta_1 = 2[\ (h/2) - \beta_o]/h$, and if $\phi_o = x_o - q_o/\beta_o$,

$$x(h/2) = q_o/\beta_o + \phi_o \exp(-\beta_o h/2) \qquad (3\text{-}22)$$

Then if $\Psi_o = \phi_o - \dfrac{\beta_1 q(h/2) - q_1 \beta_o(h/2)}{\beta^2(h/2)}$ and the term R is neglected, $\Psi(h) = \Psi_o \exp[-h\beta(h/2)]$ and the resulting expression

$$x(h) = \frac{q_o + hq_1}{\beta_o + h\beta_1} + \frac{\beta_1 q_o - \beta_o q_1}{(\beta_o + h\beta_1/2)^3}\left[1 - \exp(-h\beta_o - h^2\beta_1/2)\right]$$

$$+ \left(x_o - \frac{q_o}{\beta_o}\right) \exp\left(-h\beta_o - h^2\beta_1/2\right) \qquad (3\text{-}23)$$

when expanded in series, agrees through terms in h^2 with Equations (3-12).

The same analysis may be applied to Equation (3-4). In this case the appropriate substitution is;

$$x = \frac{2q}{\sigma + b} + \phi \quad \text{and}$$

$$\phi = -\frac{d}{dt}\frac{2\sigma}{\sigma + b} - \sigma\phi - \alpha\phi^2$$

$$= \phi - \sigma\phi - \alpha\phi^2 \qquad (3\text{-}24)$$

setting

$\Delta^2 = \sigma^2 + 4\alpha\phi$, make the further substitution

$$\phi = \Psi + \frac{2\phi}{\Delta + \sigma}$$

to obtain

$$\Psi' = -\frac{d}{dt}\left(\frac{2\phi}{\Delta + \beta}\right) - \Delta\Psi - \alpha\Psi^2$$

$$= R - \Delta\Psi - \alpha\Psi^2 \qquad (3\text{-}25)$$

Again neglecting the R term in Equation (3-25), the method becomes

$$x(h/2) = \frac{2q_o}{\sigma_o + \beta_o} + \frac{\sigma_o \phi_o \exp(-\sigma_o h/2)}{\alpha_o[1 - \exp(-\sigma_o h/2)]\phi_o} + \sigma_o \qquad (3\text{-}26)$$

The $\alpha_o, \alpha_1, \beta_o, \beta_1, q_o, q_1$, and Ψ_o are evaluated as before from the values available at t = 0 and t = h/2 and;

$$x(h) = \frac{2q(h)}{\sigma(h) + \beta(h)} + \frac{2\phi(h/2)}{\Delta(h/2) + \sigma(h/2)} + \frac{\Delta(h/2)\Psi_o \exp[-h\Delta(h/2)]}{\alpha(h/2\{1-\exp[-h\Delta(h/2)\Psi_o\}+\Delta(h/2)} \qquad (3\text{-}27a)$$

When Δ^2 is negative, the solution must be modified to

$$x(h) = \frac{2q(h)}{\sigma(h)+\beta(h)} + \frac{\Delta(h/2)\tan\left\{\arctan[2\alpha(h/2)\Psi_o+\delta(h/2)/\Delta(h/2)]-1/2\,\Delta(h/2)t\right\}-\sigma(h/2)}{2\alpha(h/2)} \qquad (3\text{-}27b)$$

The best value of βh or σh to divide the region of applicability of the two methods was studied by sample computations, but results were not appreciably better than with the basic algorithm.

SECTION IV

RESULTS FROM COMPUTATIONAL MODEL

A. Introduction

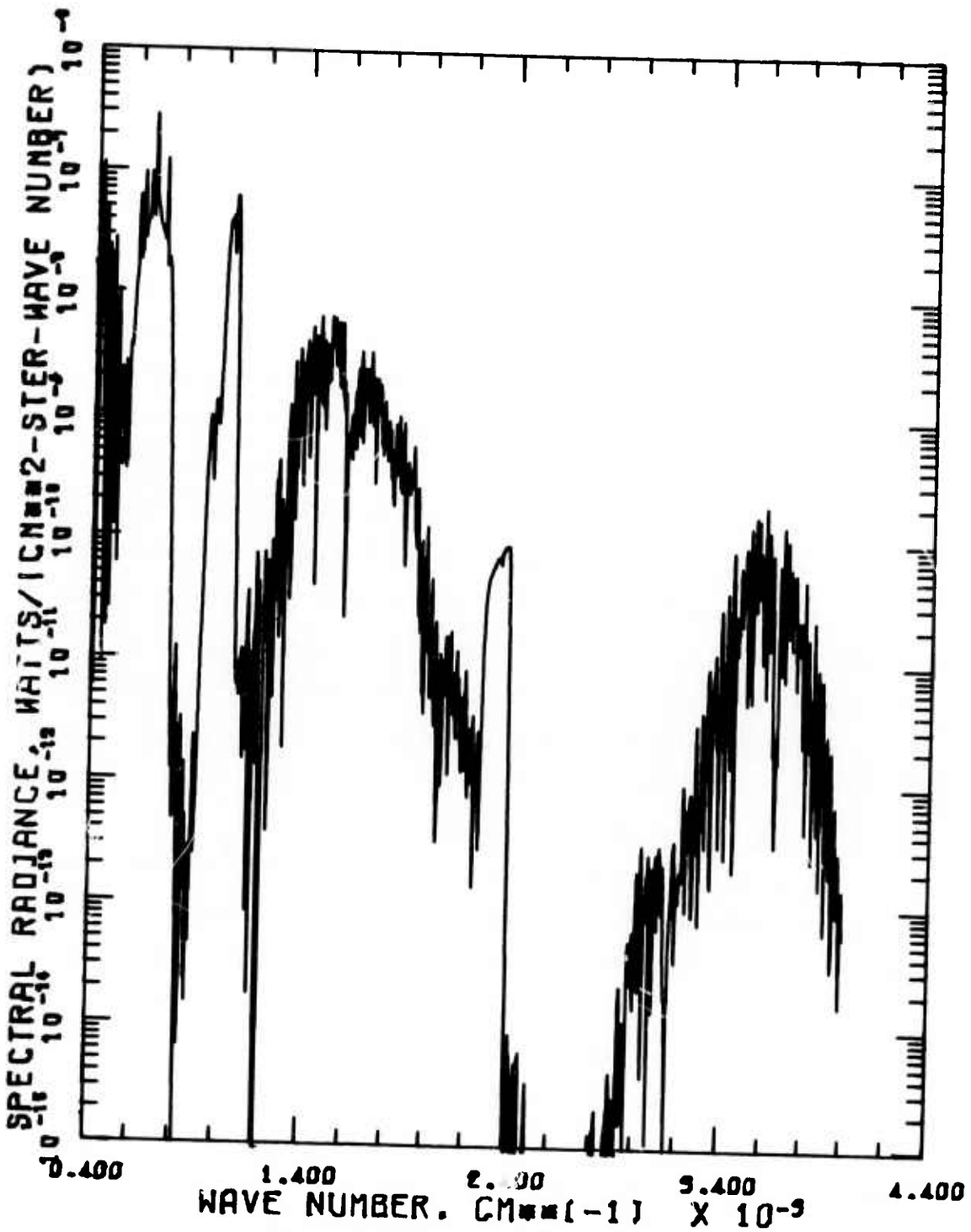
The revised high altitude spectral radiance programs can produce spectral radiance profiles for wavelengths between 2.5 and 25 micrometers or between 400 and 4000 wave numbers. Results are presented here based on estimated abundance profiles of the four species CO_2, H_2O, O_3 and NO. The temperature and pressure profiles used were taken from the U.S. Standard Atmosphere, 1962. Upwelling terrestrial and lower atmosphere radiances are assumed to be those corresponding to a black body source at the 60 km temperature.

Spectral radiance plots are shown for typical tangent heights for day and night conditions. Features of the spectral radiance curves are explained with the aid of altitude profiles for total band radiances of individual vibrational-rotational bands.

B. Limb-Viewing Spectral Radiances

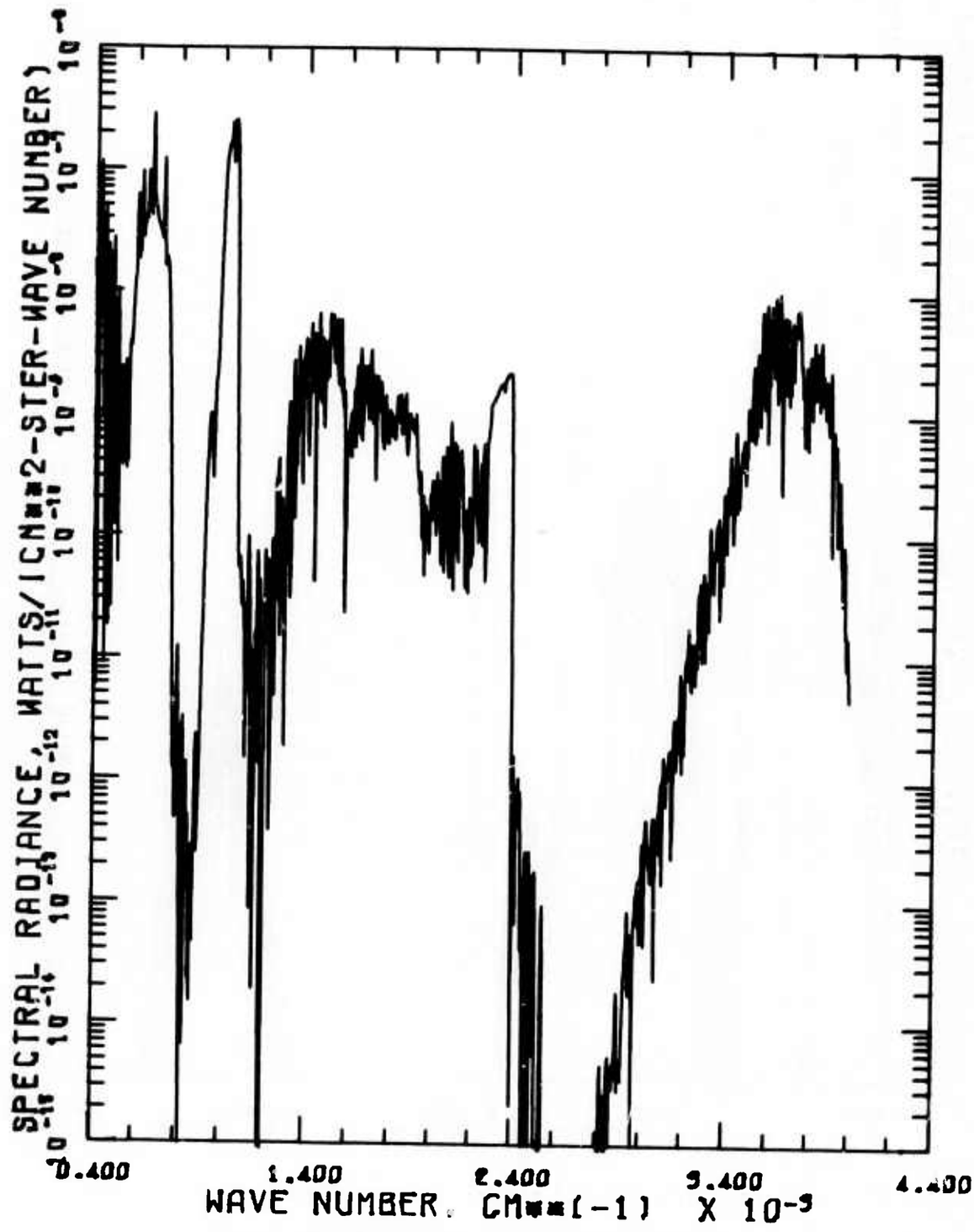
Representative limb viewing radiance profiles are shown in Figures IV-1 through IV-6. These are based on the concentration profiles of Section III, except that the experimental curve of Roble and Hays (1974) has been used for night time ozone and a constant mixing ratio has been used for CO_2. The temperatures and number densities used are those of the U.S. Standard Atmosphere, 1962.

The results shown are quite different from those of Degges (1972) for many of the carbon dioxide bands, where the earlier version of the program predicted appreciably lower radiance values at high altitudes. The night time ozone radiance



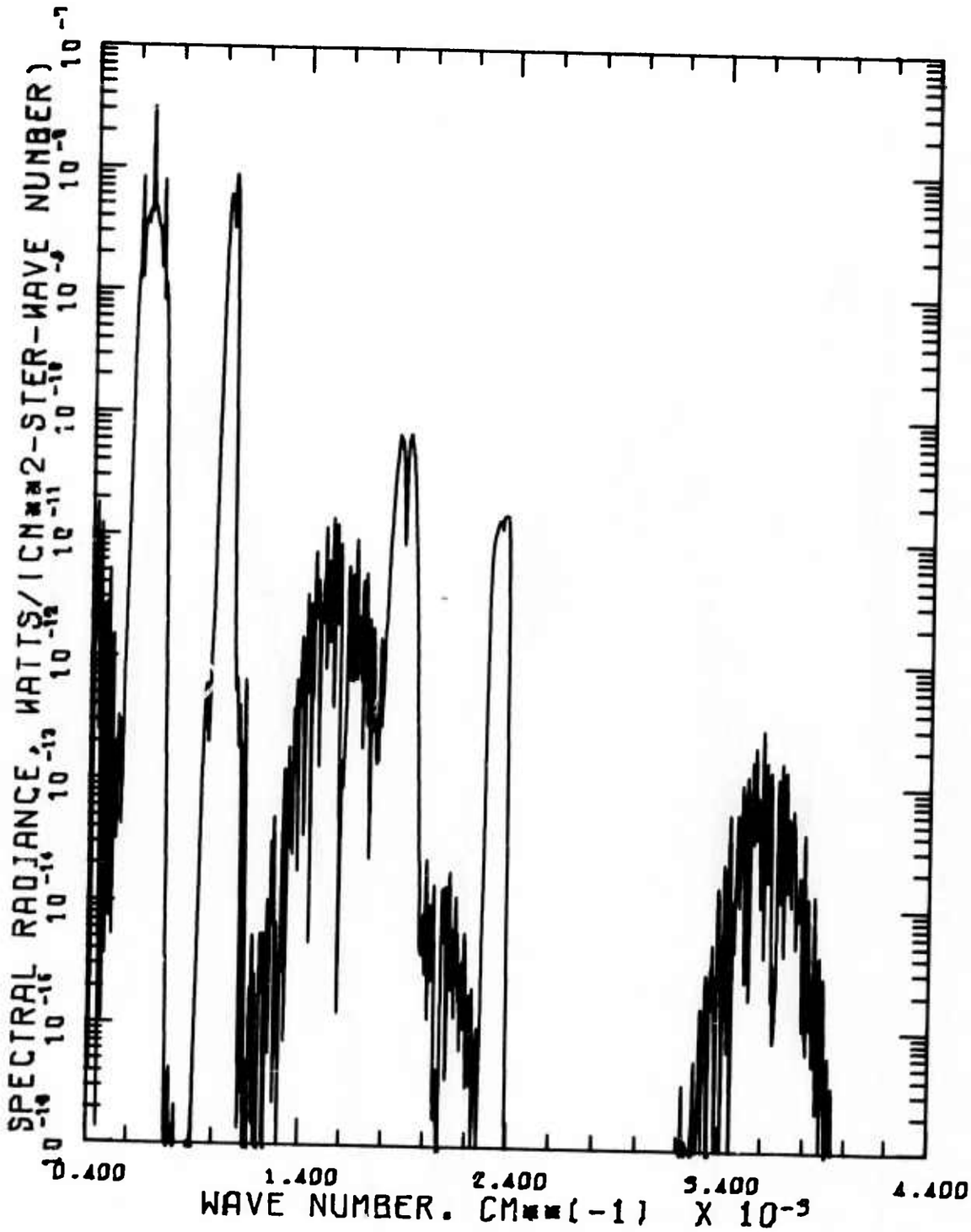
Night Limb Viewing Spectral Radiance,
400 to 4000 cm^{-1}, 60 km Tangent Height

FIGURE IV-1



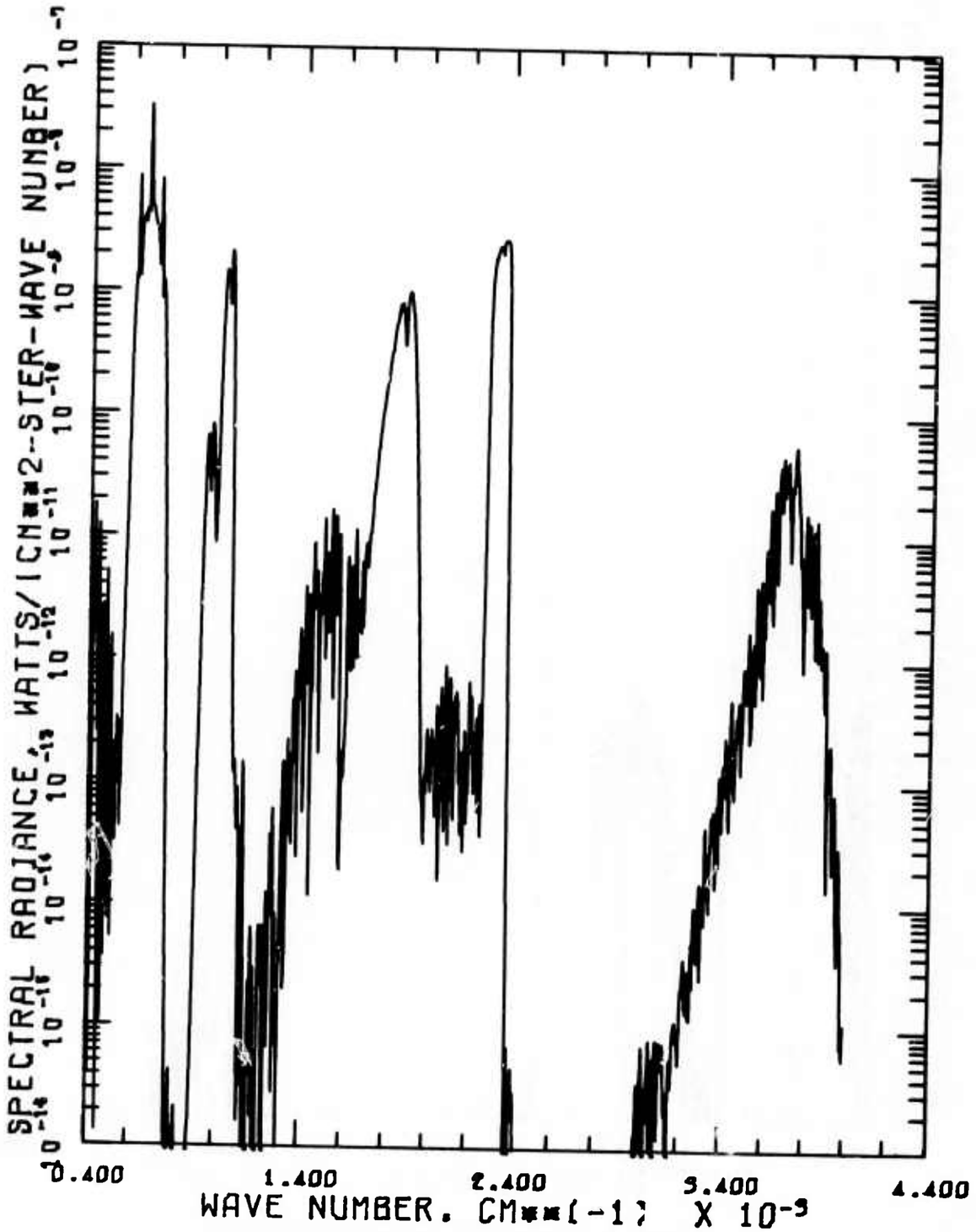
Day Limb Viewing Spectral Radiance,
400 to 4000 cm^{-1}, 60km Tangent Height.

FIGURE IV-2



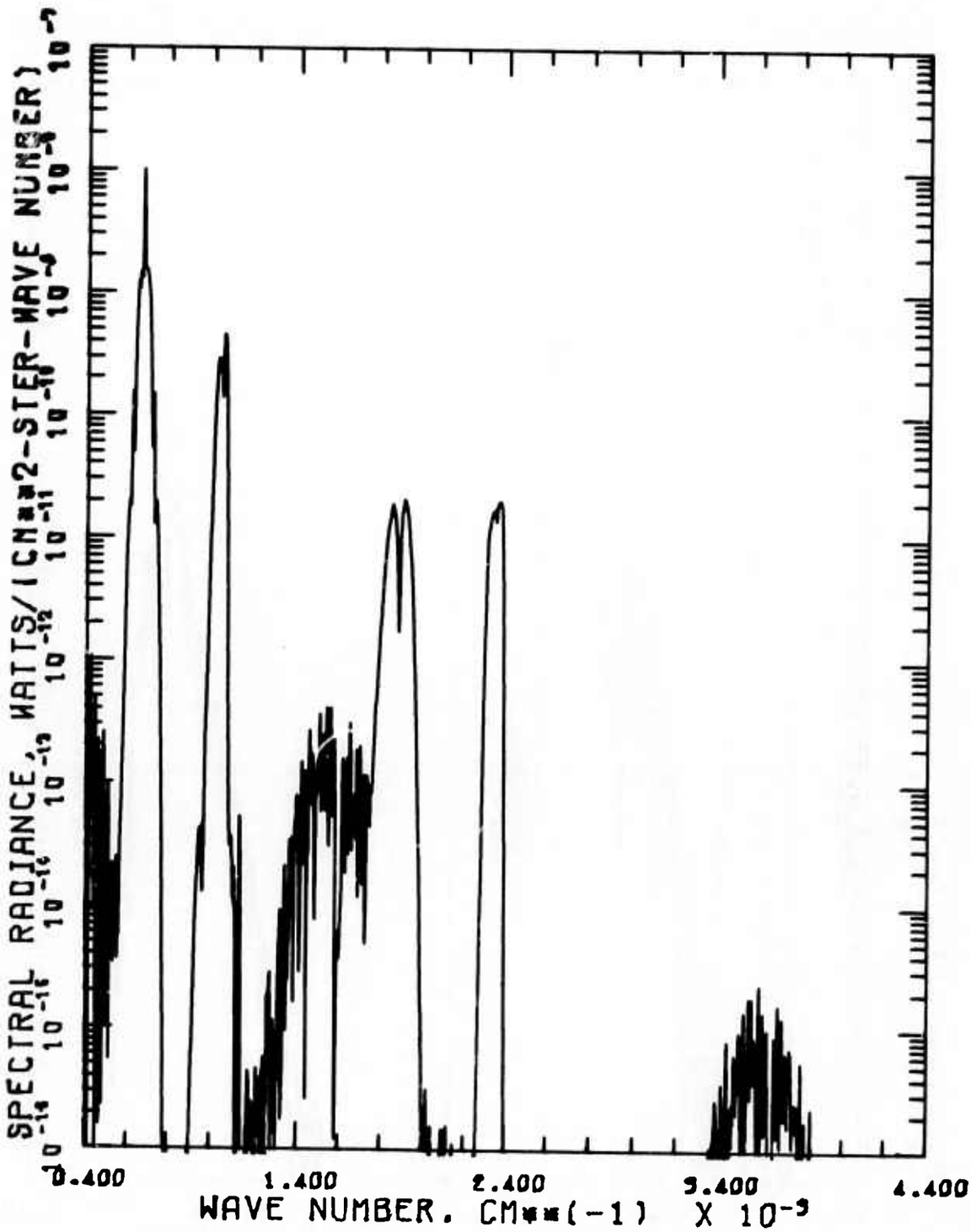
Night Limb Viewing Spectral Radiance,
400 to 4000 cm^{-1}, 80 km Tangent Height.

FIGURE IV-3



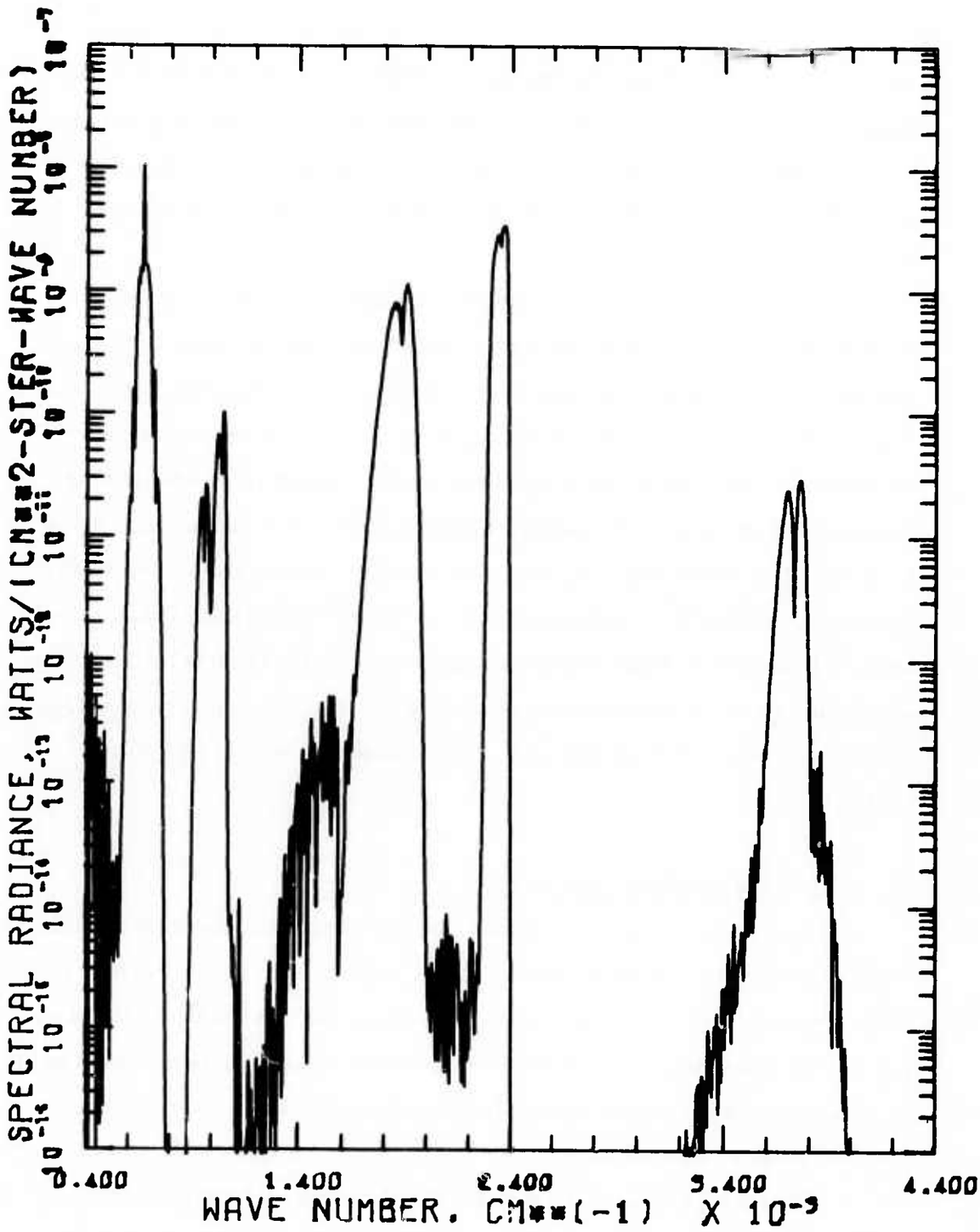
Day Limb Viewing Spectral Radiance,
400 to 4000 cm^{-1}, 80 km Tangent Height.

FIGURE IV-4



Night Limb Viewing Spectral Radiance,
400 to 4000 cm^{-1}, 100 km Tangent Height.

FIGURE IV-5



Day Lamb Viewing Spectral Radiance,
400 to 4000 cm^{-1}, 100 km Tangent Height.

FIGURE IV-6

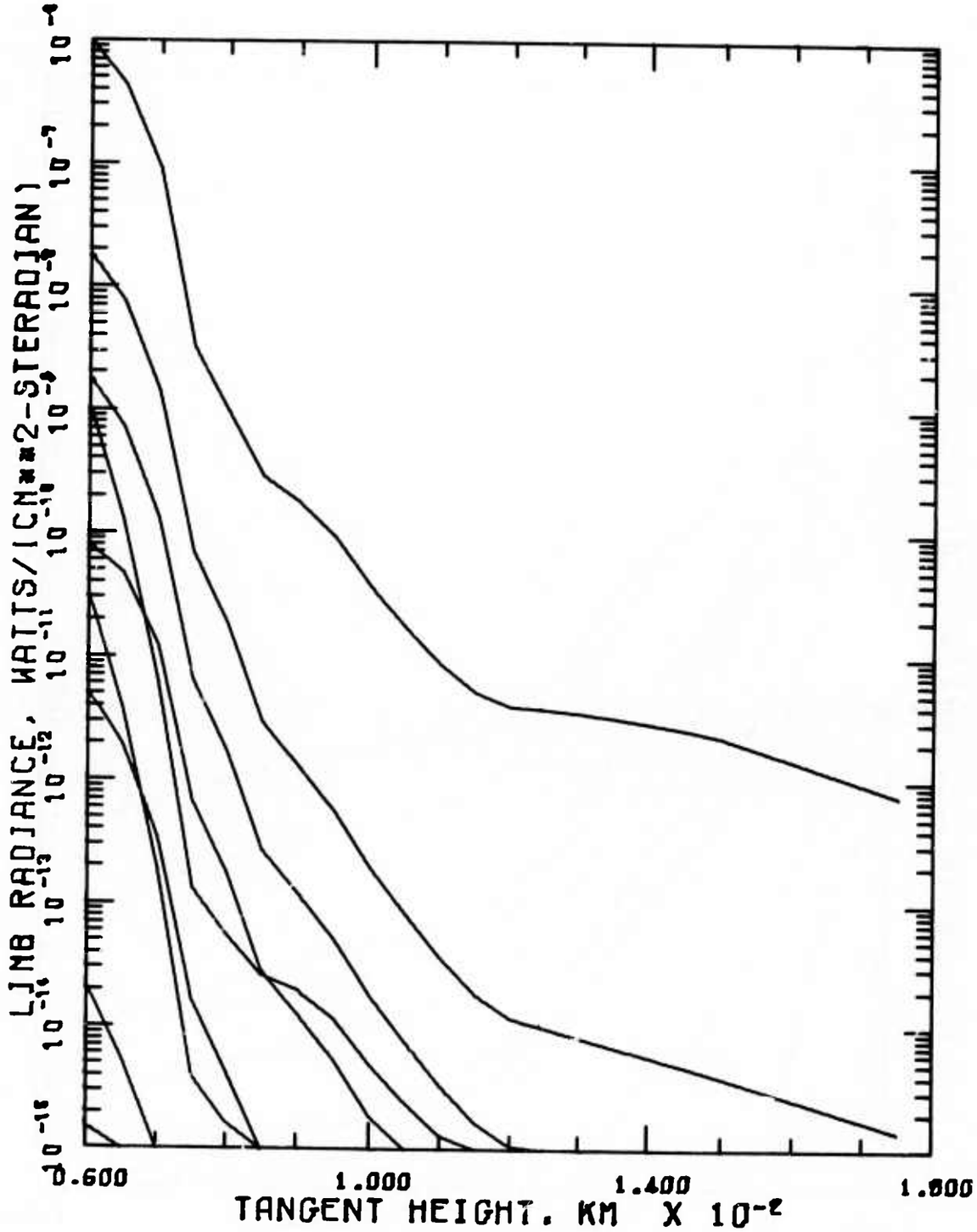
values are also higher in spite of a reduction in total column count (molecules/cm^2) above 60 km. The differences in the CO_2 values are primarily because the early program did not correctly integrate the radiation field produced by molecules radiating above 60 km.

Comparison of day and night profiles for a given tangent height shows that the low energy (small wave number and long wave length) band radiances do not vary appreciably during the day. The high altitude high energy bands, however, do vary in intensity by as much as two orders of magnitude as the result of resonant and fluorescent emission following absorption of solar radiation. This is shown clearly in the CO_2 4.3 micrometer bands and in the water vapor complex at 2.7 micrometers. A faint indication of this effect is visible in the 5.4 micrometer nitric oxide fundamental where the day time curve is broadened because of the fluorescent contribution from absorption of sunlight at 2.7 micrometers and in the ultraviolet.

C. Individual Band Radiances

For some purposes, it may be more convenient to have available the total radiance of an individual band rather than the spectrum of the region where the band is located. This may aid in making allowances for species abundances different from those assumed in a computation or in assigning an experimentally observed radiance value to one species rather than another.

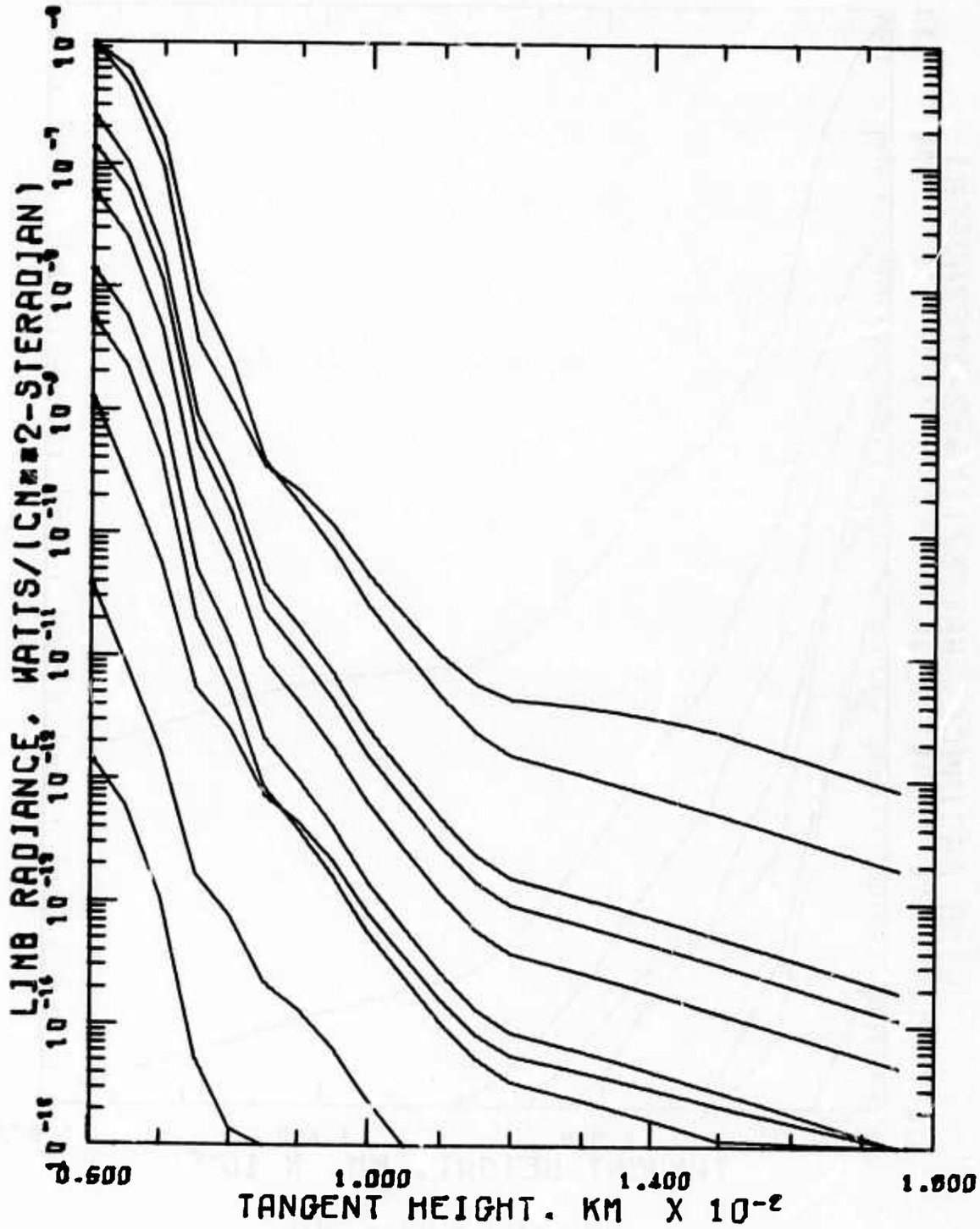
Figures IV-7 and IV-8 show the day and night total band radiances for the ten water vapor bands included in the spectra of Figures IV-1 through IV-6. The 6.3 micrometer ν_2 band is the



Night Limb Viewing Total

Band Radiance Values for Water Vapor Bands

FIGURE IV-7



Day Limb Viewing Total

Band Radiance Values for Water Vapor Bands

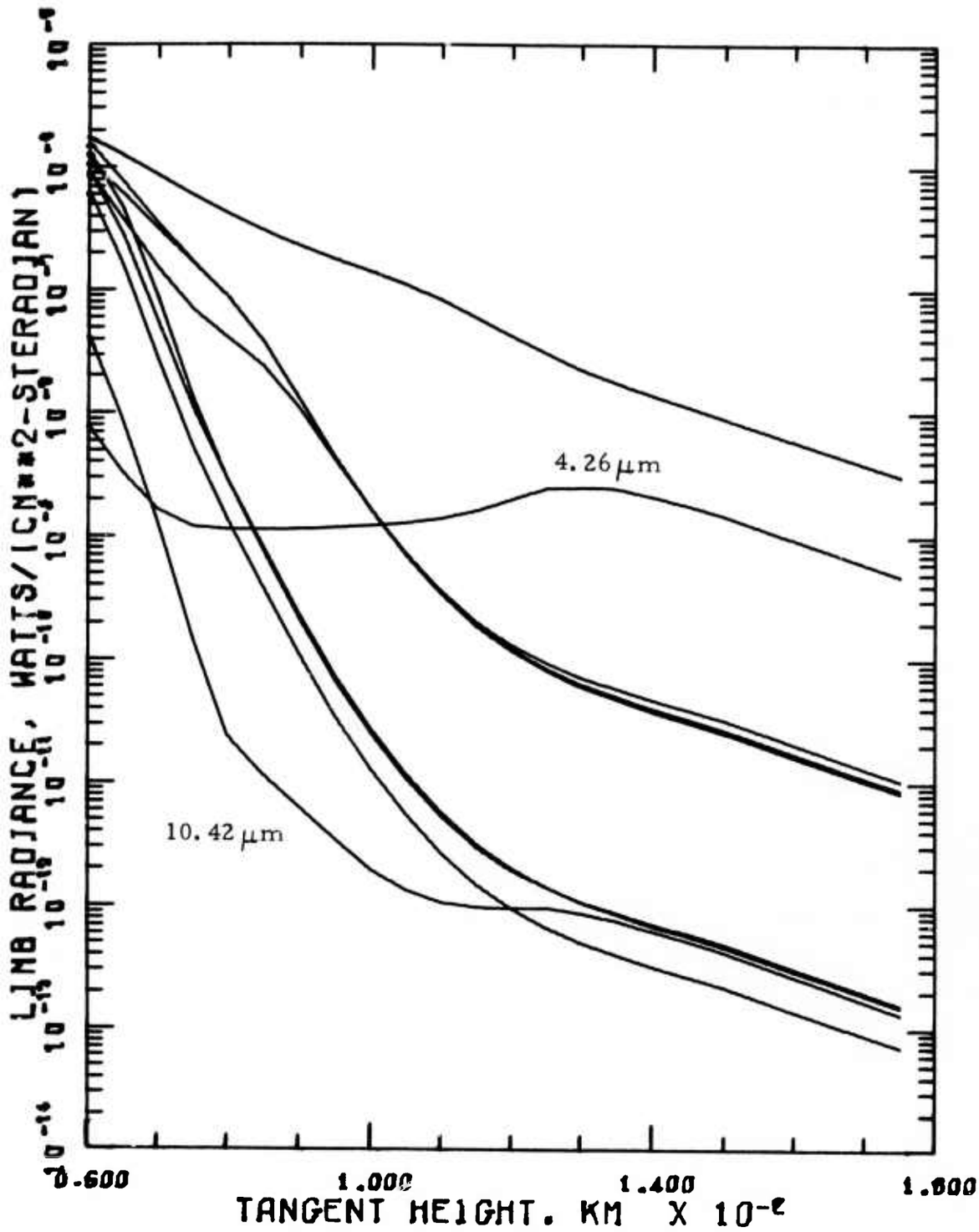
FIGURE IV-8

strongest, day or night. During the day, the ν_3 band at 2.66 micrometers strongly absorbs solar radiation and becomes almost as bright at a tangent height near 90 km. At night this band is nearly four orders of magnitude fainter.

Figures IV-9 and IV-10 show night and day limb radiance values for seven bands of the CO_2 15 micrometer band complex, the 4.3 micrometer ν_3 band and one of the two fluorescent bands originating from the ν_3 state, the 10.42 micrometer 00011 - 10001 Band. The 15 micrometer band radiances vary little between day and night. During the day, however, the radiance of the 4.3 micrometer band rises from its night value of between 10^{-9} and 10^{-8} watts/cm^2-ster to a value of between 10^{-8} and 2×10^{-7} watts/cm^2-ster. This band is thick in the limb viewing geometry, and most of the increase is due to increased vibrational temperatures at altitudes above 90 km. The 10.41 micrometer band, which has much lower optical thickness, gives a better indication of the altitudes at which the solar excitation is important. The 60 km value is about 4×10^{-8} watts/cm^2-ster day or night. Figure IV-11 shows night limb radiance values for the same CO_2 bands when the mesopause temperature is raised to 230 K. The main effect is to double or triple the band radiance values, indicating the importance of the temperature profile in computing radiance values.

D. Comparison with Observations

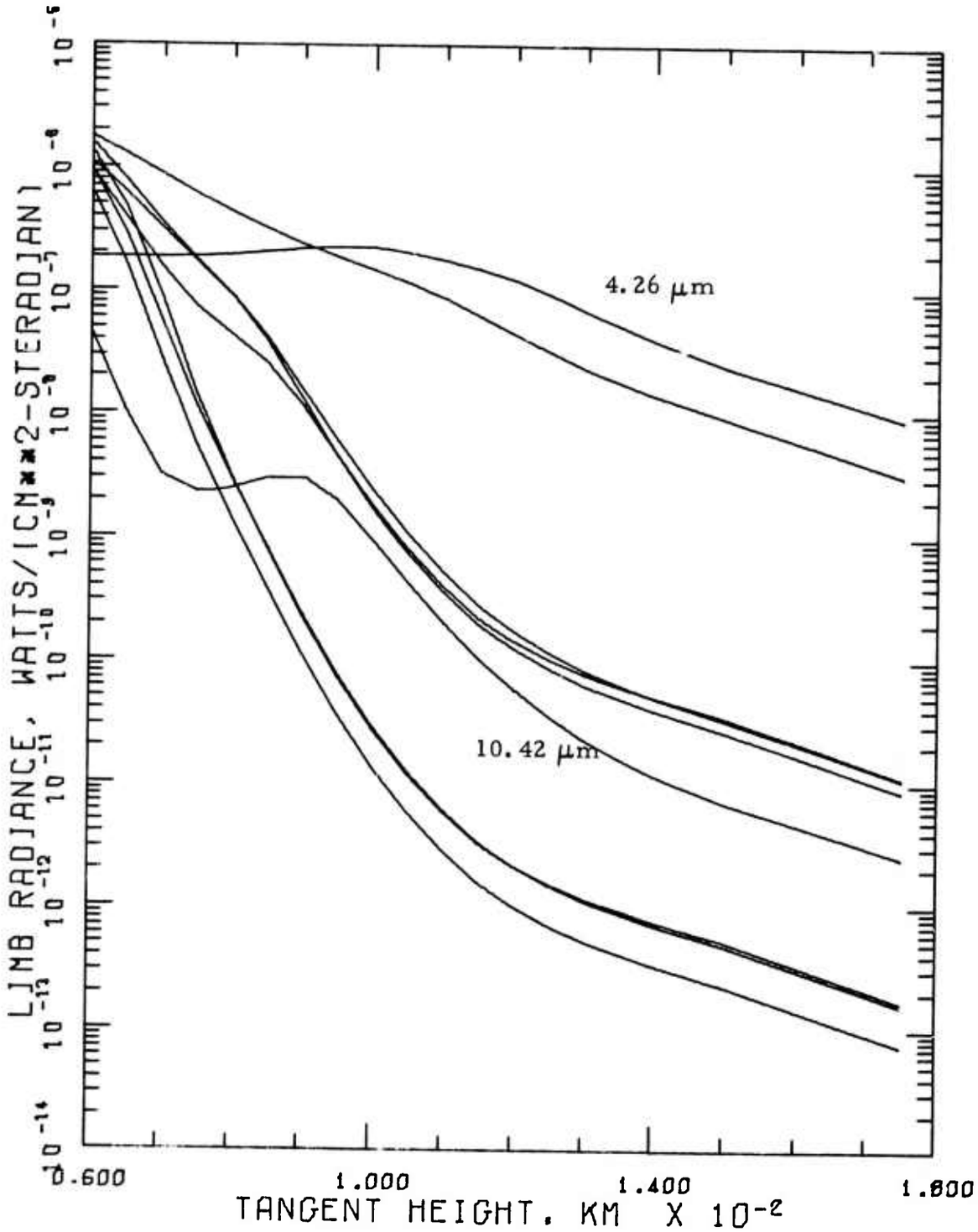
A limited amount of data from the ICECAP program is available for comparison with the results of the radiance computer program. Figure IV-12 compares peak spectral radiance values for the 15 micrometer band measured by Stair, et al (1974) with those



Nighttime Limb Radiance, CO_2

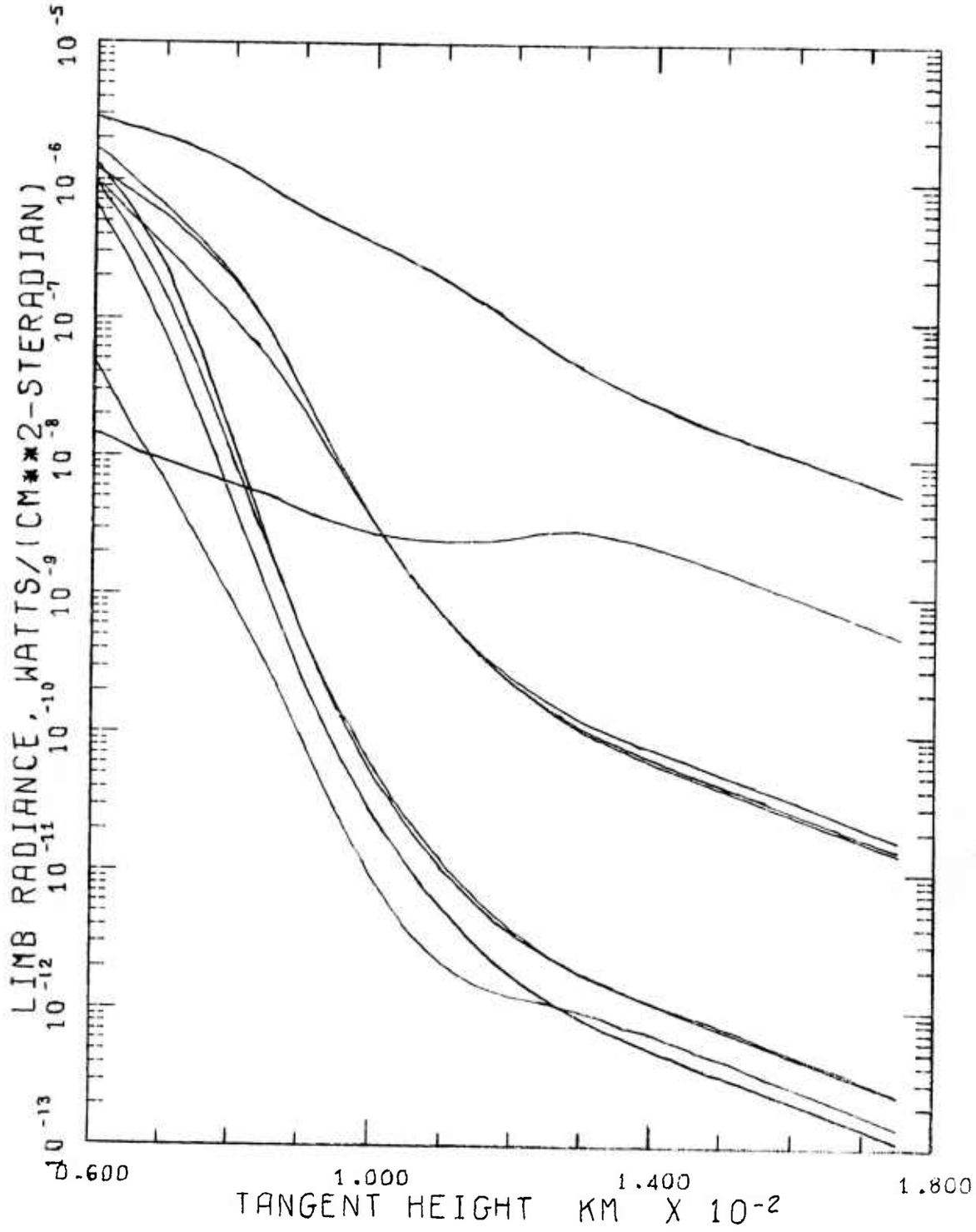
15 μm Bands, 10.42 μm Band and 4.26 μm Band.

FIGURE IV-9



Daytime Limb Radiance, CO_2
15 μm Bands, 10.42 μm Band and 4.26 μm Band.

FIGURE IV-10

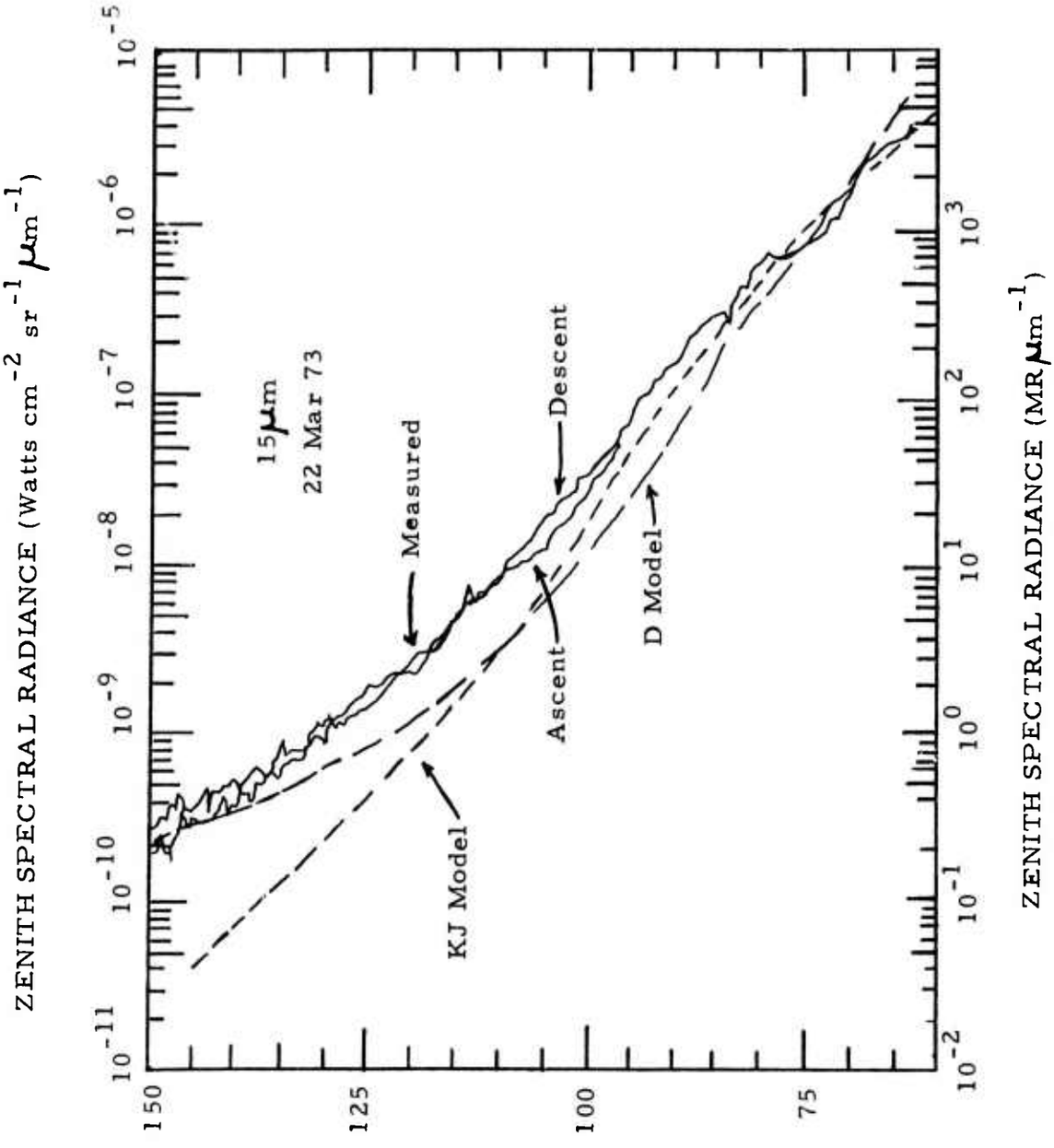


NIGHTTIME LIMB RADIANCE, CO_2

15 μm BANDS, 10.42 μm BAND and 4.26 μm BAND

230 K MESOPAUSE

FIGURE IV-11



Zenith peak spectral radiance at 15μm measured with LWIR spectrometer aboard rocket flown at Poker Flat, Alaska on 22 March 73 compared to theoretical model. The solid curve is the measured profiles (ascent and descent) while the dashed curves show model calculations of Kumer and James (KJ) and Corbin and Degges (D).

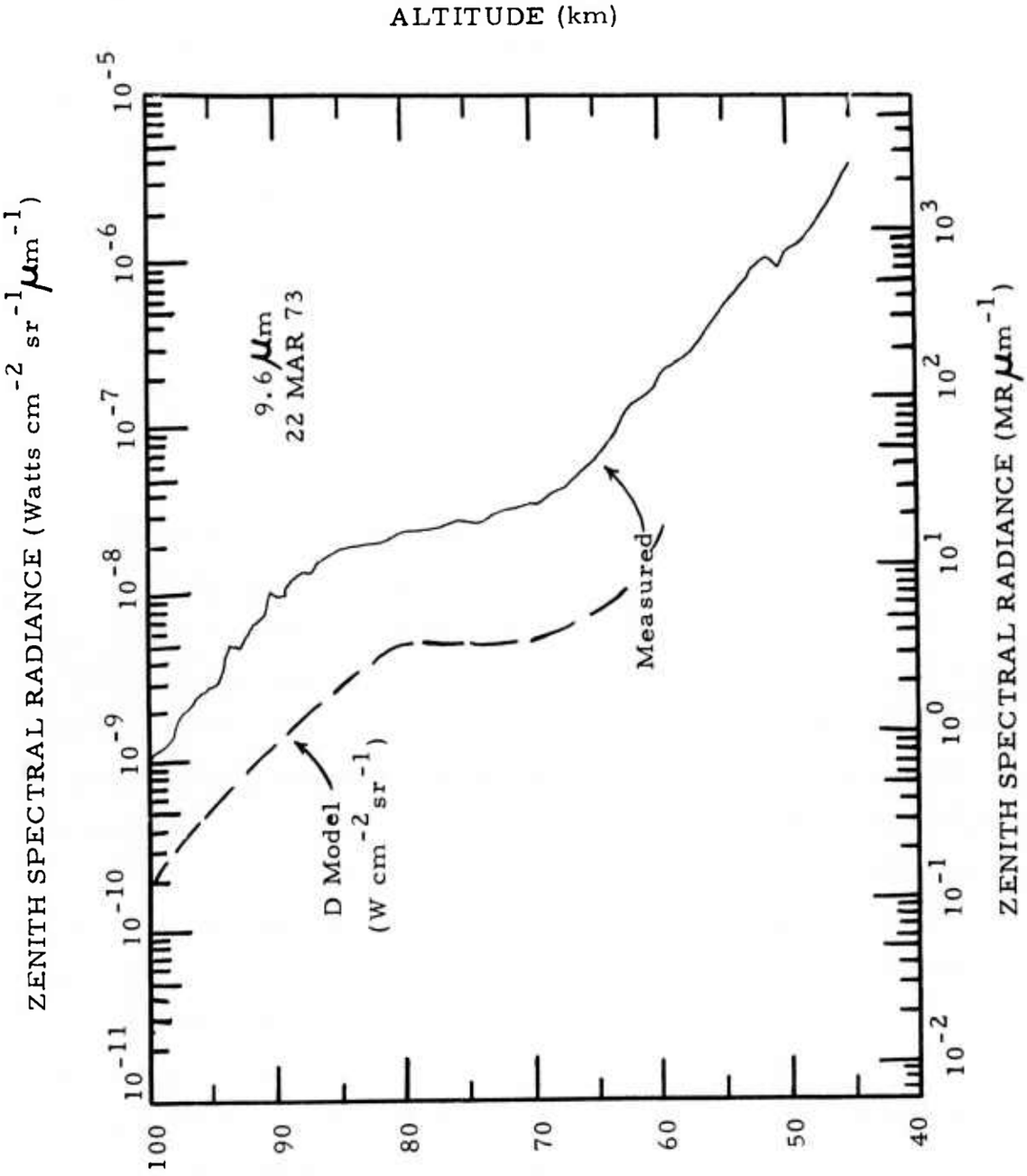
FIGURE IV-12

computed by the radiance program (D Model) using the U. S. Standard Atmosphere, 1962 temperature profile with 180 K mesopause temperature and that computed by J. B. Kumer (K. J. Model) using the Kumer and James (1973) model radiance program. The differences between the results of the two computer programs are due principally to differing temperature profiles and assumed CO_2 mixing ratios. The degree of agreement between the two models and the experimental data give an indication of what can be expected from use of the computer programs and the need to improve both the computational algorithms and knowledge of atmospheric composition.

Figure IV-13 compares the radiance model prediction with the data of Stair, et al (1974), this time for the ozone ν_3 band. The amount of ozone used for the calculation is optically thin in the upward direction, and doubling the amount would double the computed radiance. The effective width of the observed ozone band is about one-third micrometer, and the plotted experimental data agrees with the measurement to within a factor of two. The shape of the observed radiance curve supports the accuracy of the ozone measurement of Hays and Roble and their interpretation that their observed ultraviolet absorption is due to ozone and not a mesospheric aerosol layer. Figure IV-13, as well as Figure IV-12, shows that at least some results of the radiance program are in agreement with measured radiances, and suggests that with more accurate data on atmospheric temperature and composition, observations will either confirm model predictions or show how modeling algorithms must be modified.

E. Model Evaluation

The most important general areas of investigation in this research effort have been (a) the chemistry and spatial variation



Zenith peak spectral radiance at 9.6 μm measured with LWIR spectrometer aboard rocket flown at Poker Flat, Alaska on 22 March 73 compared to theoretical model. The solid curve is the measured profile while the dashed curve shows the D Model calculations in watts cm^{-2} ster^{-1}.

FIGURE IV-13

of atmospheric species important to infrared radiation and (b) the improvement of models for the transport of infrared radiation.

Section III dealt with atmospheric composition and modifications to the chemistry program written during a previous research effort (Degges, 1972). This program remains limited in its usefullness to short time extrapolation from a given set of atmospheric composition profiles. It can be used to give limited information on such problems as auroral chemistry, but requires knowledge of the limitations imposed by the limited chemistry set and uncertainties in global transport of such important species as atomic oxygen.

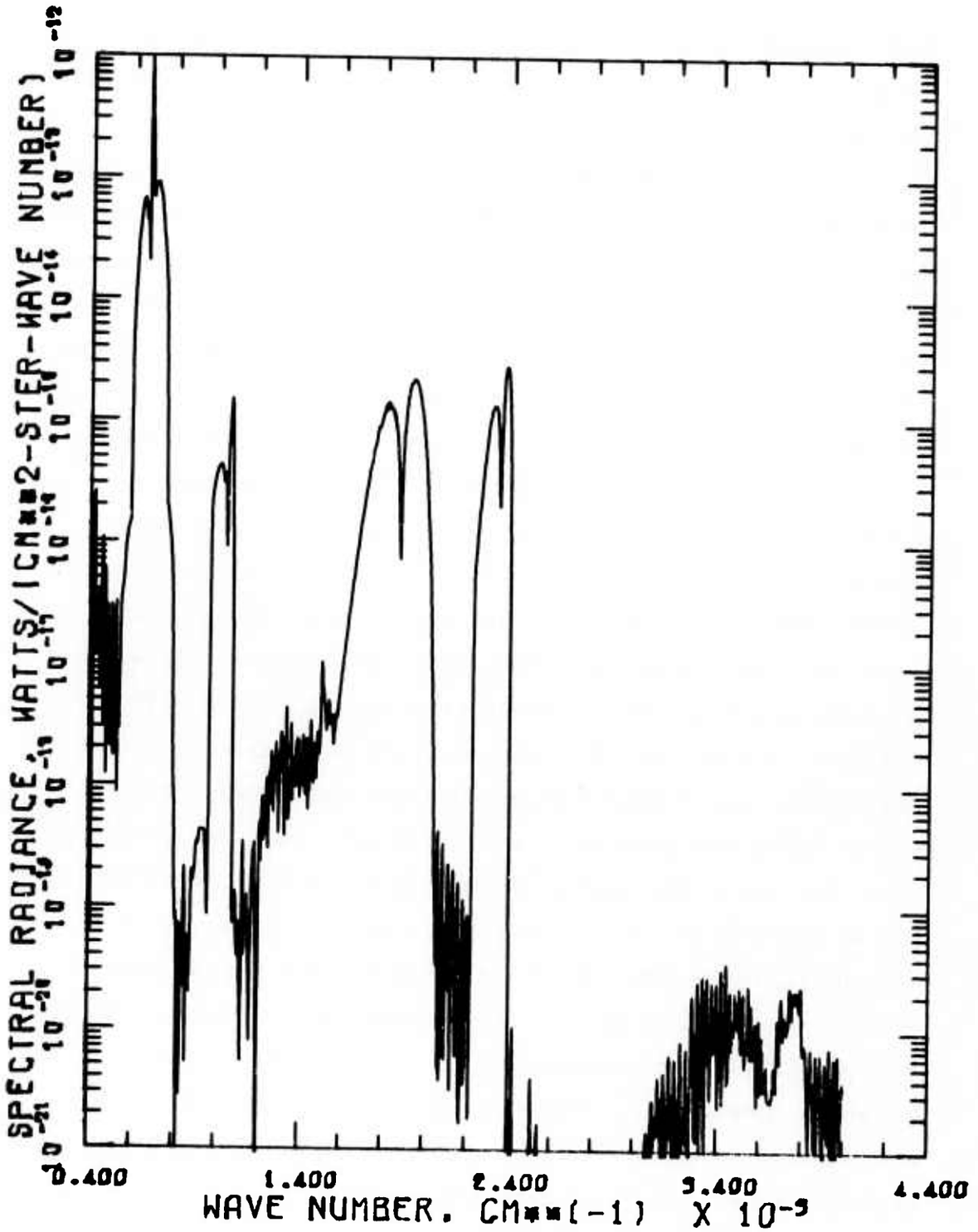
More important advances have been made in the radiative transport program. A serious programming error has been corrected, and an improved formulation of the transport problem has been coded. The program produces results which agree with a limited amount of observational data. The agreement with the carbon dioxide radiance model of James and Kumer (1973) is believed to be satisfactory. There remain areas in which results were not as good as those anticipated when the work was initiated.

The program for computing band radiances requires an excessive amount of computer time for bands that are optically thick in the upward direction above the altitude at which collisions are important in determining vibrational populations. The spectral radiance program is much slower now that a relatively complete set of spectral lines is used for the water vapor and ozone bands. The first of these problems can be resolved by using a matrix inversion to obtain a first approximation to the vibrational populations of the optically thick bands, as is done in the program developed by James and Kumer. The second problem is a result of trying for completeness,

and a smaller set of lines may be selected and used for some purposes, using more complete results as a guide to what may be safely omitted.

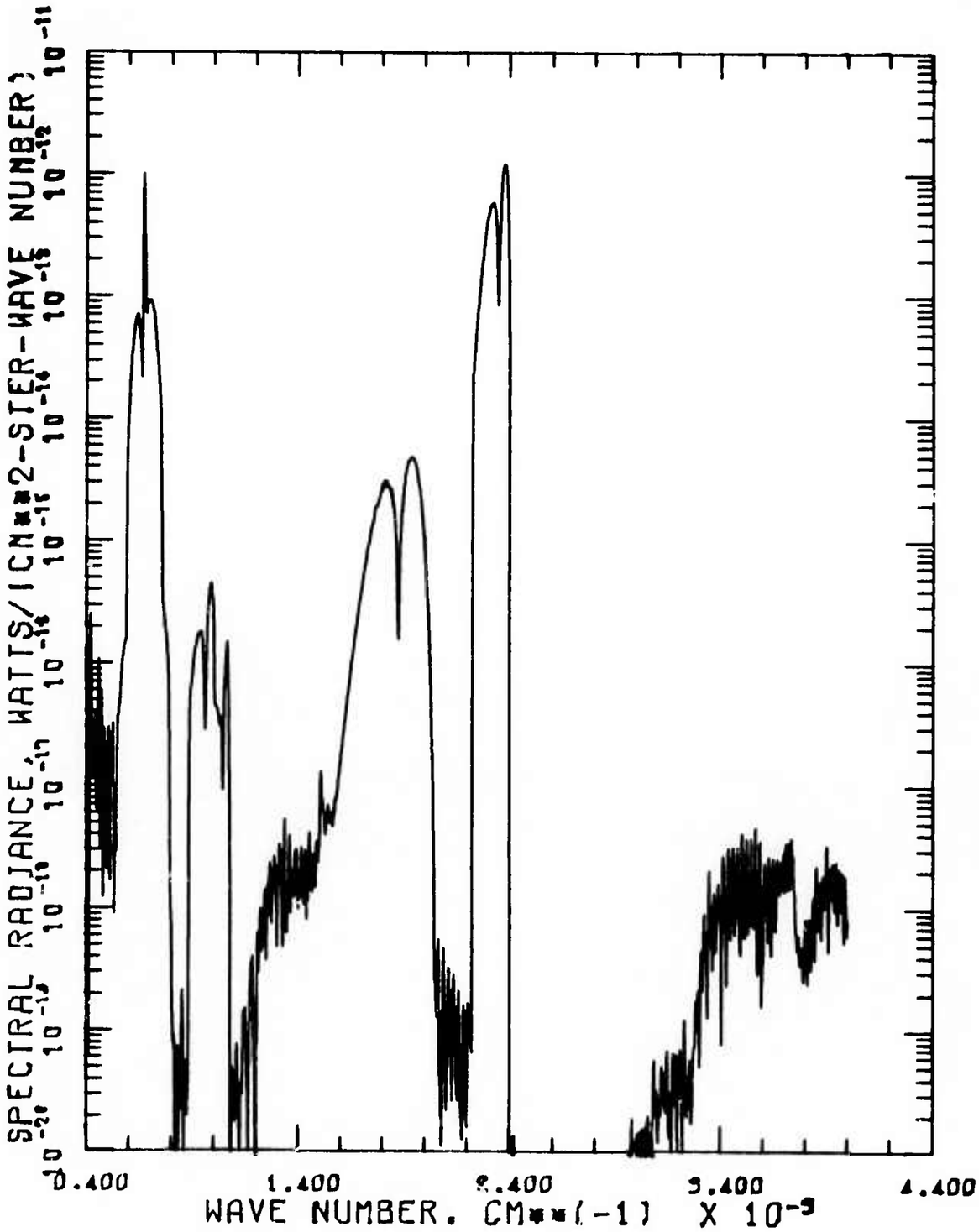
The program could be improved in some other ways by adding features from programs such as that of Whiting, et al (1969), which was written to compute line-by-line spectra of diatomic molecules. Use of this program with either of the Voight profile subroutines used in the current research effort would permit improved calculation of limb radiances, provided composition data are sufficiently accurate to justify the additional programming effort and computer time.

Finally, we use Figures IV-14 and IV-15 to illustrate one remaining problem. These figures give night and day spectral radiance values for a limb view with a tangent height of 500 km. Comparison with Figures IV-1 through IV-6 shows that the band widths are appreciably wider than for the lower atmosphere where the temperature is lower. The program assumes that the effective radiation received from the lower atmosphere and from the lower parts of the upper atmosphere is such as to produce a constant vibrational temperature above 110 or 120 km. Above these altitudes, collisions play no part and excitation is entirely by radiative transfer. The widening of the bands at very high altitudes will permit absorption of radiation in the wings of the band, and this effect is not properly accounted for in the program. Additionally, at an altitude of 500 km, it is probable that the rotational temperature is not in equilibrium with the kinetic temperature. Further work is required to investigate the actual rotational temperature when collisions are not sufficiently rapid to maintain rotational temperature in equilibrium with kinetic temperature.



Night Limb Viewing Spectral Radiance,
400 to 4000 cm^{-1}, 500 km Tangent Height.

FIGURE IV-14



Day Limb Viewing Spectral Radiance,
400 to 4000 cm^{-1}, 500 km Tangent Height.

FIGURE IV-15

SECTION V

HIGH ALTITUDE RADIANCE PROGRAMS

The high altitude (>60 km) radiance program set includes four programs, LOWRAD, BCKGND, SPCTRA and TRYCRT. Each program may be used separately or in appropriate combination with one or more of the others within the CDC 6600 operating system.

LOWRAD is used only to provide an approximation to the spectral radiance in the upward direction from the lower atmosphere. Input to LOWRAD is from two separate files, TAPE 1 which holds the premanent data and data for five model atmospheres and TAPE 5 = INPUT which directs the choice of one of the five atmospheres or another atmosphere, in which case data for this atmosphere must also be on TAPE 5. Printed output is written to TAPE 6 = OUTPUT and output in a form suitable for use by BCKGND is written to TAPE 2, which may be a temporary or permanent disc file or a magnetic tape.

BCKGND computes vibrational populations and total band radiances for an earth's limb viewing geometry at 5 km invervals between 60 and 120 and at 25 km intervals between 120 and 500 km. Control information is read from TAPE 5 = INPUT and permanent data is read from TAPE 1. Low altitude spectral radiances are read from TAPE 2. Atmospheric composition at 2 km intervals between 60 and 150 km and at 5 km intervals between 150 and 700 km may be read either from a sample transferred from the UPDATE set to TAPE 1 or from TAPE 5. Output for printing is written to TAPE 6 = OUTPUT and to TAPE 3 for storage or for use as input to program SPCTRA.

Preceding page blank

SPCTRA computes spectral radiances corresponding to the total band radiances output by BCKGND. Control information is read from TAPE 5 = INPUT. Permanent data is read from TAPE 1 and the results of the BCKGND computation are read from TAPE 3. Output data is sent to TAPE 6 = OUTPUT for printing and to TAPE 4 for use by the plotting program. Optional output of unfiltered spectral radiances may be placed on TAPE 7 and saved for later use.

TRYPEN and TRYCRT plot spectral radiance curves using values computed by program SPCTRA. TRYCRT produces microfilm output; TRYPEN uses a pen plotter.

The programs are listed and described below.

Program LOWRAD

LOWRAD is used to provide an approximation to the spectral radiance in the upward direction from the lower atmosphere. It is based on the computer program, LOWTRAN 1, written by J. E. A. Selby of AFCRL. LOWTRAN 1 provides numerical values based on the atmospheric transmission model of McClatchey, et al (1970).

LOWTRAN 1 is capable of computing transmittances with 20 wavenumber resolution at wavenumbers between 350 and 40,000 cm^{-1} (0.25 to 28.5 micrometers). Computations are restricted to points with numerical value a multiple of five wavenumbers. LOWTRAN 1 uses atmospheric parameters (temperature, pressure, and composition) at one kilometer intervals between 0 and 25 km, at five kilometer intervals to 50 km, and at 70 and 100 km. Variables required at intermediate altitudes are obtained by interpolation.

The modifications to LOWTRAN 1 to make LOWRAD are simple and straightforward. Beginning at 350 cm^{-1}, transmission between 60 km and each lower level of the atmosphere is computed for 5 nadir angles. The Planck function at that wavenumber is computed for ground level temperature and the mean temperature between altitude levels used by the program. The ground level value of the Planck function is multiplied by the transmittance between ground level and 60 km altitude. This and the product of the Planck function and transmittance differences for the intermediate levels are summed to produce spectral radiance values at each of the given angles. The version of LOWRAD listed here uses the five angles arcsin $\sqrt{(2I - 1)/10}$ with I taking values one to five. Equal weights are assigned to the radiances computed with each value of I, and the arithmetic mean is taken to be the value of the upwelling spectral

radiance. An effective radiation temperature is computed from the final spectral radiance value for use (if desired) with program BCKGND. This procedure is repeated at 5 cm^{-1} intervals until 4000 cm^{-1} (2.5 micrometers) for a total of 731 output radiance values. These output values are written both to the line printer and to a disc file for later use.

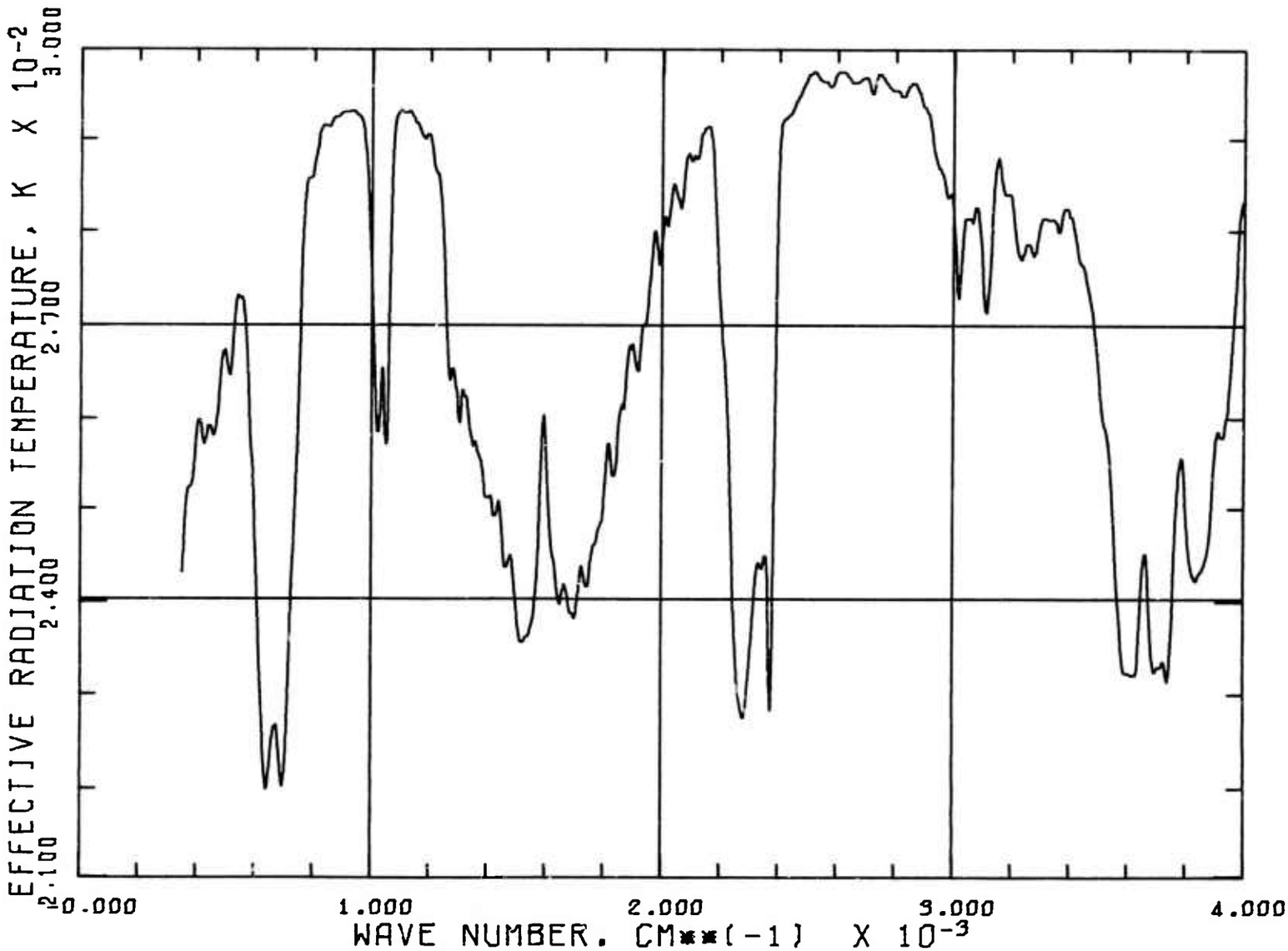
Figures V-1 through V-5 show the resulting values of radiation temperature for the five model atmospheres provided by McClatchey, et al (1970). The effects of the strong absorption bands of CO_2, H_2O and O_3 are easily seen in the lower temperatures at the most intense parts of the bands, where a detector looking down would in effect see the cold lower stratosphere.

Table V-1 lists radiation temperature values for the five model atmospheres, (1) tropical, (2) midlatitude summer, (3) midlatitude winter, (4) sub-artic summer and (5) sub-artic winter.

The radiance values and effective radiation temperatures computed by LOWRAD are limited in usefulness by the low resolution available. A molecule in the upper atmosphere acts as its own spectrometer with a resolution comparable to its line width which is of the order of between one hundredth and one tenth of a wave number. For most bands, a better value of effective radiation temperature is that of the kinetic temperature at the altitude taken to be the base of the high altitude region. This is 60 km for the high altitude radiance program BCKGND.

Input to LOWRAD is from two separate files, TAPE 5, which holds permanent transmission data and data for five model atmospheres and TAPE 4 which directs the choice of one of the five atmospheres or another atmosphere, in which case data for this atmosphere must also be on TAPE 5. Printed output is written to TAPE 6 = OUTPUT and

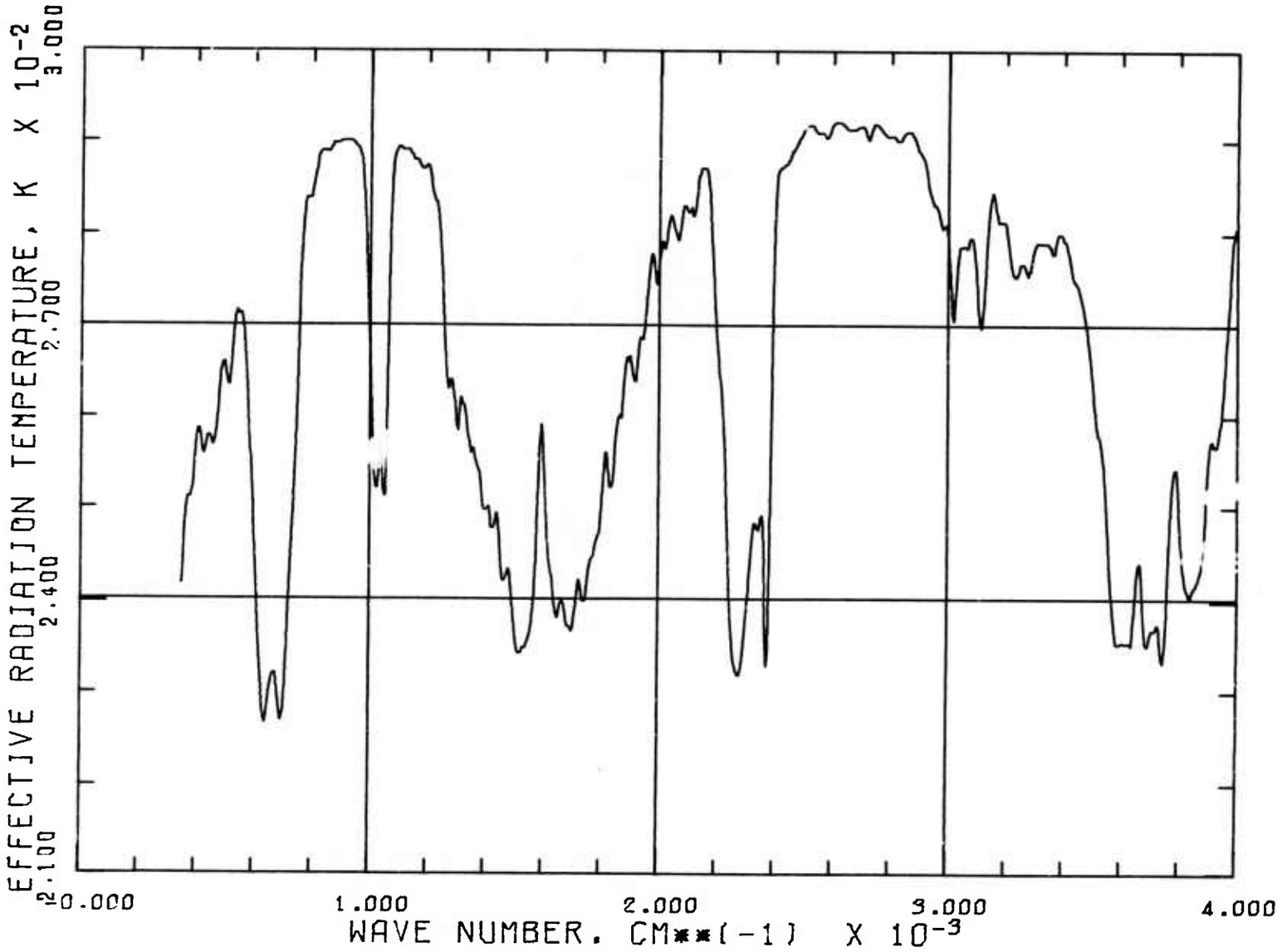
output in a form suitable for use by BCKGND is written to TAPE 2, which may be a temporary or permanent disc file or a magnetic tape.



Lower Atmosphere Radiation Temperature

For Tropical Model Atmosphere

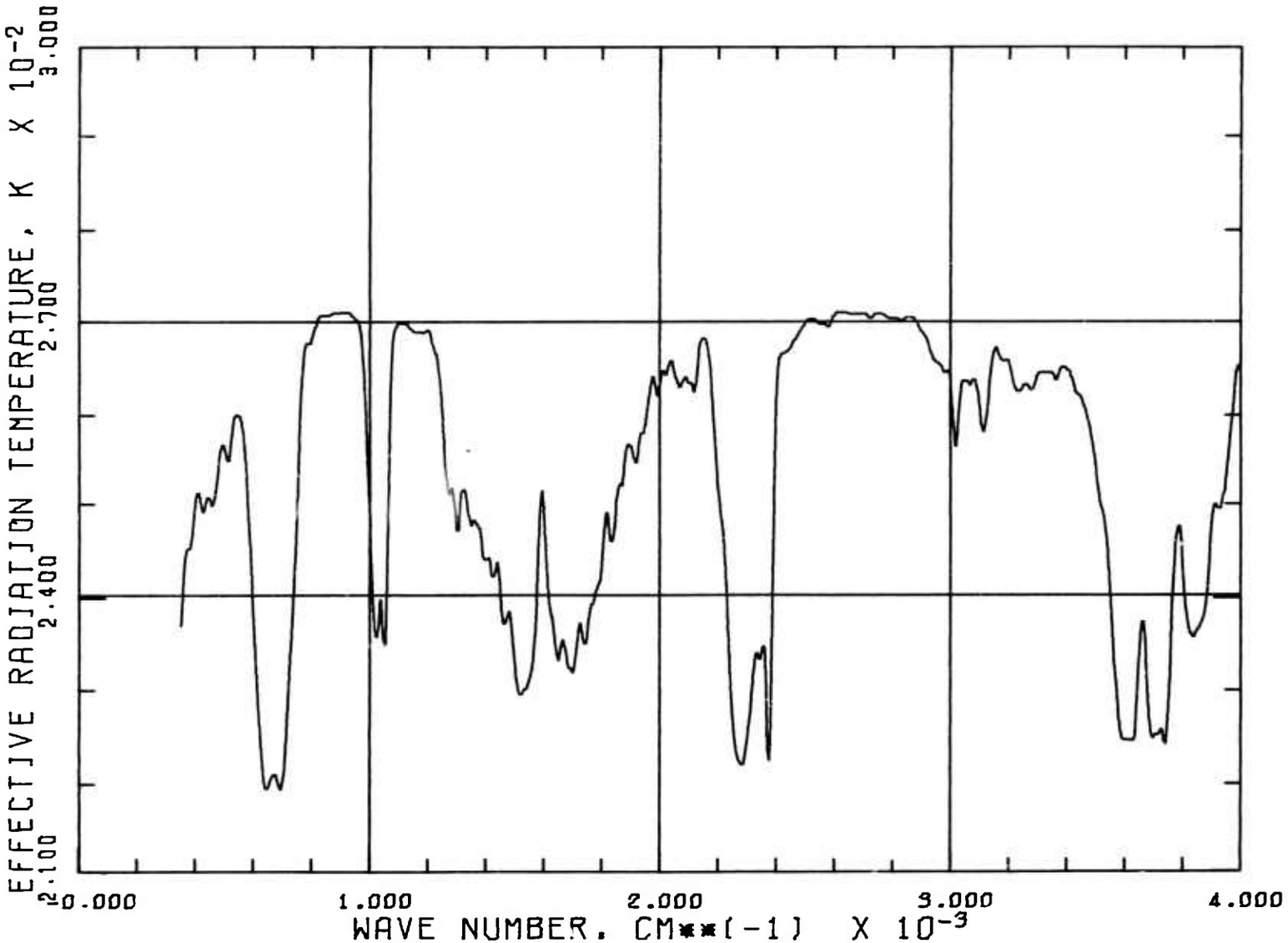
FIGURE V-1



Lower Atmosphere Radiation Temperature

For Midlatitude Summer Model Atmosphere

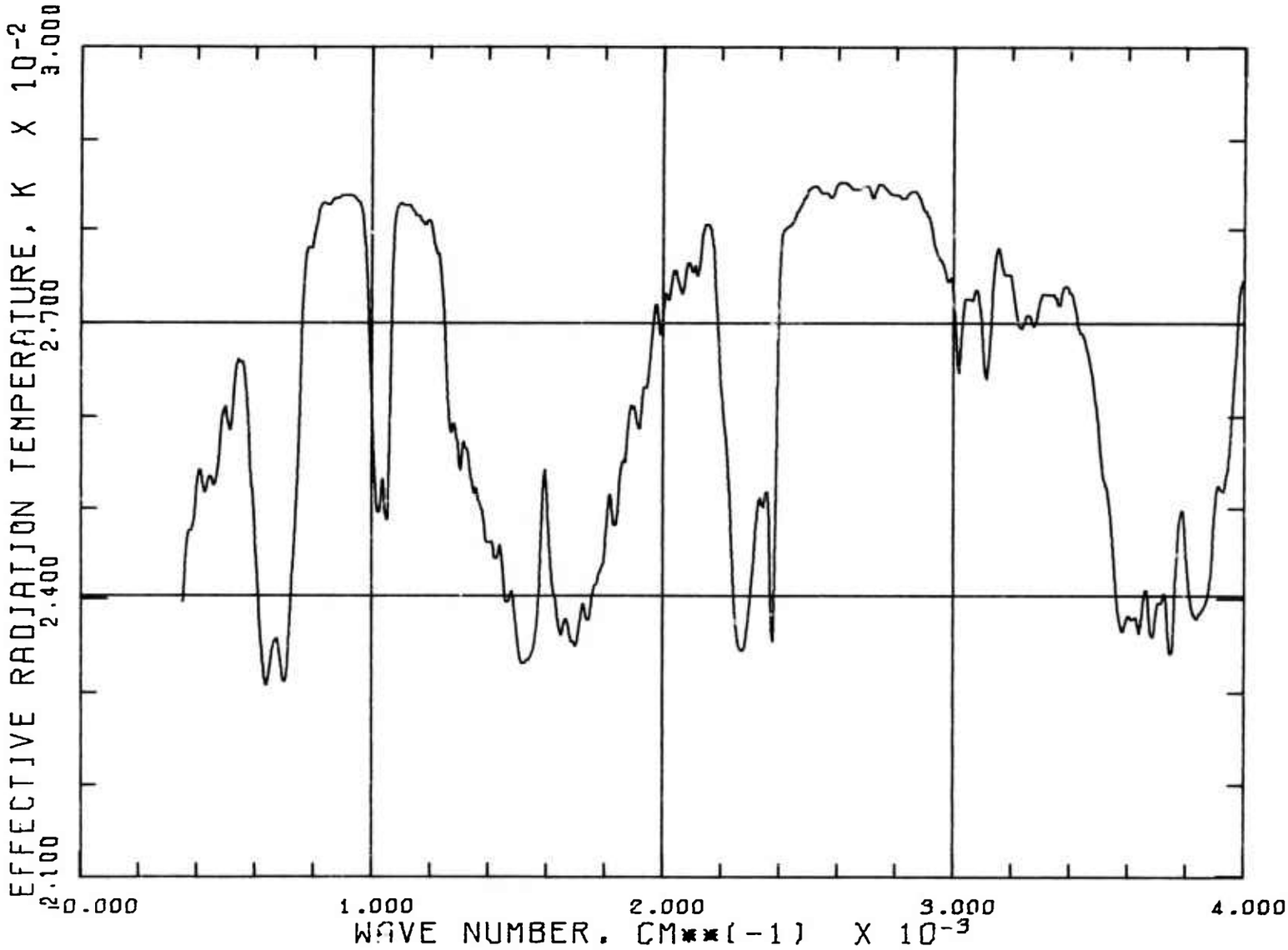
FIGURE V-2



Lower Atmosphere Radiation Temperature

For Midlatitude Winter Model Atmosphere

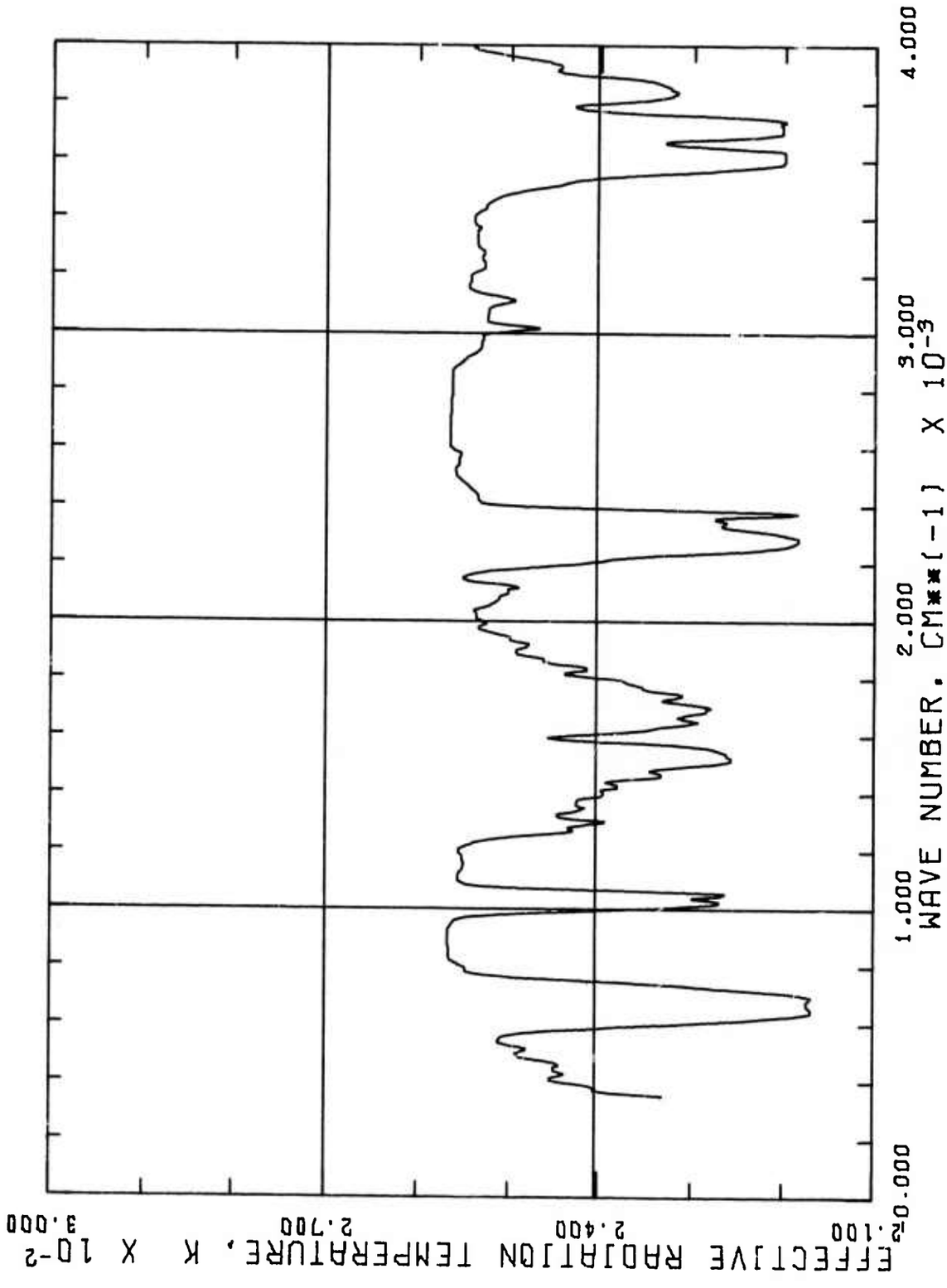
FIGURE V-3



Lower Atmosphere Radiation Temperature

For Sub-Arctic Summer Model Atmosphere

FIGURE V-4



Lower Atmosphere Radiation Temperature

For Sub-Arctic Winter Model Atmosphere

FIGURE V-5

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
350	242.93	241.75	236.85	239.43	232.82
355	246.31	245.03	239.88	242.29	235.71
360	249.36	248.09	242.55	244.87	238.15
365	251.46	250.19	244.34	246.70	239.75
370	252.35	251.03	245.10	247.48	240.41
375	252.36	251.04	245.10	247.48	240.42
380	252.55	251.27	245.26	247.65	240.56
385	253.08	251.81	245.72	248.12	240.95
390	254.51	253.24	246.92	249.38	241.97
395	256.71	255.42	248.71	251.31	243.44
400	258.75	257.45	250.35	253.15	244.71
405	259.88	258.59	251.23	254.16	245.33
410	259.70	258.41	251.09	254.00	245.23
415	259.15	257.85	250.66	253.50	244.96
420	257.67	256.33	249.49	252.17	244.05
425	256.95	255.63	248.91	251.53	243.59
430	257.70	256.41	249.51	252.19	244.07
435	258.64	257.34	250.26	253.04	244.64
440	259.21	257.91	250.70	253.55	244.99
445	259.03	257.73	250.56	253.39	244.88
450	258.67	257.37	250.28	253.06	244.66
455	257.94	256.63	249.70	252.40	244.21
460	258.32	257.03	250.01	252.74	244.45
465	258.05	257.78	250.50	253.42	244.91
470	260.40	259.11	251.62	254.61	245.69
475	262.45	261.16	253.20	256.49	246.85
480	264.72	263.33	254.86	258.53	248.05
485	266.44	265.07	256.07	260.09	248.79
490	267.20	265.82	256.61	260.79	249.14
495	267.40	266.01	256.74	260.97	249.23
500	266.47	265.10	256.09	260.11	248.81
505	265.54	264.18	255.44	259.26	248.38
510	264.60	263.25	254.77	258.41	247.94
515	264.80	263.45	254.90	258.59	248.02
520	266.73	265.35	256.22	260.35	248.85
525	269.29	267.83	257.92	262.66	249.93
530	271.17	269.65	259.07	264.35	250.69
535	272.62	271.01	259.91	265.62	251.03
540	273.13	271.48	260.14	266.06	251.11
545	272.64	271.02	259.75	265.64	250.81
550	272.92	271.29	259.79	265.91	250.73
555	272.71	271.09	259.50	265.74	250.42
560	271.95	270.40	258.85	265.11	249.87

TABLE V-1: Effective Lower Atmosphere Radiation Temperature

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
565	270.34	268.92	257.61	263.77	248.90
570	267.68	266.43	255.61	261.57	247.37
575	264.34	263.41	253.08	258.84	245.41
580	260.48	259.86	250.05	255.72	242.96
585	256.58	256.41	246.62	252.70	239.90
590	254.97	255.02	244.75	251.60	237.99
595	253.61	251.13	240.90	248.49	234.55
600	245.98	247.22	237.32	245.26	231.52
605	242.25	244.03	234.54	242.73	229.20
610	237.55	240.16	231.26	239.69	226.50
615	233.32	236.67	228.41	237.07	224.19
620	229.86	233.88	226.14	235.00	222.37
625	226.32	231.15	223.86	233.11	220.53
630	222.53	228.31	221.41	231.33	218.54
635	220.52	226.93	220.08	230.67	217.46
640	219.66	226.49	219.29	230.76	216.79
645	220.59	227.41	219.26	231.80	216.64
650	222.38	228.84	219.70	233.03	216.79
655	224.00	230.07	220.16	234.00	217.00
660	225.35	231.09	220.57	234.83	217.23
665	225.37	231.85	221.91	235.44	217.40
670	226.69	232.10	221.02	235.64	217.47
675	226.46	231.93	220.95	235.50	217.42
680	224.75	230.61	220.39	234.44	217.13
685	222.90	229.22	219.85	233.34	216.87
690	221.00	227.64	219.36	232.03	216.66
695	220.05	226.77	219.29	231.05	216.74
700	221.31	227.39	220.36	230.90	217.66
705	222.98	228.53	221.38	231.48	218.47
710	227.03	231.51	223.79	233.34	220.36
715	232.46	235.71	227.16	235.29	222.97
720	236.66	239.13	229.81	238.87	225.07
725	241.74	243.34	233.19	242.22	227.68
730	245.43	246.50	235.74	244.71	229.69
735	249.10	249.75	238.46	247.38	231.86
740	253.26	253.47	241.62	250.50	234.40
745	257.30	257.10	244.83	253.60	237.00
750	264.06	263.31	250.03	259.07	241.09
755	270.39	269.22	255.04	264.44	245.00
760	275.90	275.36	260.20	270.11	248.93
765	280.48	278.73	263.10	273.31	251.10
770	283.30	281.48	265.49	275.82	252.99
775	285.16	283.30	267.05	277.53	254.22

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
780	285.56	283.69	267.52	277.88	254.68
785	285.71	283.81	267.64	277.99	254.83
790	285.68	283.75	267.63	277.93	254.87
795	285.80	283.86	267.71	278.03	254.93
800	286.39	284.41	268.06	278.53	255.19
805	287.58	285.43	268.75	279.56	255.58
810	288.61	286.42	269.34	280.47	255.91
815	289.39	287.12	269.74	281.14	256.13
820	290.64	288.24	270.36	282.21	256.42
825	291.11	288.64	270.58	282.59	256.53
830	291.41	288.89	270.71	282.83	256.59
835	291.46	288.91	270.72	282.84	256.61
840	291.24	288.69	270.60	282.60	256.58
845	291.18	288.61	270.54	282.52	256.56
850	291.12	288.51	270.48	282.42	256.54
855	291.36	288.72	270.57	282.62	256.57
860	291.70	289.01	270.70	282.88	256.61
865	292.11	289.35	270.85	283.21	256.67
870	292.42	289.61	270.96	283.45	256.71
875	292.46	289.62	270.96	283.46	256.70
880	292.45	289.61	270.94	283.43	256.69
885	292.54	289.65	270.95	283.48	256.70
890	292.70	289.77	270.99	283.58	256.71
895	292.84	289.87	271.03	283.68	256.72
900	292.99	289.93	271.07	283.77	256.73
905	292.92	289.91	271.00	283.68	256.69
910	292.91	289.87	270.96	283.65	256.66
915	292.88	289.82	270.89	283.59	256.61
920	292.80	289.81	270.84	283.57	256.56
925	293.15	289.93	270.85	283.74	256.51
930	293.07	289.83	270.72	283.63	256.41
935	292.96	289.75	270.56	283.49	256.23
940	292.79	289.57	270.41	283.32	256.18
945	292.54	289.33	270.24	283.07	256.07
950	292.39	289.12	270.07	282.86	255.94
955	292.26	288.93	269.94	282.71	255.81
960	292.16	288.81	269.75	282.53	255.65
965	291.65	288.33	269.22	282.02	255.12
970	291.24	287.54	268.40	281.22	254.34
975	289.56	285.13	265.93	278.82	251.93
980	288.49	283.63	264.32	277.30	250.44
985	286.03	280.44	261.01	274.18	247.28
990	282.45	275.78	256.20	269.61	242.83

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
995	279.31	271.89	252.53	265.92	239.64
1000	275.50	267.53	248.48	261.90	236.29
1005	269.61	261.23	242.92	256.39	232.07
1010	265.15	257.04	239.56	253.03	229.73
1015	259.98	253.02	236.47	250.07	227.56
1020	258.34	251.91	235.57	249.32	226.86
1025	258.60	252.03	235.71	249.42	226.97
1030	261.33	253.93	237.16	250.75	228.07
1035	265.48	257.32	239.73	253.22	229.84
1040	264.85	256.75	239.28	252.77	229.52
1045	258.30	251.86	235.48	249.24	226.80
1050	256.93	250.93	234.75	248.64	226.28
1055	257.68	251.46	235.18	248.96	226.59
1060	270.03	261.77	243.42	256.75	232.47
1065	278.40	271.12	252.01	265.14	239.33
1070	286.56	281.25	261.98	274.91	248.42
1075	289.95	285.47	266.13	279.04	252.26
1080	291.69	287.62	268.18	281.19	254.13
1085	292.52	288.56	269.06	282.11	254.92
1090	292.82	288.83	269.41	282.44	255.24
1095	293.09	289.21	269.77	282.75	255.56
1100	293.03	289.03	269.80	282.65	255.64
1105	292.72	288.91	269.75	282.48	255.64
1110	292.61	288.79	269.67	282.37	255.57
1115	292.78	288.91	269.69	282.47	255.53
1120	292.87	288.91	269.58	282.47	255.36
1125	293.01	288.93	269.52	282.51	255.25
1130	293.07	288.99	269.45	282.52	255.15
1135	292.65	288.61	269.10	282.14	254.93
1140	292.29	288.31	268.99	281.86	254.88
1145	292.00	288.07	268.87	281.65	254.83
1150	291.56	287.71	268.66	281.31	254.75
1155	291.49	287.71	268.75	281.33	254.90
1160	291.31	287.64	268.84	281.27	255.06
1165	290.89	287.37	268.82	281.04	255.16
1170	290.37	287.01	268.73	280.70	255.21
1175	290.04	286.75	268.65	280.49	255.24
1180	289.81	286.54	268.59	280.33	255.24
1185	289.78	286.53	268.58	280.33	255.26
1190	290.32	287.07	268.91	280.90	255.46
1195	290.53	287.25	269.05	281.09	255.56
1200	290.20	287.01	268.94	280.[illegible]	255.54
1205	289.48	286.41	268.65	280.21	255.42

TABLE V-1: Continued

WAVE NUMBER	1	1001 ATMOSPHERE 2	3	4	5
1210	288.47	285.47	263.12	279.30	255.14
1215	287.04	284.21	267.44	278.17	254.76
1220	286.40	283.63	266.97	277.61	254.44
1225	285.92	283.21	266.44	277.14	253.98
1230	285.87	283.08	266.13	277.12	253.64
1235	284.81	282.15	265.28	276.14	252.93
1240	283.11	280.49	264.01	274.58	252.04
1245	281.53	278.13	262.21	272.32	250.77
1250	277.40	275.31	261.21	269.69	249.41
1255	272.12	270.33	256.80	265.07	247.15
1260	268.84	267.31	254.25	262.21	245.16
1265	266.60	264.31	252.11	259.44	243.56
1270	264.07	262.87	251.99	259.13	242.71
1275	264.85	263.41	251.29	258.61	243.01
1280	265.49	264.13	251.60	259.31	243.46
1285	264.83	263.54	251.16	258.73	242.78
1290	263.36	262.12	249.69	257.45	241.45
1295	262.63	261.41	248.93	256.92	240.75
1300	261.00	259.97	246.87	254.64	239.11
1305	264.32	258.31	246.91	254.29	239.28
1310	262.28	261.19	249.91	256.54	242.16
1315	263.20	262.58	251.42	257.33	243.84
1320	262.33	261.28	251.59	256.48	244.52
1325	262.19	261.07	251.65	256.32	244.72
1330	261.62	260.54	251.05	255.84	244.15
1335	259.69	258.71	249.51	254.18	242.89
1340	256.56	257.61	248.69	253.19	242.27
1345	257.86	256.94	248.08	252.59	241.77
1350	256.70	255.73	247.36	251.54	241.34
1355	257.44	256.45	248.13	252.12	242.11
1360	257.21	256.15	248.25	251.81	242.44
1365	256.19	255.15	247.74	250.78	242.30
1370	255.72	254.93	247.50	250.31	242.23
1375	255.19	253.95	247.19	249.78	242.10
1380	254.98	253.70	247.10	249.55	242.10
1385	253.56	252.34	246.05	248.36	241.26
1390	251.30	249.93	244.08	246.23	239.55
1395	251.02	249.65	243.85	245.97	239.34
1400	251.05	249.68	243.87	246.00	239.38
1405	251.24	249.87	244.03	246.17	239.53
1410	251.43	250.07	244.20	246.33	239.67
1415	250.74	249.37	243.60	245.72	239.14
1420	249.02	247.67	242.10	244.22	237.78

TABLE V-1: Continued

Reproduced from best available copy.

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
1425	249.03	247.64	242.09	244.21	237.77
1430	249.34	247.96	242.36	244.49	238.02
1435	250.16	248.87	243.08	245.32	238.66
1440	250.85	249.49	243.57	245.82	239.19
1445	249.48	248.11	242.48	244.62	238.12
1450	247.50	246.03	240.72	242.86	236.49
1455	245.08	243.63	238.53	240.72	234.41
1460	243.49	242.04	237.07	239.27	233.03
1465	243.51	242.05	237.08	239.38	233.04
1470	244.01	242.54	237.54	239.79	233.47
1475	244.51	243.04	237.99	240.22	233.91
1480	245.01	243.54	238.45	240.64	234.35
1485	244.23	242.76	237.74	239.97	233.67
1490	242.36	240.88	236.01	238.40	232.01
1495	241.02	239.56	234.75	237.28	230.77
1500	239.12	237.67	232.96	235.74	229.01
1505	237.13	235.81	231.09	234.22	227.22
1510	236.81	234.67	229.93	233.36	226.13
1515	235.40	234.19	229.44	232.93	225.65
1520	235.27	234.07	229.31	232.91	225.53
1525	235.53	234.31	229.55	233.18	225.76
1530	236.03	234.73	230.04	233.44	226.23
1535	236.14	234.83	230.15	233.52	226.33
1540	236.14	234.84	230.15	233.52	226.33
1545	236.58	235.23	230.55	233.82	226.71
1550	237.17	235.83	231.10	234.24	227.23
1555	237.62	236.27	231.52	234.57	227.64
1560	238.52	237.09	232.37	235.24	228.43
1565	240.29	238.82	234.04	236.64	230.06
1570	242.40	240.91	236.07	238.39	232.02
1575	245.90	244.44	239.26	241.40	235.11
1580	250.78	249.41	243.61	245.72	239.16
1585	255.80	254.43	247.84	250.21	242.80
1590	259.29	258.00	251.64	253.36	245.92
1595	260.41	259.13	251.51	254.38	245.67
1600	257.81	256.51	249.46	252.02	244.10
1605	253.62	252.23	245.02	248.25	241.25
1610	250.26	248.84	243.15	245.27	238.73
1615	247.90	246.43	241.05	243.15	236.80
1620	245.77	244.31	239.13	241.27	234.98
1625	244.65	243.17	238.10	240.29	234.01
1630	244.03	242.55	237.53	239.76	233.47
1635	242.35	240.85	235.97	238.32	231.96

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
1640	241.18	239.67	234.94	237.32	230.86
1645	240.00	238.51	233.74	236.37	229.77
1650	239.43	237.95	233.21	235.92	229.25
1655	240.03	238.54	233.77	236.39	229.83
1660	241.23	239.73	234.91	237.37	230.92
1665	241.69	240.18	235.34	237.76	231.35
1670	241.10	239.60	234.78	237.26	230.80
1675	240.52	239.01	234.23	236.78	230.24
1680	239.49	238.00	233.25	235.95	229.28
1685	239.03	237.01	232.28	235.15	228.35
1690	238.48	237.00	232.27	235.15	228.34
1695	238.19	236.72	231.99	234.92	228.08
1700	237.90	236.43	231.71	234.71	227.81
1705	238.77	237.24	232.53	235.36	228.57
1710	240.23	238.71	233.99	236.51	229.91
1715	241.42	239.89	235.02	237.50	231.00
1720	242.81	241.27	236.37	238.66	232.30
1725	243.76	242.24	237.27	239.48	233.13
1730	242.99	241.45	236.51	238.81	232.48
1735	241.91	240.35	235.45	237.89	231.44
1740	241.42	239.86	234.94	237.47	230.92
1745	241.36	239.80	234.88	237.41	230.92
1750	242.45	240.91	235.92	238.36	232.01
1755	244.35	242.85	237.81	239.98	233.74
1760	245.17	243.67	238.56	241.69	234.45
1765	246.06	244.51	239.32	241.41	235.17
1770	246.18	244.63	239.47	241.56	235.31
1775	246.52	245.04	239.78	241.87	235.61
1780	247.36	245.90	240.55	242.61	236.33
1785	247.87	246.42	241.01	243.06	236.77
1790	248.36	246.95	241.47	243.53	237.20
1795	248.73	247.30	241.78	243.84	237.49
1800	250.10	248.71	242.99	245.06	238.61
1805	252.94	251.53	245.43	247.58	240.77
1810	255.48	254.15	247.54	249.97	242.57
1815	257.32	256.01	249.05	251.53	243.79
1820	257.15	255.84	248.90	251.37	243.68
1825	255.15	253.83	247.27	249.57	242.34
1830	253.36	251.99	245.78	247.94	241.06
1835	253.37	252.00	245.79	247.96	241.08
1840	253.93	252.58	246.26	248.46	241.49
1845	255.39	254.07	247.48	249.79	242.52
1850	259.09	257.91	251.46	253.14	244.91

TABLE V-1: Continued

Reproduced from best available copy.

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
1855	260.41	259.12	251.50	254.34	245.69
1860	261.17	259.89	252.09	255.04	246.13
1865	261.56	260.23	252.38	255.39	246.35
1870	261.01	259.73	251.96	254.89	246.04
1875	263.47	262.17	253.90	257.14	247.37
1880	265.34	264.04	255.15	258.87	248.28
1885	267.18	265.85	256.42	260.57	249.11
1890	267.85	266.51	256.84	261.17	249.35
1895	267.35	266.02	256.46	260.72	249.07
1900	267.88	266.51	256.73	261.19	249.17
1905	266.93	265.63	256.03	260.32	248.67
1910	265.69	264.31	255.13	259.18	248.04
1915	265.10	263.81	254.70	258.64	247.75
1920	264.95	263.65	254.59	259.50	247.68
1925	267.01	265.67	256.02	261.39	248.62
1930	269.12	267.71	257.43	262.31	249.51
1935	270.07	268.64	258.12	263.19	249.99
1940	270.10	268.67	258.20	263.22	250.11
1945	269.68	268.27	257.93	262.84	250.01
1950	271.30	269.83	259.06	264.32	250.69
1955	272.70	271.15	259.96	265.59	251.25
1960	274.88	273.19	261.23	267.52	251.98
1965	276.99	275.13	262.41	269.35	252.61
1970	278.88	276.83	263.41	271.07	253.15
1975	279.91	277.74	263.95	271.85	253.44
1980	279.47	277.35	263.74	271.48	253.34
1985	278.02	276.07	263.00	271.24	252.96
1990	276.14	274.35	261.99	269.63	252.42
1995	276.68	274.85	262.29	269.09	252.57
2000	279.00	276.85	263.45	271.98	253.13
2005	280.37	278.14	264.18	272.21	253.55
2010	281.62	279.22	264.77	273.23	253.94
2015	280.80	278.51	264.35	272.55	253.51
2020	280.41	278.15	264.12	272.22	253.46
2025	281.69	279.25	264.63	273.24	253.69
2030	283.25	280.55	265.31	274.47	253.91
2035	284.53	281.53	265.71	275.41	253.98
2040	284.95	281.84	265.61	275.63	253.74
2045	284.64	281.44	265.05	275.24	253.18
2050	283.80	280.57	264.19	274.39	252.41
2055	283.41	280.15	263.79	273.98	252.07
2060	282.87	279.65	263.42	273.48	251.79
2065	282.31	279.07	262.97	272.93	251.40

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
2070	283.40	279.83	263.14	273.57	251.23
2075	284.59	280.62	263.33	274.26	251.07
2080	286.91	282.22	263.78	275.67	250.93
2085	288.04	282.97	264.03	276.33	250.91
2090	288.28	282.96	263.93	276.28	250.58
2095	287.94	282.52	263.42	275.92	250.13
2100	287.32	282.05	263.23	275.38	251.04
2105	287.46	282.11	263.19	275.43	249.96
2110	286.14	282.65	263.47	275.92	250.10
2115	287.61	281.71	262.23	274.93	248.31
2120	287.56	281.91	262.70	275.16	249.32
2125	288.05	282.61	263.51	275.88	250.17
2130	289.23	284.15	265.20	277.47	251.89
2135	290.58	286.03	267.33	279.49	254.05
2140	291.61	286.33	267.74	279.78	254.56
2145	291.07	286.83	268.22	280.34	255.04
2150	291.23	286.93	268.21	280.43	255.03
2155	291.34	286.93	267.99	280.39	254.73
2160	291.24	286.75	267.57	280.14	254.39
2165	291.07	286.31	266.92	279.73	253.74
2170	289.88	285.22	265.83	278.58	252.88
2175	288.44	283.84	264.61	277.24	251.94
2180	285.07	280.68	261.95	274.21	249.33
2185	281.82	277.65	259.41	271.31	247.89
2190	278.90	274.91	257.10	268.71	246.06
2195	274.82	271.11	253.92	265.14	243.52
2200	272.04	268.55	251.81	262.75	241.36
2205	270.15	266.85	250.44	261.17	240.73
2210	267.45	264.33	248.43	258.91	239.22
2215	266.63	263.61	247.78	258.20	238.71
2220	264.19	261.35	245.88	256.15	237.19
2225	261.02	258.43	243.44	253.53	235.23
2230	257.53	255.23	240.87	251.76	233.17
2235	252.01	250.39	236.92	246.56	230.07
2240	246.89	245.93	233.44	242.94	227.32
2245	240.37	240.51	229.17	238.90	224.03
2250	236.70	237.53	225.85	235.95	222.22
2255	234.26	235.77	225.44	235.85	221.08
2260	231.20	233.62	223.74	234.75	219.76
2265	229.63	232.63	222.90	234.38	219.12
2270	228.59	232.05	222.37	234.29	218.71
2275	227.81	231.70	221.97	234.37	218.38
2280	227.39	231.67	221.79	234.73	218.19

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
2285	227.60	232.13	221.96	235.54	218.22
2290	228.60	233.32	222.46	235.73	218.49
2295	230.08	234.75	223.25	238.11	218.94
2300	232.16	236.67	224.46	239.86	219.67
2305	234.51	238.80	225.93	241.90	220.63
2310	236.22	240.48	227.10	243.39	221.43
2315	238.60	242.68	228.94	245.46	222.79
2320	240.95	245.03	230.87	247.77	224.35
2325	243.07	247.03	232.83	249.90	225.82
2330	244.05	248.17	233.98	251.93	226.71
2335	244.27	248.41	234.11	251.17	226.90
2340	243.51	247.55	233.20	250.30	226.21
2345	243.33	247.35	233.11	251.08	226.04
2350	244.21	248.35	234.06	251.11	226.86
2355	244.85	249.02	234.70	251.78	227.38
2360	244.97	249.14	234.81	251.89	227.49
2365	242.00	246.01	231.81	248.82	225.09
2370	231.24	235.83	223.94	239.11	219.34
2375	228.05	232.59	222.18	235.89	218.34
2380	232.30	234.35	224.19	235.18	220.05
2385	248.38	247.05	234.06	243.70	227.72
2390	262.73	259.73	244.15	254.54	235.74
2395	274.62	270.69	253.06	264.53	242.88
2400	284.50	279.85	261.62	273.18	248.95
2405	289.28	284.21	264.27	277.41	251.81
2410	291.64	286.48	266.06	279.52	253.21
2415	291.94	286.75	265.28	279.78	253.36
2420	292.21	286.99	266.48	281.02	253.49
2425	292.32	287.09	266.56	282.13	253.54
2430	292.44	287.21	266.65	280.24	253.59
2435	292.62	287.35	266.77	280.39	253.66
2440	292.74	287.46	266.85	280.51	253.70
2445	292.98	287.67	267.02	280.71	253.82
2450	293.30	287.95	267.26	280.99	253.97
2455	293.74	288.37	267.59	281.39	254.19
2460	294.16	288.75	267.90	281.77	254.40
2465	294.43	288.99	268.10	282.01	254.53
2470	294.77	289.30	268.35	282.31	254.69
2475	295.10	289.60	269.50	282.61	254.84
2480	295.44	289.90	268.84	282.91	255.00
2485	295.88	290.30	269.17	283.31	255.22
2490	296.22	290.61	269.42	283.61	255.40
2495	296.54	290.91	269.66	283.90	255.57

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
2500	296.91	291.24	269.94	284.24	255.75
2505	297.24	291.55	270.19	284.54	255.91
2510	297.33	291.63	270.26	284.62	255.95
2515	297.46	291.75	270.36	284.74	256.02
2520	297.50	291.78	270.40	284.78	256.03
2525	297.39	291.68	270.32	284.68	255.97
2530	297.15	291.47	270.15	284.49	255.84
2535	296.87	291.21	269.96	284.24	255.69
2540	296.57	290.94	269.75	283.99	255.52
2545	296.35	290.74	269.60	283.80	255.39
2550	296.24	290.65	269.55	283.72	255.36
2555	296.27	290.71	269.60	283.77	255.40
2560	296.31	290.75	269.67	283.82	255.46
2565	296.19	290.65	269.63	283.75	255.44
2570	295.92	290.42	269.45	283.51	255.30
2575	295.66	290.20	269.30	283.31	255.20
2580	295.67	290.23	269.36	283.35	255.25
2585	296.06	290.62	269.71	283.73	255.52
2590	296.55	291.03	270.14	284.19	255.87
2595	297.02	291.54	270.54	284.63	256.13
2600	297.32	291.82	270.78	284.91	256.36
2605	297.53	292.03	270.99	285.12	256.52
2610	297.56	292.09	271.07	285.18	256.59
2615	297.57	292.10	271.09	285.19	256.61
2620	297.39	291.96	271.04	285.06	256.60
2625	297.25	291.85	270.90	284.97	256.58
2630	297.16	291.78	270.95	284.91	256.56
2635	296.92	291.61	270.87	284.74	256.53
2640	296.70	291.44	270.80	284.59	256.50
2645	296.52	291.30	270.73	284.47	256.48
2650	296.19	291.05	270.62	284.24	256.43
2655	296.13	291.01	270.60	284.20	256.43
2660	296.10	290.99	270.59	284.18	256.43
2665	296.15	291.03	270.62	284.21	256.44
2670	296.39	291.21	270.70	284.38	256.47
2675	296.44	291.24	270.72	284.42	256.48
2680	296.56	291.35	270.77	284.52	256.50
2685	296.79	291.51	270.85	284.66	256.54
2690	296.81	291.54	270.87	284.69	256.55
2695	296.91	291.62	270.91	284.76	256.57
2700	296.96	291.65	270.93	284.78	256.57
2705	296.61	291.34	270.80	284.55	256.52
2710	296.04	290.96	270.61	284.16	256.44

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
2715	295.35	290.44	270.37	283.69	256.34
2720	294.80	290.03	270.18	283.32	256.26
2725	295.06	290.22	270.26	283.48	256.29
2730	295.95	290.88	270.56	284.08	256.41
2735	296.68	291.41	270.79	284.56	256.49
2740	297.14	291.76	270.94	284.88	256.54
2745	297.20	291.80	270.95	284.92	256.54
2750	297.01	291.65	270.88	284.77	256.50
2755	296.80	291.49	270.80	284.63	256.47
2760	296.58	291.32	270.72	284.48	256.43
2765	296.32	291.12	270.61	284.29	256.37
2770	296.05	290.91	270.50	284.09	256.31
2775	295.79	290.71	270.38	283.90	256.25
2780	295.55	290.51	270.27	283.72	256.19
2785	295.37	290.36	270.19	283.59	256.15
2790	295.24	290.27	270.16	283.51	256.14
2795	295.29	290.31	270.18	283.55	256.15
2800	295.29	290.34	270.23	283.59	256.21
2805	295.27	290.34	270.25	283.59	256.24
2810	295.17	290.27	270.23	283.53	256.25
2815	295.01	290.15	270.17	283.42	256.23
2820	294.63	289.87	270.03	283.16	256.16
2825	294.67	289.89	270.04	283.18	256.16
2830	294.75	289.95	270.07	283.24	256.18
2835	294.97	290.13	270.16	283.40	256.22
2840	295.37	290.43	270.30	283.68	256.29
2845	295.78	290.74	270.46	283.95	256.36
2850	296.02	290.91	270.53	284.11	256.38
2855	296.20	291.03	270.55	284.21	256.37
2860	296.17	290.99	270.51	284.17	256.33
2865	296.12	290.94	270.45	284.12	256.28
2870	295.95	290.80	270.34	283.98	256.21
2875	295.60	290.49	270.12	283.68	256.06
2880	295.20	290.15	269.86	283.36	255.90
2885	294.85	289.85	269.56	283.08	255.74
2890	294.24	289.32	269.30	282.59	255.51
2895	293.48	288.71	268.92	282.01	255.27
2900	293.29	288.51	268.73	281.81	255.11
2905	293.19	288.37	268.54	281.67	254.94
2910	292.79	288.00	268.26	281.32	254.72
2915	292.45	287.71	268.06	281.05	254.59
2920	291.85	287.23	267.75	281.59	254.40
2925	291.81	286.37	267.25	279.80	254.09

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
2930	289.24	285.12	266.57	278.64	253.73
2935	288.37	284.42	266.21	277.98	253.55
2940	287.69	283.84	265.96	277.43	253.33
2945	287.06	283.32	265.55	276.95	253.16
2950	286.59	282.93	265.35	276.58	253.05
2955	286.47	282.84	265.32	276.50	253.05
2960	286.12	282.57	265.22	276.24	253.03
2965	285.74	282.27	265.10	275.96	253.01
2970	284.95	281.65	264.87	275.40	252.98
2975	283.93	280.84	264.52	274.64	252.87
2980	283.40	280.42	264.35	274.25	252.84
2985	283.78	280.73	264.62	274.58	253.02
2990	283.94	280.91	264.74	274.71	253.11
2995	284.10	281.02	264.79	274.81	253.11
3000	282.34	279.31	263.31	273.15	251.92
3005	279.33	276.45	260.89	270.38	250.05
3010	276.80	274.11	259.07	268.12	248.65
3015	273.11	270.63	256.39	264.86	246.66
3020	272.65	270.37	256.53	264.59	246.97
3025	275.14	272.85	258.90	266.99	249.11
3030	277.61	275.23	261.21	269.35	250.94
3035	279.50	277.02	262.48	271.01	251.83
3040	281.25	278.61	263.58	272.53	252.56
3045	281.25	278.57	263.50	272.50	252.48
3050	281.41	278.69	263.48	272.59	252.40
3055	281.30	278.64	263.45	272.56	252.44
3060	281.16	278.53	263.39	272.48	252.40
3065	281.76	278.13	263.17	272.15	252.23
3070	281.49	278.81	263.43	272.72	252.36
3075	282.49	279.62	263.92	273.50	252.51
3080	282.40	279.53	263.70	273.40	252.40
3085	281.92	279.11	263.44	273.01	252.24
3090	280.44	277.85	262.74	271.82	251.90
3095	279.91	276.55	262.02	270.60	251.54
3100	276.62	274.53	260.92	268.70	250.99
3105	272.91	271.24	259.01	265.54	250.00
3110	271.34	269.79	258.15	264.15	249.55
3115	271.07	269.54	258.02	263.91	249.48
3120	272.67	271.04	259.07	265.35	250.13
3125	274.63	272.84	260.15	267.07	250.79
3130	277.70	275.60	261.83	269.71	251.72
3135	281.52	278.91	263.73	272.85	252.72
3140	284.47	281.43	265.20	275.23	253.51

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
3145	286.52	283.17	266.21	276.88	254.07
3150	287.44	283.93	266.77	277.66	254.43
3155	287.78	284.31	267.06	277.98	254.56
3160	287.00	283.72	266.87	277.43	254.68
3165	285.35	282.33	266.24	276.14	254.45
3170	284.48	281.63	265.94	275.53	254.37
3175	283.70	281.04	265.61	274.93	254.24
3180	283.63	280.99	265.61	274.88	254.25
3185	283.69	281.05	265.67	274.95	254.31
3190	283.74	281.11	265.72	275.00	254.35
3195	283.91	281.25	265.81	275.15	254.40
3200	283.37	280.73	265.56	274.71	254.30
3205	281.90	279.54	264.88	273.53	253.98
3210	280.51	278.34	264.22	272.38	253.65
3215	278.90	276.91	263.40	271.03	253.24
3220	278.07	276.13	262.97	270.32	253.01
3225	277.23	275.42	262.53	269.59	252.78
3230	276.90	275.12	262.35	269.30	252.68
3235	276.90	275.12	262.35	269.31	252.69
3240	277.43	275.61	262.63	269.77	252.84
3245	277.77	275.92	262.82	270.07	252.93
3250	278.63	276.68	263.26	270.81	253.17
3255	278.64	276.69	263.27	270.81	253.17
3260	278.48	276.55	263.18	270.68	253.13
3265	278.15	276.26	263.01	270.40	253.04
3270	277.48	275.65	262.66	269.82	252.85
3275	277.15	275.35	262.48	269.53	252.76
3280	277.67	275.82	262.76	269.98	252.91
3285	277.85	275.98	262.86	270.14	252.96
3290	279.20	277.18	263.55	271.29	253.31
3295	280.16	278.04	264.03	272.09	253.55
3300	280.79	278.58	264.33	272.60	253.69
3305	281.23	278.95	264.53	272.97	253.78
3310	281.22	278.95	264.52	272.96	253.77
3315	281.07	278.81	264.44	272.83	253.73
3320	281.09	278.83	264.45	272.85	253.74
3325	280.96	278.73	264.40	272.75	253.72
3330	281.14	278.89	264.50	272.90	253.77
3335	281.15	278.89	264.49	272.90	253.77
3340	281.13	278.87	264.47	272.88	253.75
3345	281.09	278.83	264.44	272.85	253.72
3350	280.73	278.51	264.23	272.53	253.60
3355	280.52	278.32	264.10	272.35	253.52

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
3360	279.71	277.61	263.69	271.67	253.30
3365	279.75	277.65	263.72	271.71	253.33
3370	280.78	278.55	264.26	272.58	253.62
3375	281.63	279.30	264.70	273.29	253.86
3380	282.27	279.85	265.02	273.83	254.03
3385	282.48	280.05	265.14	274.01	254.11
3390	282.21	279.82	265.03	273.80	254.06
3395	281.95	279.57	264.90	273.56	253.98
3400	281.32	279.03	264.61	273.06	253.85
3405	281.33	279.05	264.61	273.07	253.85
3410	281.87	278.65	264.39	272.69	253.74
3415	280.08	277.97	264.00	272.03	253.54
3420	279.27	277.25	263.59	271.36	253.34
3425	278.28	276.38	263.07	270.51	253.08
3430	276.93	275.15	262.36	269.34	252.70
3435	276.41	274.69	262.08	268.89	252.55
3440	276.25	274.54	261.99	268.74	252.50
3445	276.07	274.38	261.89	268.59	252.44
3450	275.71	274.05	261.68	268.27	252.33
3455	275.17	273.55	261.36	267.79	252.14
3460	274.23	272.67	260.80	266.94	251.81
3465	273.63	272.09	260.41	266.38	251.57
3470	272.64	271.15	259.76	265.47	251.13
3475	271.97	270.51	259.32	264.86	250.88
3480	271.07	269.64	258.68	264.01	250.45
3485	269.39	268.03	257.54	262.46	249.70
3490	268.11	266.77	256.56	261.24	248.98
3495	266.63	265.31	255.44	259.85	248.16
3500	265.37	264.07	254.47	258.68	247.44
3505	262.66	261.40	252.44	256.16	245.98
3510	261.35	260.13	251.36	254.94	245.12
3515	259.73	258.55	250.12	253.51	244.16
3520	259.07	257.84	249.52	252.85	243.67
3525	258.60	257.37	248.96	252.41	243.03
3530	257.93	256.70	248.13	251.77	242.31
3535	256.86	255.65	246.96	250.79	241.07
3540	255.30	254.13	245.46	249.41	239.67
3545	252.47	251.33	242.94	246.98	237.43
3550	249.64	248.65	240.47	244.65	235.24
3555	246.44	245.61	237.62	242.15	232.63
3560	243.69	243.11	235.14	240.25	230.29
3565	240.75	240.52	232.36	238.48	227.64
3570	239.01	239.07	231.75	237.56	226.07

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
3575	236.62	237.27	229.54	235.75	223.95
3580	234.62	235.94	226.64	235.36	222.25
3585	232.96	235.17	225.29	235.46	220.78
3590	232.04	235.02	224.54	237.15	219.98
3595	231.92	235.31	224.43	237.76	219.81
3600	231.98	235.51	224.48	238.08	219.79
3605	231.89	235.35	224.42	237.86	219.84
3610	231.81	235.14	224.39	237.57	219.87
3615	231.75	235.11	224.34	237.58	219.89
3620	231.83	235.24	224.38	237.83	219.88
3625	231.93	235.44	224.45	238.01	219.86
3630	231.91	235.20	224.46	237.58	219.92
3635	232.59	234.93	225.03	236.54	220.57
3640	235.61	236.35	227.94	236.17	223.56
3645	238.91	238.71	231.17	237.15	226.74
3650	241.59	240.90	233.97	238.39	229.53
3655	243.84	242.88	236.21	239.83	231.77
3660	245.10	244.07	237.40	240.76	232.93
3665	245.05	244.02	237.34	240.72	232.87
3670	243.16	242.27	235.51	239.38	231.09
3675	239.87	239.35	232.34	237.37	227.99
3680	236.47	236.62	229.06	235.92	224.82
3685	233.83	235.03	226.50	235.67	222.27
3690	232.78	234.88	225.27	235.32	220.91
3695	232.08	235.21	224.59	237.51	220.06
3700	232.54	236.27	224.91	238.98	220.06
3705	232.84	236.71	225.20	239.53	220.23
3710	232.80	236.65	225.16	239.46	220.18
3715	232.68	236.51	225.04	239.33	220.11
3720	233.07	237.03	225.39	239.90	220.26
3725	233.47	237.50	225.73	240.41	220.48
3730	232.98	236.94	225.30	239.83	220.25
3735	231.76	235.36	224.34	238.07	219.81
3740	231.09	233.41	224.04	235.46	220.15
3745	232.28	233.03	225.50	233.94	221.65
3750	235.16	234.61	228.39	233.94	224.46
3755	238.32	237.07	231.61	235.16	227.60
3760	243.66	242.43	237.07	239.30	233.01
3765	247.80	246.41	240.81	242.72	236.65
3770	250.62	249.32	243.39	245.29	239.08
3775	253.84	252.61	246.17	248.23	241.59
3780	254.59	253.37	246.80	248.90	242.13
3785	255.53	254.33	247.57	249.74	242.77

TABLE V-1: Continued

WAVE NUMBER	MODEL ATMOSPHERE 1	2	3	4	5
3790	255.15	253.95	247.27	249.42	242.52
3795	252.79	251.54	245.30	247.28	240.32
3800	249.78	248.44	242.79	244.53	238.47
3805	247.40	245.96	240.56	242.36	236.48
3810	245.26	243.74	238.60	240.45	234.58
3815	244.17	242.57	237.57	239.46	233.54
3820	243.23	241.51	236.69	238.63	232.66
3825	242.63	240.95	236.19	238.09	232.07
3830	242.32	240.64	235.91	237.84	231.79
3835	242.02	240.33	235.58	237.59	231.51
3840	242.34	240.65	235.82	237.85	231.81
3845	242.82	241.13	236.29	238.25	232.25
3850	243.13	241.47	236.58	238.52	232.56
3855	243.29	241.64	236.73	238.68	232.71
3860	243.61	241.99	237.04	238.96	233.02
3865	243.93	242.32	237.34	239.24	233.32
3870	244.42	242.83	237.80	239.67	233.78
3875	245.06	243.51	238.40	240.24	234.38
3880	246.03	244.53	239.31	241.13	235.29
3885	247.36	245.93	240.53	242.32	236.47
3890	250.11	248.81	243.00	244.83	238.75
3895	253.70	252.48	246.06	248.10	241.51
3900	255.93	254.74	247.90	251.10	243.06
3905	257.82	256.62	249.41	251.81	244.23
3910	258.58	257.38	250.01	252.51	244.76
3915	258.41	257.20	249.87	252.35	244.65
3920	257.84	256.63	249.43	251.84	244.30
3925	257.67	256.47	249.29	251.68	244.19
3930	257.86	256.67	249.45	251.86	244.31
3935	258.63	257.43	250.04	252.55	244.79
3940	259.97	258.77	251.08	253.77	245.53
3945	260.54	259.35	251.52	254.29	245.92
3950	263.03	261.81	253.39	256.56	247.25
3955	265.32	264.07	255.04	258.67	248.39
3960	267.41	266.12	256.50	260.62	249.33
3965	270.03	268.63	258.25	263.07	250.41
3970	272.42	270.99	259.75	265.29	251.29
3975	275.12	273.53	261.34	267.76	252.17
3980	278.42	276.52	263.13	271.64	253.13
3985	281.34	279.03	264.59	273.10	253.86
3990	282.42	280.02	265.11	273.98	254.11
3995	283.02	280.53	265.38	274.47	254.25
4000	282.88	280.41	265.32	274.36	254.22

TABLE V-1: Continued

```
      PROGRAM LOWRAD (TAPE2,TAPE4,TAPE5,TAPE6)                          10
       DIMENSION WTAU(34,8),TTAU(34)                                    20
      COMMON P(5,34),T(5,34),WH(5,34),WO(5,34),HZ1(34),HZ2(5),Z(34)     30
      COMMON C1(2581),C2(1575),C3(540),C4(133),C5(15),C8(102)           40
      COMMON TR(67),EW(67),EO(67),HZ(2),EH(8,34),VH(8),W(8),TX(9)       50
       COMMON /TRNCOM/ IATM,NL,M1,IV                                    60
C*****************************************************************     70
C       J.E.A.SELBY  AFCRL(OPI)                                         80
C      PROGRAM LOWTRAN1 CALCULATES THE TRANSMITTANCE OF THE ATMOSPHERE   90
C      FROM 350 CM-1 TO 40000 CM-1 (0.2 TO 28.6 MICRONS) AT 20 CM-1     100
C      SPECTRAL RESOLUTION ON A LINEAR WAVENUMBER SCALE.                110
C*****************************************************************    120
  400 FORMAT(3I3)                                                       130
  408 FORMAT(/28H HORIZONTAL PATH, ALTITUDE =,F7.2,11H KM,RANGE =,F7.2, 140
     1 3H KM  )                                                         150
  409  FORMAT(/50H SLANT PATH BETWEEN ALTITUDES H1 AND H2 WHERE H1 =,    160
     1 F7.2,8H KM H2 =,F7.2,18H KM,ZENITH ANGLE =,F7.3,8H DEGREES )    170
  410 FORMAT(/39H SLANT PATH TO SPACE FROM ALTITUDE H1 =, F7.2,19H KM,  180
     1ZENITH ANGLE=    ,F7.2,8H DEGREES  )                             190
  411 FORMAT(/18H MODEL ATMOSPHERE ,I1, 11H = TROPICAL)                 200
  412 FORMAT(/18H MODEL ATMOSPHERE ,I1, 21H = MIDLATITUDE SUMMER)       210
  413 FORMAT(/18H MODEL ATMOSPHERE ,I1, 21H = MIDLATITUDE WINTER)       220
  414 FORMAT(/18H MODEL ATMOSPHERE ,I1, 21H =  SUB-ARCTIC SUMMER )      230
  415 FORMAT(/18H MODEL ATMOSPHERE ,I1, 21H =  SUB-ARCTIC WINTER )      240
  416 FORMAT(/18H         HAZE MODEL ,I1, 3H = ,A5,13H VISUAL RANGE )   250
  417 FORMAT(//21H FREQUENCY RANGE V1= ,F7.2,13H CM-1 TO V2 = ,F7.1,14H 260
     1CM-1 FOR DV =,F6.1,7H CM-1  ,//)                                  270
       PLANCK(T,V) = 1.191E-12*V**3/(EXP(1.43879*V/T) - 1.0)            280
       HZ(1) = 5H23 KM                                                  290
       HZ(2) = 5H 5 KM                                                  300
       CALL READ5                                                       310
C*****************************************************************    320
```

```
C    SETTING "MODEL" =1,2,3,4,OR5 SELECTS ONE OF THE FOLLOWING MODEL ATM
C    TROPICAL,MIDLATITUDE SUMMER,MIDLATITUDE WINTER,SUBARCTIC SUMMER,OR S
C    WINTER
C
C    SETTING IHAZE=1 OR 2 SELECTS A HAZE MODEL CORRESPONDING TO 23 KM OR
C    5 KM VISUAL RANGE RESPECTIVELY
C
C        THE TYPE OF ATMOSPHERIC PATH IS INDICATED BY "ITYPE"=1,2, AND 3
C    TYPE=1, CORRESPONDS TO A HORIZONTAL (CONSTANT PRESSURE PATH)
C    TYPE=2,VERTICAL OR SLANT PATH BETWEEN TWO ALTITUDES
C    TYPE=3,VERTICAL OR SLANT PATH TO SPACE
C
C     H1=LOWEST ALTITUDE IN KMS
C     H2=HIGHEST ALTITUDE IN KMS (H2 BLANK FOR TYPE 1 OR 3     )
C     ANGLE=ZENITH ANGLE(BLANK FOR TYPE 1 )
C     RANGE=PATH LENGTH IN KMS (BLANK FOR TYPE 2 AND 3)
C
C     IV1=INITIAL FREQUENCY (WAVENUMBER) INTEGER VALUE
C     IV2=FINAL FREQUENCY(WAVENUMBER) INTEGER VALUE
C     DV= FREQUENCY INTERVALS AT WHICH TRANSMITTANCE IS PRINTED OUT
C     (DV MUST BE A MULTIPLE OF 5 )
C*****************************************************************
       NLL = NL - 1
    2  CONTINUE
       WRITE (6,426)
  426  FORMAT (1H1)
      READ(4,400) MODEL,IHAZE,ITYPE
      WRITE(6,400) MODEL,IHAZE,ITYPE
       H1 = 0.)
       H2 = 60.0
       ANGLE = 0.0
       V1 = 350.0
```

Reproduced from best available copy.

```
      V2 = 4000.0
      DV = 5.0
      IF(ITYPE.EQ.1) WRITE(6,408) H1,RANGE
      IF(ITYPE.EQ.2) WRITE(6,409) H1,H2,ANGLE
      IF(ITYPE.EQ.3) WRITE(6,410) H1,ANGLE
      M=MODEL
      IF(M.EQ.1)  WRITE(6,411) M
      IF(M.EQ.2)  WRITE(6,412) M
      IF(M.EQ.3)  WRITE(6,413) M
      IF(M.EQ.4)  WRITE(6,414) M
      IF(M.EQ.5)  WRITE(6,415) M
      WRITE(6,416) IHAZE,HZ(IHAZE)
      WRITE(6,417) V1,V2,DV
      IF(IHAZE.EQ.1) GO TO 4
      DO 3 I=1,5
    3 HZ1(I)=HZ2(I)
    4 CONTINUE
C***   NOW DEFINE CONSTANT PRESSURE PATH QUANTITES  EH(1-8)
      DO 9925 ITAU = 1, 31
      H1 = Z(ITAU)
      DO 5 I=1,NL
      PS=P(M,I)/1013.0
      TS=273.0/T(M,I)
      X=  PS*TS
      PT=PS*SQRT(TS)
      D=0.1*WH(M,I)
      EH(1,I)=D*PT**0.9
      EH(2,I)=X*PT**0.75
      EH(3,I)=WO(M,I)*PT**3.4
      EH(4,I)=1.8*PT*X
      PPW=4.56E-6*WH(M,I)*T(M,I)
      EH(5,I)=(PPW+2.205*(PS-PPW))*D
```

```
      EH(6,I)=(
      EH(7,I)=3.53365-4*471(I)
      EH(8,I)=V0(1,I)
    5 CONTINUE
      IK=1
      X=H1
    6 DO 7 I=1,NL
      J1=I
      IF(X-Z(I))8,11,7
    7 CONTINUE
    8 J2=J1
      IF(X.GT.Z(NLL)) X = Z(NLL)
      J1=J1-1
      J11=J1
      FAC=(X-Z(J1))/(Z(J2)-Z(J1))
      IF(IK.EQ.1) J11=J2
      DO 9 K=1,4
      EH(K,J11)=EH(K,J1)*(EH(K,J2)/EH(K,J1))**FAC
    9 CONTINUE
   10 IF(ITYPE.EQ.1) GO TO 12
      IF(IK.EQ.1) GO TO 11
      J2=NLL
      IF(ITYPE.EQ.3)GO TO 12
      IJK=J1
      IK=1
      X=H2
      GO TO 6
   11 J1=IJK
      J2=J2-1
   12 CONTINUE
      DO 13 K=1,8
      VH(K)=1.
```

Reproduced from
best available copy.

```
   13 CONTINUE
      IF(J2.LT.NLL) GO TO 15
C     NOW DEFINE VERTICAL PATH QUANTITIES VH(1-9)
      DO 15 K=1,9
      EV=(Z(NLL)-Z(ILL-1))*EH(K,ILL)
      IF(EH(K,ILL).EQ.E4(K,ILL-1)) GO TO 14
      EV=EV/ALOG(EH(K,ILL-1)/EH(K,ILL))
   14 VH(K)=EV+VH(K)
   15 CONTINUE
      I=NLL
      IF(J2.EQ.ILL) J2=ILL-1
   16 DO 19 I =J1,J2
      X=Z(I)
      IF(I.EQ.J1) X=H1
      DZ=Z(I+1)-X
      IF(I.EQ.J2) DZ=H2-X
      DO 19 K=1,9
      IF(E4(K,I).EQ.EH(K,I+1))GO TO 17
      EV=DZ*(E4(K,I)-E4(K,I+1))/ALOG(E4(K,I)/EH(K,I+1))
      GO TO 18
   17 EV=DZ*EH(K,I)
   18 VH(K)=VH(K)+EV
   19 CONTINUE
      GO TO 22
   20 DO 21 K=1,9
      W(K)=RANGE*EH(K,J1)
   21 CONTINUE
      GO TO 24
C     IF(ANGLE.GT.90) GO TO 24
   22 Y=3.14159*ANGLE/180.0
      DO 23 K=1,9
      W(K)=VH(K)/COS(Y)
```

```
   23 CONTINUE
C**** INSERT WITH CURVATURE AND REFRACTION
   24 CONTINUE
      DO 9925 JTAU = 1, 9
       WTAU(ITAU,JTAU) = W(JTAU)
 9925  CONTINUE
      I=1
      IV1=V1/5.
      IV2=V2/5.+.99
      IV1=5*IV1
      IV2=5*IV2
      IF(IV1.LT.35) IV1=35
      IF(IV2.GT.5000) IV2=5000
      IF(DV.LT.5.) DV=5.
      IDV=DV
      IV=IV1-IDV
      ICOUNT=0
C****  BEGINING OF TRANSMITTANCE CALCULATIONS
   30 IV=IV+IDV
      IF(ICOUNT.EQ.1) GO TO 31
      IF(ICOUNT.EQ.55) GO TO 31
      GO TO 32
   31 ICOUNT=0
       WRITE (6,+29)
   32  CONTINUE
       V = IV
       SUMTAU = 0.0
      DO 9940 KTAU = 1, 5
       FACTAU = 1.0/COS(ATAN(SQRT((2.0*KTAU - 1.0)/10.0)))
      DO 9940 TTAU = 1, 31
      DO 9933 JTAU = 1, 9
       W(JTAU) = FACTAU*WTAU(TTAU,JTAU)
```

```
9933  CONTINUE                                                          1930
      CALL TRANS                                                        1940
      TTAU(ITAU) = TX(2)                                                1950
      IF (ITAU .NE. 1)  GO TO 9935                                      1960
      SUMTAU = SUMTAU + PLANCK(T(M,1),V)*TTAU(1)                        1970
      GO TO 9939                                                        1980
9935  CONTINUE                                                          1990
      TMTAU = (T(M,ITAU) + T(M,ITAU-1))*0.5                             2000
      SUMTAU = SUMTAU + PLANCK(TMTAU,V)*(TTAU(ITAU) - TTAU(ITAU-1))     2010
9939  CONTINUE                                                          2020
9940  CONTINUE                                                          2030
      SUMTAU = 0.2*SUMTAU                                               2040
      ICOUNT = ICOUNT + 1                                               2050
      V = IV                                                            2060
      ALAM=1.0E+04/V                                                    2070
      TMTAU = 1.43879*V/ALOG(1.19106E-12*V**3/SUMTAU + 1.0)             2080
      WRITE (6,430) IV,ALAM,TMTAU,SUMTAU                                2090
      WRITE (2,430) IV,ALAM,TMTAU,SUMTAU                                2100
 430  FORMAT (1X,I6,F10.4,F8.2,1PE12.3)                                 2110
     IF(IV.GE.IV2) GO TO 76                                             2120
     GO TO 30                                                           2130
  76  CONTINUE                                                          2140
     STOP                                                               2150
     END                                                                2160
```

```
      SUBROUTINE TRANS                                                  10
      COMMON P(5,34),T(5,34),WH(5,34),WO(5,34),HZ1(34),HZ2(5),Z(34)     20
      COMMON G1(2501),G2(1275),G3(540),G4(132),G5(15),G6(102)           30
      COMMON T2(67),EW(67),EO(67),HZ(2),E1(5,34),VH(5),W(9),TX(9)       40
      COMMON /TRNCOM/ TATM,NL,M,IV                                      50
      DO 33 K=1,9                                                       60
      TX(K)=0.0                                                         70
      IF(K.LT.4) TX(K)=1.0                                              80
   33 CONTINUE                                                          90
      SUM=0.0                                                          100
      V=IV                                                             110
      I=(IV-350)/5+1                                                   120
      IF(IV.LT.1400) GO TO 41                                          130
      IF(IV.LT.2740) GO TO 48                                          140
C********    MOLECULAR SCATTERING                                     150
      C6=9.80E-28*(V**4.0117)                                          160
      TX(6)=C6*W(6)                                                    170
      SUM=SUM+TX(6)                                                    180
      IF(IV.LT.9200) GO TO 33                                          190
      IF(IV.LT.13000)GO TO 50                                          200
C********  UV OZONE                                                   210
      IF(IV.LE.23400) GO TO 34                                         220
      IF(IV.GE.27500) GO TO 35                                         230
      GO TO 70                                                         240
   34 XX=200.0                                                         250
      XI=(V-13000.0)/XX+1.0                                            260
      L1=1                                                             270
      L2=53                                                            280
      GO TO 36                                                         290
   35 XX=500.0                                                         300
      XI=(V-27000.0)/XX+53.0                                           310
      L1=67                                                            320
```

```
      L2=102
   36 DO 37 N=L1,L2
      XD=XI-FLOAT(N)
      IF(XD)39,39,37
   37 CONTINUE
   38 TX(8)=W(8)*C8(N)
      GO TO 40
   39 TX(8) =C8(N)+XD*(C8(N)-C8(N-1))
      TX(8)=W(8)*TX(8)
   40 SUM=SUM+TX(8)
      A=C8(N)
      IF(IV.GT.14500) GO TO  70
      GO TO  50
C********     WATER VAPOUR CONTINUUM
   41 IF(IV.LE.670) GO TO 53
      IF(IV.LT.700) GO TO 46
   42 XI=(V-700.)/50.+1.
      DO 43 NH=1,15
      XH=XI-FLOAT(NH)
      IF(XH)45,44,43
   43 CONTINUE
   44 TX(5)=C5(NH)
      GO TO 47
   45    TX(5)=C5(NH)+XH*(C5(NH)-C5(NH-1))
      GO TO 47
   46 TX(5)=(V-670.)*0.83
   47 TX(5)=W(5)*TX(5)
      SUM=SUM+TX(5)
      A=C5(NH)
      GO TO 53
C********   NITROGEN CONTINUUM
   48 IF(IV.LT.2080) GO TO 53
```

```
      K4=I-346
      TX(4)=C4(K4)*W(4)
      SUM=SUM+TX(4)
      A=C4(K4)
      GO TO  53
C********  WATER VAPOUR
   50 IF(IV.LT.12800.AND.IV.GE.12275) GO TO 51
      IF(IV.LE.14520.AND.IV.GE.13400) GO TO 52
      GO TO 57
   51 I=I-135
      GO TO 53
   52 I=I-255
   53 K1=1
      WS1=ALOG10(W(1))+C1(T)
      IF(WS1.LT.-2.3468) GO TO 57
      IF(WS1.GT.3.5682) GO TO 56
      IF(WS1.GT.2.0) K1=40
      DO 54 K=K1,67
      IF(WS1.LE.FW(K)) GO TO 55
   54 CONTINUE
   55 TX(1)= TR(K)+(TR(K-1)-TR(K))* (FW(K)-WS1)/(FW(K)-FW(K-1)    )
       GO TO 57
   56 TX(1)=0.0
   57 CONTINUE
C********   UNIFORMLY MIXED GASES
      IF(IV.LT.3060.AND.IV.GE.500) GO TO 58
      IF(IV.LT.13190.AND.IV.GT.12870) GO TO 59
       GO TO 60
   58 J=I-30
      GO TO 60
   59 J=(IV-12950)/5+1516
   60 CONTINUE
```

Reproduced from best available copy.

Reproduced from best available copy.

```
      K1=1                                                                    970
      WS2=ALOG10(W(2))+C2(J)                                                  980
      IF(WS2.LT.-2.3468)GO TO 64                                              990
      IF(WS2.GT.3.5582) GO TO 63                                             1000
      IF(WS2.GT.2.0) K1=46                                                   1010
      DO 61 K=K1,67                                                          1020
      IF(WS2.LE.FW(K)) GO TO 62                                              1030
   61 CONTINUE                                                               1040
   62 TX(2)= T2(K)+(T2(K-1)-T2(K))*(FW(K)-WS2)/(FW(K)-FW(K-1))               1050
       GO TO 64                                                             1060
   63 TX(2)=0.0                                                             1070
   64 CONTINUE                                                              1080
C********  OZONE                                                           1090
      IF(IV.LT.5750.OR.IV.GT.38700) GO TO 68                                1100
      L=I-45                                                                1110
      K1=1                                                                  1120
      WS3=ALOG10(W(3))+C3(L)                                                1130
      IF(WS3.LT.-1.6778)GO TO 68                                            1140
      IF(WS3.GT.3.9345) GO TO 67                                            1150
      IF(WS3.GT.1.5) K1=36                                                  1160
      DO 65 K=K1,67                                                         1170
      IF(WS3.LE.FO(K)) GO TO 66                                             1180
   65 CONTINUE                                                              1190
   66 TX(3)= T2(K)-(T2(K)-T2(K-1))*(FO(K)-WS3)/(FO(K)-FO(K-1))              1200
      GO TO 68                                                              1210
   67 TX(3)=0.0                                                             1220
   68 CONTINUE                                                              1230
C********   AEROSOL EXTINCTION                                             1240
   70  C7=3.75+E-5*(V**2.0543)                                              1250
      IF(IV.LT.30000) GO TO 71                                              1260
      GO TO 72                                                              1270
   71 C7=C7+(2.8E-7)*(30000.0-V)                                            1280
```

```
 72 TX(7)=C7*W(7)                                                        1290
     SUM=SUM+TX(7)                                                       1300
    TX(9)=SUM                                                            1310
443 FORMAT(  3(I3,F10.7),5E13.3)                                         1320
    DO 75 K=4,9                                                          1330
    IF(TX(K).EQ.0.0) GO TO 74                                            1340
    IF(TX(K).LE.0.1) GO TO 73                                            1350
    TX(K)=EXP(-TX(K))                                                    1360
    GO TO 75                                                             1370
 73 TX(K)=1.0-TX(K)+0.5*TX(K)*TX(K)                                      1380
    GO TO 75                                                             1390
 74 TX(K)=1.0                                                            1400
 75 CONTINUE                                                             1410
    TX(9)=TX(1)*TX(2)*TX(3)*TX(9)                                        1420
    IF(IV.GE.13000) TX(7)=TX(8)                                          1430
     RETURN                                                              1440
     END                                                                 1450
```

```
      SUBROUTINE READ F                                                  10
      COMMON P(5,34),T(5,34),WH(5,34),WO(5,34),H71(34),HZ2(5),Z(34)      20
      COMMON C1(2580),C2(1575),C3(540),C4(133),C5(15),C8(102)            30
      COMMON TP(67),EW(67),EO(67),HZ(2),EH(8,34),VH(8),W(8),TX(8         40
      COMMON /TRNCOM/ IATM,NL,M,IV                                       50
400   FORMAT(3I3)                                                        60
401   FORMAT(7E10.3)                                                     70
402   FORMAT(F6.1,2(E10.3,F6.1,2E10.1))                                  80
403   FORMAT(F5.1,E10.3,F7.1,2E10.1,E14.4)                               90
404   FORMAT(+(F6.3,2F7.4))                                             100
405   FORMAT( 15F5.2)                                                   110
406   FORMAT(8F9.2)                                                     120
      HZ(1) = 5423 K1                                                   130
      HZ(2) = 54 5 KM                                                   140
      READ (5,400) IATM,NL                                              150
     IF(IATM.NE.5) GO TO 501                                            160
      READ (5,401)(HZ2(I),I=1,5)                                        170
     J=-1                                                               180
     DO 500 KP=1,2                                                      190
     K1=J+2                                                             200
     K2=K1+1                                                            210
     J=1                                                                220
     DO 500 I=1,NL                                                      230
500   READ (5,402)Z(I),(P(K,I),T(K,I),WH(K,I),WO(K,I),K=K1,K2)          240
501 CONTINUE                                                            250
     K=IATM                                                             260
     DO 502 I=1,NL                                                      270
502   READ (5,403)Z(I),P(K,I),T(K,I),WH(K,I),WO(K,I),H71(I)             280
      READ (5,404)(TP(I),EW(I),EO(I),I=1,67)                            290
      READ (5,405)(C1(I),I=1,2580)                                      300
      READ (5,405)(C2(I),I=1,1575)                                      310
      READ (5,405)(C3(I),I=1,540)                                       320
```

```
      READ (5,406)(C4(I),I=1,133)
      READ (5,405)(C5(I),I=1,15)
      READ (5,406) (C8(I),I=1,102)
      RETURN
      END
```

Program SPCTRA

Purpose: SPCTRA computes the spectral radiances in the 5 to 25μm interval corresponding to the band radiances computed by BCKGND. The input data to SPCTRA are the molecular band line constants, molecular vibrational level and band constants used by, and vibrational population and band radiance data generated by BCKGND. In its present form, SPCTRA computes spectral radiances between 2.5 and 25μm for limb views and endo-atmospheric zenith angles of 0 and 90 degrees at 5 km intervals between 60 and 200 km.

Program Flow: When program execution begins, values are assigned to the integer variables NOUT, NLN and NALT and the subroutine ROTATE, which reads in data for the molecular lines, is called. The outer DO range ending with statement 500 is entered. The DO parameters are fixed at present to produce two sets of spectra, using night and day input from program BCKGND. The program may be made more flexible by reading in DO parameters prior to execution of the DO statement. The variable BLAM is set to 5.0 and the outer DO range ending with statement 105 is entered. The effect of the first two statements is to set up a table of wavenumbers at 2 cm^{-1} intervals between 2.5 and 25μm in real array AVLAM. These are the finest wavenumbers at which spectral radiances may be printed out. The inner DO range ending with statement 105 sets the values of output spectral radiances equal to zero.

The DO range ending with statement 110 sets temperature values in array TEMP to those corresponding to the U.S. Standard Atmosphere, 1962. Temperature values may also be read in prior to execution of the first DO statement. Following final execution of this DO range, temperatures are printed out.

Next, the number of molecular species is read in and stored in integer variable NSPTOT. The outer DO range ending with statement 400 is entered and executed NSPTOT times. Within this DO range, molecular vibrational level and band data and vibrational population data are read in and mean temperatures and total number of molecules per unit area in the vertical and horizontal directions are computed for each vibrational state. Total band radiances corresponding to limb viewing and vertical and horizontal endo-atmospheric viewing are read in. Relative line radiances are computed and normalized to the total band radiances. The spectra corresponding to the normalized line radiances are computed and stored in arrays SPECA, SPECB and SPECC. Details of these computations are given below.

The code number of the species ISP is read in. The number of vibrational levels NLEVEL, the number of bands NBAND and the molecular weight RM are read in. In the DO range ending with statement 210, the level description codes LSC and LCC, the energy and the mean rotational constant for each vibrational level are read in. In the DO range ending with statement 220 the band description codes LBC and LBU and the band strength

for each band are read in. The number of altitudes NALT for which vibrational populations are computed for the species and the number of altitudes NRLEV for which band radiances are computed by BCKGND are read in. Vibrational level populations are read in and stored in array CON.

The mean temperatures and total number of molecules per square centimeter in the vertical and horizontal directions at 5 km intervals between 60 and 200 km are determined in the DO range ending with statement 129. The product of temperature and vibrational level population is computed in the DO range ending with statement 112. The integrals of total number of molecules and the product of temperature and number density in the vertical direction are computed in the DO range ending with statement 120. Simpson's rule integration is used. The total number of molecules in a given vibrational level above a given altitude is stored in array COLCNT. The corresponding mean temperatures, obtained by dividing the integral of the product of temperature and number density by the integral of number density, are stored in array AVTEMP. After exit from DO range, the column counts and temperatures are printed out.

The DO range ending with statement 128 performs a similar computation for column counts and mean temperatures in the horizontal direction. The integrations required are performed in function QINT. Upon exit from this DO range, column counts stored in array HORCNT and mean temperatures stored in array BVTEMP are printed out.

The inner DO range ending with statement 400 is now entered and executed for each band. This DO range controls the computation of the spectra.

The altitude HT, the limb band radiance RADC, and endoatmospheric radiances RADA (upward direction) and RADB (horizontal direction) are read in the DO range ending with statement 230. Band information is computed from band and level constants. A test is made on the species code ISP to determine whether the molecule is linear or nonlinear.

Following the test of ISP, if the molecule is linear and is not in McClatchey's tabulation, line positions are computed in subroutine ROTPOS. The DO range ending with statement 250 is entered and executed for each of the 20 altitudes for which spectra are computed. Line radiances are computed in subroutine ROTSTR for vertical endo-atmospheric viewing. Successive calls to subroutine FILTER compute the spectral contributions from lines in the P, Q and R branches. The sequence of calls to ROTSTR and FILTER is repeated for endo-atmospheric viewing in the horizontal direction and for limb viewing. Upon exit from this DO range, control is transferred to statement 280.

If the test on ISP determines that the molecular line data is available control is transferred to statement 260 and the DO range ending with statement 270 is executed for each of the 28 altitudes for which spectra are computed. Spectra are generated from the line radiances by calls to subroutine WATSTR and FILTER, for each of the three

viewing conditions.

Following the final execution of the DO ranges ending with statement 400, the spectra are printed out in the order endo-atmospheric viewing in the upward direction, endo-atmospheric viewing in the horizontal direction and limb viewing.

Upon final execution of the DO range ending with statement 500, the program halts with a STOP statement.

Mnemonic Variables:

AVLAM	Real array in which output wavenumbers are stored.
AVTEMP	Real array in which mean temperatures in the vertical direction are stored.
BV	Real array in which rotational constants for each vibrational level are stored.
BVTEMP	Real array in which mean temperatures in the horizontal direction are stored.
COLCNT	Real array in which total number of molecules per square centimeter above 60, 65, ... 200 km in each vibrational level are stored.
CON	Real array in which number densities for each vibrational level are stored.
HHOLAM	Real array in which wavenumbers of band lines are stored.
HHOLSE	Real array in which lower state energies of band lines are stored.
HHOSTR	Real array in which strengths of lines are stored.
HORCNT	Real array in which total number of molecules per square centimeter in each vibrational level in the horizontal direction at altitudes 60, 65, ... 200 km are stored.

HRAD	Real array in which line radiances are stored.
P	Real array in which wavelengths of P branch of linear molecule are returned by subroutine ROTPOS.
PRAD	Real array in which products of number density and temperature are stored.
PS	Real array in which P branch line radiances are returned by subroutine ROTSTR.
Q	Real array in which Q branch line wavelengths are returned by subroutine ROTPOS.
QS	Real array in which Q branch line radiances are returned by subroutine ROTSTR.
R	Real array in which R branch line wavelengths are returned by subroutine ROTPOS.
RADA	Real array into which vertical direction and endo-atmospheric band radiances are read.
RADB	Real array into which horizontal direction endo-atmospheric band radiances are read.
RADC	Real array in which limb view band radiances are read.
RCM	Real array into which vibrational level energies are read.
ROTEA	Real array used by subroutines ROTATE and WATSTR, stores factors needed to compute line radiances in water vapor pure rotational spectrum.
ROTLAM	Real array in which water vapor pure rotational line wavelengths are stored.
ROTLSE	Real array in which lower state energies of water vapor pure rotational lines are stored.
ROTST	Real array in which absorption strengths of water vapor pure rotational lines are stored.
RS	Real array in which R branch line radiances are returned by subroutine ROTSTR.

SPECA	Real array in which spectral radiances for vertical viewing endo-atmospheric geometry are generated.
SPECB	Real array in which spectral radiances for endo-atmospheric horizontal viewing are generated.
SPECC	Real array in which spectral radiances for limb viewing are generated.
STR	Real array used to store band strengths.
TEMP	Real array used to store atmospheric temperatures.
BLAM	Real variable used as temporary storage during generation of values in array AVLAM.
CENTER	Real variable used to store wavenumber of band center.
HT	Real variable into which altitudes at which band radiances are computed by BCKGND are read.
H1, H2, H3, H4, H5	Real variables used to store distance increments used in computing total column counts and mean temperatures in the horizontal direction.
RBASE	Real variable used to store distance of base altitude from center of earth in computing horizontal column counts.
RB2	Real variable used to store value of RBASE**2.
RM	Real variable used to store value of molecular weight.
RTOP	Real variable used to store distance from center of earth of given altitude level.
SNB	Real variable used to store current value of band strength.
STH	Real variable used as temporary storage is computing horizontal column counts.
SVH	Real variable used as temporary storage in computing horizontal column counts.
TC	Real variable used to accumulate integral of number density in computing vertical column counts.

TCH	Real variable used to accumulate integral of number density in computing horizontal column counts.
TP	Real variable used to accumulate integral of product of temperature in the vertical direction.
TPH	Real variable used to accumulate integral of product of temperature and number density in computing mean temperature in the horizontal direction.
LBC	Integer array storing level numbers of initial and final states of a band.
LBU	Integer array storing band type and symmetry information.
LCC	Integer array storing vibrational level information.
LSC	Integer array storing vibrational level information.
I	Integer variable used as index in DO loops.
ISP	Integer variable into which species identification code is read (2 for CO_2, 3 for H_2O, 4 for NO, 5 for N_2O and 6 or 7 for ozone).
J	Integer variable used as index in DO loops.
K	Integer variable (set equal to 28 in current program) used as final parameter in DO statement controlling computation of column counts and mean temperatures. The value of K is the number of altitudes for which spectra are computed.
KB	Integer variable used to store band type code.
KK	Integer variable used as final parameter in DO range ending with statement 125, computed from K and I.
KL	Integer variable used to store band symmetry code.
NA	Integer variable used as altitude level index in computing horizontal column counts.
NALT	Integer variable storing total number of altitude levels.

NB	Integer variable used as index for number of current band.
NBAND	Integer variable storing total number of bands considered for a given species.
NBC	Integer variable used to store current value of LBC.
NCASES	Integer variable used as index for spectrum set.
NF	Integer variable used as index for lower vibrational state of band.
NI	Integer variable used as index for upper vibrational state of band.
NL	Integer variable used as index for vibrational level number.
NLEVEL	Integer variable in which is stored total number of vibrational levels for a molecular species.
NLN	Integer variable controlling number of lines treated in each branch of a band of a linear molecule, set to 100 in current program.
NRLEV	Integer variable into which is read number of altitudes at which band radiances for a given species are computed by BCKGND.
NSP	Integer variable used as index for DO range executed NSPTOT times.
NSPTOT	Integer variable into which is read total number of species to be included in a spectrum.
NZ	Integer variable used as altitude index in DO loops.

```
      PROGRAM SPCTRA (OUTPUT,TAPE1,TAPE2=OUTPUT,                        10
     1  TAPE4,TAPE5)                                                    20
      DIMENSION LSC(25),LCC(25),RCM(25),BV(25),LBC(25),LBU(25),STR(25)  30
      DIMENSION P(100),Q(100),R(100),PS(100),QS(100),RS(100)            40
      DIMENSION CON(201,10),TEMP(201),PROD(201)                         50
      DIMENSION AVTEMP(40,25),BVTEMP(40,25)                             60
      DIMENSION COLCNT(40,25),HORCNT(40,25)                             70
      DIMENSION RADA( 50),RADB( 50),RADC( 50)                           80
      COMMON /ISPCOM/ ISP                                               90
      COMMON WVNUM(2500),ENLS(2500),STRL(2500),HRAD(2500)              100
      COMMON SPEC(1801)                                                110
      NLN = 100                                                        120
      READ (1,12) TEMP                                                 130
     DO 500 NCASES = 1, 2                                              140
      REWIND 5                                                         150
     DO 106 I = 1, 1801                                                160
      SPEC(I) = 0.0                                                    170
106   CONTINUE                                                         180
      READ (1,6) NSPTOT                                                190
     DO 400 NSP = 1, NSPTOT                                            200
      READ (1,6) ISP                                                   210
      READ (1,41) NLEVEL,NBAND,RM                                      220
     DO 210 I = 1, NLEVEL                                              230
      READ (1,41) LSC(I),LCC(I),RCM(I),BV(I)                           240
210   CONTINUE                                                         250
     DO 220 I = 1, NBAND                                               260
      READ (1,41) LBC(I),LBU(I),STR(I)                                 270
220   CONTINUE                                                         280
      READ (1,6) NALT,NRLEV                                            290
     DO 221 NZ = 1, NALT                                               300
      READ (1,42) (CON(NZ,NL), NL = 1, NLEVEL)                         310
221   CONTINUE                                                         320
```

```
C       COMPUTE MEAN TEMPERATURES AND COLUMN COUNTS                      330
        K = NRLEV                                                        340
       DO 129 NL = 1, NLEVEL                                             350
       DO 112 I = 1, NALT                                                360
        PROD(I) = CON(I,NL)*TEMP(I)                                      370
  112   CONTINUE                                                         380
        TP = 0.0                                                         390
        TC = 0.0                                                         400
        NA = NALT                                                        410
       DO 120 I = 1, K                                                   420
        TP = TP + SIMP(PROD(NA - 2),1.0E+5,2)                            430
     1   + SIMP(PROD(NA - 3),1.0E+5,3) + SIMP(PROD(NA - 5),1.0E+5,3)     440
        TC = TC + SIMP(CON(NA - 2,NL),1.0E+5,2)                          450
     1    + SIMP(CON(NA - 3,NL),1.0E+5,3) + SIMP(CON(NA - 5,NL),1.0E+5,3) 460
        COLCNT(K - I + 1,NL) = TC                                        470
        AVTEMP(K - I + 1,NL) = TP/TC                                     480
        NA = NA - 5                                                      490
        HORCNT(I,NL) = 0.0                                               500
  120   CONTINUE                                                         510
        RBASE = 6.431E+8                                                 520
       DO 128 I = 1, K                                                   530
        TPH = 0.0                                                        540
        TCH = 0.0                                                        550
        NA = NALT                                                        560
        RB2 = RBASE*RBASE                                                570
        RTOP = 6.431E+8 + NALT*1.0E+5                                    580
        SVH = SQRT(RTOP*RTOP - RB2)                                      590
        KK = K - I + 1                                                   600
       DO 125 J = 1, KK                                                  610
        RTOP = RTOP - 1.0E+5                                             620
        STH = SQRT(RTOP*RTOP - RB2)                                      630
        H5 = SVH - STH                                                   640
```

```
      SVH = STH                                                          650
      RTOP = RTOP - 1.0E+5                                               660
      STH = SQRT(RTOP*RTOP - RB2)                                        670
      H4 = SVH - STH                                                     680
      SVH = STH                                                          690
      RTOP = RTOP - 1.0E+5                                               700
      STH = SQRT(RTOP*RTOP - RB2)                                        710
      H3 = SVH - STH                                                     720
      SVH = STH                                                          730
      RTOP = RTOP - 1.0E+5                                               740
      STH = SQRT(RTOP*RTOP - RB2)                                        750
      H2 = SVH - STH                                                     760
      SVH = STH                                                          770
      RTOP = RTOP - 1.0E+5                                               780
      STH = SQRT(ABS(RTOP*RTOP - RB2))                                   790
      H1 = SVH - STH                                                     800
      SVH = STH                                                          810
      TPH = TPH + QINT(PROD(NA - 2),H4,H5,2)                             820
     1  + QINT(PROD(NA - 3),H3,H4,3) + QINT(PROD(NA - 5),H1,H2,3)         830
      TCH = TCH + QINT(CON(NA - 2,NL),H4,H5,2)                           840
     1  + QINT(CON(NA - 3,NL),H3,H4,3) + QINT(CON(NA - 5,NL),H1,H2,3)     850
      NA = NA - 5                                                        860
  125 CONTINUE                                                           870
      HORCNT(I,NL) = TCH                                                 880
      BVTEMP(I,NL) = TPH/TCH                                             890
      RBASE = RBASE + 5.0E+5                                             900
  128 CONTINUE                                                           910
  129 CONTINUE                                                           920
C     ****** SET J VALUE                                                930
      J = 1                                                              940
C     ****** OR SET UP DO LOOP                                          950
      IF (ISP .NE. 3)  GO TO 225                                         960
```

```
      CNTHR = 2.0*HORCNT(J,1)                                              970
      CALL ROTATE (5,LINES)                                                980
C     ROTATE WILL READ FROM TAPE5                                          990
      CALL RATSTR (RADC(J),RM,BVTEMP(J,1),CNTHR,1,LINES)                  1000
      CALL FILTER (SPEC,HRAD,WVNUM,LINES)                                 1010
  225 CONTINUE                                                            1020
      DO 400 NB = 1, NBAND                                                1030
      DO 230 I = 1, NRLEV                                                 1040
      READ (1,43) HT,RADC(I),RADA(I),RADB(I)                              1050
  230 CONTINUE                                                            1060
      NBC = LBC(NB)                                                       1070
      NI = MOD(NBC/100,100)                                               1080
      NF = MOD(NBC,100)                                                   1090
      CENTER = RCM(NI) - RCM(NF)                                          1100
      SNB = STR(NB)*EXP(1.43879*RCM(NF)/300.0)                            1110
      KL = MOD(LBU(NB)/10,10)                                             1120
      KB = MOD(LBU(NB),10)                                                1130
      IF (ISP .NE. 4)  GO TO 260                                          1140
      CALL ROTPOS (CENTER,BV(NI),BV(NF),P,Q,R,1)                          1150
C     CALL ROTSTR (RADA(J),RM,BV(NF),AVTEMP(J,NF),COLCNT(J,NF),           1160
C    1  SNB,PS,QS,RS,KB,1,P,Q,R)                                          1170
C     CALL FILTER (SPECA,PS,P,NLN)                                        1180
C     CALL FILTER (SPECA,QS,Q,NLN)                                        1190
C     CALL FILTER (SPECA,RS,R,NLN)                                        1200
C     CALL ROTSTR (RADB(J),RM,BV(NF),BVTEMP(J,NF),HORCNT(J,NF),           1210
C    1  SNB,PS,QS,RS,KB,1,P,Q,R)                                          1220
C     CALL FILTER (SPECB,PS,P,NLN)                                        1230
C     CALL FILTER (SPECB,QS,Q,NLN)                                        1240
C     CALL FILTER (SPECB,RS,R,NLN)                                        1250
      CALL ROTSTR (RADC(J),RM,BV(NF),BVTEMP(J,NF),2.0*HORCNT(J,NF),       1260
     1  SNB,PS,QS,RS,KB,1,P,Q,R)                                          1270
      CALL FILTER (SPEC,PS,P,NLN)                                         1280
```

```
      CALL FILTER (SPEC,QS,Q,NLN)                                                    1290
      CALL FILTER (SPEC,RS,R,NLN)                                                    1300
250   CONTINUE                                                                       1310
      GO TO 400                                                                      1320
260   CONTINUE                                                                       1330
      IF (NR .GT. 10 .AND. ISP .EQ. 3)  GO TO 400                                    1340
      CALL ROTATE (5,LINES)                                                          1350
      CALL WATSTR (RADC(J),RM,BVTEMP(J,NF),2.0*HORCNT(J,NF),2,LINES)                 1360
      CALL FILTER (SPEC,HRAD,WVNUM,LINES)                                            1370
400   CONTINUE                                                                       1380
      WRITE (2,16)                                                                   1390
      WRITE (2,8) SPEC                                                               1400
      WRITE (4,8) SPEC                                                               1410
500   CONTINUE                                                                       1420
      STOP                                                                           1430
  6   FORMAT (6I6)                                                                   1440
  8   FORMAT (1P6E11.3)                                                              1450
 14   FORMAT (1X)                                                                    1460
 12   FORMAT (1P6E12.4)                                                              1470
 16   FORMAT (1H1)                                                                   1480
 41   FORMAT (2I6,5E12.4)                                                            1490
 42   FORMAT (8E10.4)                                                                1500
 43   FORMAT (F6.1,3E17.4)                                                           1510
      END                                                                            1520
```

Subroutine FILTER

Purpose. FILTER computes the contribution to spectral radiance from each line in a band or sub-band, in the interval 2.5μm to 25μm. Provision is made for varying resolution in either wavelength or wavenumber space. Sufficient storage is provided to compute radiances at 2 wavenumber intervals with triangular slit resolution of 5 wavenumbers.

Method: FILTER searches an array of emission line strengths and assigns a weighted strength to an output wavenumber or wavelength.

Program Flow: The outer DO range ending with statement 150 controls the distribution of spectral radiance from the Ith of N given lines, and computes the indexes JS and JF of the first and last output points in array SPEC to which a given line contributes. The inner DO range ending with statement 120 computes the contribution of a given line to spectral radiance at output wavelengths or wavenumbers.

Inputs: RAD, WVLTH, N, KOMPUT

Output: SPEC

Mnemonic Variables:

SPEC Real array into which spectra radiances are added at regular intervals between 2.5 and 25μm or 400 and 4000 wavenumbers.

RAD	Real array containing radiances of individual lines.
WULTH	Real array containing wavelengths of individual lines.
N	Integer variable containing number of lines in band or sub-band for which spectral radiance is to be computed.
KOMPUT	Integer variable which determines number of output spectral points and whether wavelength or wavenumber output is made.
RADT	Real variable used as temporary storage for radiance value of a single line.
RLAM	Real variable used temporary storage for wavelength value at which spectral radiance is computed.
RT	Real variable used as temporary storage for wavelength of a single line.
I	Integer variable used as index for outer DO loop which treats each line in turn.
J	Integer variable used as index for inner DO loop which assigns spectral contribution of each line to output wavelength.
JF	Integer variable used as final parameter in inner DO loop, index of greatest output wavelength for a given line.
JS	Integer variable used as initial parameter in inner DO loop, index of shortest output wavelength for a given line.

```
      SUBROUTINE FILTER (SPEC,RAD,WVNUM,N)                              10
C     FIVE WAVENUMBER RESOLUTION, TWO WAVENUMBER INTERVAL               20
      DIMENSION SPEC(1801),RAD(450),WVNUM(450)                          30
      COMMON /ISPCOM/ ISP                                               40
      DO 150 I = 1, N                                                   50
      RT = WVNUM(I)                                                     60
      IF (ISP .EQ. 4)  RT = 1.0E+4/RT                                   70
      RADT = RAD(I)                                                     80
      JS = (RT - 405.0)*0.5 + 2.0                                       90
      JF = (RT - 395.0)*0.5 + 1.0                                      100
      IF (JS .LT. 1)  JS = 1                                           110
      IF (JF .GT. 1801)  JF = 1801                                     120
      IF (JS .GT. JF)  GO TO 150                                       130
      DO 120 J = JS,JF                                                 140
      RNUM = (J - 1)*2.0 + 400.0                                       150
      SPEC(J) = SPEC(J) + (5.0 - ABS(RNUM - RT))*RADT*0.04             160
  120 CONTINUE                                                         170
  150 CONTINUE                                                         180
      RETURN                                                           190
      END                                                              200
```

Subroutine ROTATE

Purpose: Subroutine ROTATE is used to read in spectral line data from the compilation of McClatchey, et al. The input data consists of line positions and strengths for vibrational - rotational bands of water, carbon dioxide, ozone and nitrous oxide, and for the pure rotational spectrum of water. This data is processed for use by other subroutines.

Program Flow: The DO range ending with statement 110 reads in data on the strongest lines in a vibration - rotation band. This data consists of the wave number of the line, the line strength at 296 K and the wavenumber of the lower rotational state. ROTATE converts the wavenumber to wavelength in μm stored in array HHOLAM, converts the line strength to units cm^{-1}/atm - cm at 273.15k stored in array HHOSTR and stores the wavenumber of the lower state in array HHOLSE.

The rotational partition function for water vapor is computed next if water vapor data is being read. The DO range ending at statement 130 is used only for water vapor pure rotational lines and computes the wavelength (stored in array ROTLAM), Einstein A coefficient (stored in array EA), relative line strength in absorption at 273.15k (stored in array ROTST), relative line strength in emission at 273.15k (stored in array ROTEA) and the wavenumber of the lower rotational state (stored in array ROTLSE).

Mnemonic Variables:

EA	Real variable used to store Einstein A coefficient for water vapor pure rotational spectrum line.
EALC	Real variable used to store intermediate results in computing pure rotational line strengths in emission.
EALR	Real variable used to store intermediate results in computing pure rotational line strengths in absorption.
ENERGY	Real variable into which is read wavenumbers of pure rotational lines.
ENL	Real variable into which is read wavenumbers of lower states of pure rotational lines.
ENV	Real variable into which is read wavenumber of upper state of pure rotational line.
HHOLAM	Real array in which wavelengths of vibration - rotation band lines are stored.
HHOLSE	Real array in which wavenumbers of lower states of a vibration - rotation band are stored.
HHOSTR	Real array in which line strengths of vibration - rotation band are stored (cm^{-2} - $\mu\tau m^{-1}$ STP).
ROTEA	Real array in which pure rotational line strengths in emission (273.15k) are stored.
ROTLAM	Real array in which pure rotational line wavelengths are stored.
ROTLSE	Real array in which lower state wavenumber of pure rotational lines are stored.
ROTST	Real array in which pure rotational line strengths in absorption are stored.

WVLTH	Real array in which wavelengths of vibration - rotation band lines are stored (redundant).
EAFAC	Real variable used to store pre-factor in computing Einstein A coefficients for pure rotational lines.
QR	Real variable in which is stored approximate rotational partition function for water vapor molecule.
QRFAC	Real variable into which is read prefactor for computing rotational partition function.
RA	Real variable into which is stored first water vapor rotational constant.
RB	Real variable into which is stored second water vapor rotational constant.
RC	Real variable into which is stored third water vapor rotational constant.
TEFAC	Real variable in which is stored exponential factor used in converting line intensities at 296K to line strengths at 273.15K.
TQFAC	Real variable in which is stored factor relating rotational function at 296K to that at 273.15K.
I	Integer variable used as index in DO loop.
NLINES	Integer variable into which is read number of lines included in pure rotational spectrum and in any single vibration - rotation band.

```
      SUBROUTINE ROTATE (NIN,LINES)                                      10
      COMMON /ISPCOM/ ISP                                                20
      COMMON WVNUM(2500),ENLS(2500),STR(2500),HRAD(2500)                 30
      READ (NIN,2) LINES                                                 40
      DO 120 I = 1, LINES                                                50
      READ (NIN,4) WVNUM(I),STR(I),ENLS(I)                               60
      STR(I) = 2.4793E+19*STR(I)                                         70
  120 CONTINUE                                                           80
      RETURN                                                             90
    2 FORMAT (1X,I8)                                                    100
    4 FORMAT (F10.3,1PE10.3,5X,0PF10.3)                                 110
      END                                                               120
```

Subroutine WATSTR

Purpose: WATSTR computes the radiances of individual lines in vibration - rotation bands and in the pure rotational spectrum of water vapor. The radiances are computed with the assumption that the atmosphere in the line of sight may be treated as having uniform temperature. The radiances computed for the vibration - rotation bands are scaled to match the total band radiances produced by the BACKGROUND program. Absolute radiance values are computed for the water vapor pure rotational spectrum.

Program Flow: Upon entry to WATSTR, a conversion factor TQFAC is computed for later use in computing the effect of charging rotational partition function with temperature. Control is then transferred to the appropriate part of the subroutine as determined by the value of N.

For computation of the radiances of lines in the water vapor pure rotational spectrum (N = 1), control is transferred to statement 100. A radiance factor RADFAC and an optical thickness factor TAUFAC are computed. Then the optical thickness and radiance values corre cted for optical thickness and induced emission are computed for each line in the DO range ending with statement 110. Control is returned to the main program. Line constants generated by subroutine ROTATE are available in the

arrays of common area ROTCOM. Radiance values are returned in array HRAD.

Control is transferred to statement 200 for computation of radiances of lines in the other bands. An optical thickness factor TAUFAC and factor TEFAC used in computing populations of rotational states at temperature T are computed. The variable SUM, used to accumulate the sum of relative radiances, is set to zero. In the DO range ending with statement 210, the optical thickness of each line is computed and the relative radiance value, corrected for optical thickness, is computed and stored in array HRAD. The sum relative radiances is accumulated in SUM. Upon exit from this DO range, the total radiance RAD is divided by SUM and stored in RADFAC. In the DO range ending with statement 220, absolute radiance values are obtained and stored in array HRAD. Control is then returned to the main program.

Mnemonic Variables:

RAD	Real variable containing total band strength for vibration - rotation band.
RM	Real variable containing molecular weight of molecule.
T	Real variable containing value of temperature.
DEN	Real variable containing total number of ground state molecules per square centimeter in line of sight.
N	Integer variable determining whether entry to WATSTR results in computation for water vapor pure rotational spectrum (N = 1) or other band (N = 2).

HHOLAM	Real array, values as in ROTATE.
HHOLSE	Real array, values as in ROTATE.
HHOSTR	Real array, values as in ROTATE.
HRAD	Real array used to return radiance values of individual lines.
ROTEA	Real array, values as in ROTATE.
ROTLAM	Real array, values as in ROTATE.
ROTLSE	Real array, values as in ROTATE.
ROTST	Real array, values as in ROTATE.
HSTR	Real variable used as temporary storage in computing optical thickness.
RADFAC	Real variable in which is stored prefactor used in computing radiance values.
SUM	Real variable in which partial sums of relative radiances are accumulated.
TAU	Real variable in which optical thickness at line center is generated.
TAUFAC	Real variable in which prefactor for optical thickness is stored.
TEFAC	Real variable in which factor used in computing rotational populations is stored.
TQFAC	Real variable in which factor converting rotational partition function is stored.

```
      SUBROUTINE WATSTR (RAD,RM,T,DEN,N,LINES)                          10
      COMMON /ISPCOM/ ISP                                               20
      COMMON WVNUM(2500),ENLS(2500),STR(2500),HRAD(2500)                30
      TQFAC = (296.0/T)**1.5                                            40
      IF (ISP .EQ. 2)  TQFAC = 296.0/T                                  50
      GO TO (100,200,300), N                                            60
  100 CONTINUE                                                          70
      RADFAC = 1.58065E-20*DEN*TQFAC                                    80
      TAUFAC = 4.8815E-18*SQRT(RM/T)*TQFAC*DEN                          90
      DO 110 I = 1, LINES                                              100
      WHOLAM = 1.0E+4/WVNUM(I)                                         110
      TAU = TAUFAC*STR(I)*EXP(-1.43879*ENLS(I)/T)                      120
     1  *(1.0 - EXP(-14387.9/(WHOLAM*T)))                             130
      ROTEA = 2.8*STR(I)/WHOLAM**2                                     140
      HRAD(I) = RADFAC*ROTEA*SFUN(TAU)                                 150
     1  *EXP(-1.43879*(1.0E+4/WHOLAM + ENLS(I))/T)                     160
  110 CONTINUE                                                         170
      RETURN                                                           180
  200 CONTINUE                                                         190
      TAUFAC = 4.8815E-18*SQRT(RM/T)*DEN                               200
      TEFAC = (T - 296.0)/(T*296.0)*1.43879                            210
      SUM = 0.0                                                        220
      DO 210 I = 1, LINES                                              230
      HSTR = TQFAC*STR(I)*EXP(TEFAC*ENLS(I))                           240
      WHOLAM = 1.0E+4/WVNUM(I)                                         250
      TAU = TAUFAC*WHOLAM*HSTR                                         260
      HRAD(I) = TAU*SFUN(TAU)*EXP(-14387.9/(WHOLAM*T))                 270
     1  /WHOLAM**4                                                     280
      SUM = SUM + HRAD(I)                                              290
  210 CONTINUE                                                         300
      RADFAC = RAD/SUM                                                 310
      DO 220 I = 1, LINES                                              320
```

```
      HRAD(I) = HRAD(I)*RADFAC                                          330
220   CONTINUE                                                          340
      RETURN                                                            350
300   CONTINUE                                                          360
      RETURN                                                            370
      END                                                               380
```

Subroutine ROTSTR

Purpose: Subroutine ROTSTR computes the relative radiances of lines in the vibration - rotation bands of those linear molecules, which are not included in the line parameter compilation of McClatchey, et al. The radiances are corrected for effects of optical thickness, and absolute values then determined by normalizing to total band radiances computed by the BACKGROUND program. As for nonlinear molecules, a single temperature is assumed for the molecules in the line of sight.

Program Flow: Upon entry to ROTSTR, the reduced rotational temperature SIG is computed, relative strengths of the P, Q and R lines for J = 0 are assigned and values assigned to the increment of J values DJ and the variable J value AJ. Depending on the value of ND, relative strengths of rotational lines are computed in either the DO range ending with statement 110 (parallel transition) or the DO range ending with statement 120 (perpendicular transition). Control is transferred to statement 300 and the prefactor TAUFAC used in computing optical thickness is evaluated. The variable SUM which accumulates the sum of relative radiances is set to zero. In the DO range ending with statement 310, optical thicknesses for lines in the P, Q and R branches and relative radiances for these lines, corrected for the effects

of optical thickness, are computed. In the DO range ending with statement 320, absolute line radiances normalized to the total band radiance RAD are computed. Control is then returned to the main program.

Mnemonic Variables:

RAD	Real variable containing total band strength.
RM	Real variable containing molecular weight of molecule.
BVPP	Real variable containing mean rotational constant of lower vibrational state.
AJ	Real variable containing varying value of rotational quantum number J of lower vibrational state.
DJ	Real variable containing value of increment of rotational quantum number.
T	Real variable containing mean temperature of emitting layer.
DEN	Real variable containing number of molecules in lower state per square centimeter in line of sight.
STR	Real variable containing band strength at 300K.
PS	Real array in which radiances of lines in P branch are returned.
QS	Real array in which radiances of lines in Q branch are returned.
RS	Real array in which radiances of lines in R branch are returned.
NP	Integer variable determining whether band is parallel (NP = 1) or perpendicular)NP = 2).

ND	Integer variable determining whether all lines are present (ND = 1) or alternate lines are missing (ND = 2).
P	Real array containing wavelengths of lines in P branch.
Q	Real array containing wavelengths of lines in Q branch.
R	Real array containing wavelengths of lines in R branch.
FAC	Real variable used to store exponential factor in computation of line strengths.
PTAU	Real variable stcring optical thickness in a P branch line.
QTAU	Real variable storing optical thickness in a Q branch line.
RADFAC	Real variable storing ratio of total band radiance RAD and sum of relative line radiances.
RTAU	Real variable storing optical thickness of R branch line.
SIG	Real variable storing reduced rotational temperature.
SUM	Real variable which accumulates sum of relative line radiances.
TAUFAC	Real variable storing prefactor used in computing optical thickness at line centers.

```
      SUBROUTINE ROTSTR (RAD,RM,BVPP,T,DEN,STR,PS,QS,RS,NP,ND,P,Q,R)
      DIMENSION P(100),Q(100),R(100),PS(100),QS(100),RS(100)
C     ND .EQ. 1 FOR ALL LINES
C      ND .EQ. 2 FOR ALTERNATE LINES
C      NP .EQ. 1 FOR PARALLEL BAND
C      NP .EQ. 2 FOR PERPENDICULAR BAND
      SIG = 1.43879*BVPP/T
      PS(1) = 0.0
      QS(1) = 0.0
      RS(1) = 1.0
      SUM = 1.0
      AJ = ND
      DJ = ND
      IF (NP .GT. 1)  GO TO 200
     DO 110 I = 2, 100
      FAC = EXP(-SIG*AJ*(AJ + 1.0))
      PS(I) = AJ*FAC
      QS(I) = 0.0
      RS(I) = (AJ + 1.0)*FAC
      SUM = SUM + PS(I) + RS(I)
      AJ = AJ + DJ
  110 CONTINUE
      GO TO 300
  200 CONTINUE
     DO 210 I = 2, 100
      FAC = 0.5*EXP(-SIG*AJ*(AJ + 1.0))
      PS(I) = (AJ - 1.0)*FAC
      RS(I) = (AJ + 2.0)*FAC
      QS(I) = PS(I) + RS(I)
      SUM = SUM + 2.0*QS(I)
      AJ = AJ + DJ
  210 CONTINUE
```

```
300   CONTINUE                                                          330
      TAUFAC = 4.8815E-18*SQRT(RM/T)*DEN*STR/SUM                        340
      SUM = 0.0                                                         350
     DO 310 I = 1, 100                                                  360
      PTAU = PS(I)*P(I)*TAUFAC                                          370
      QTAU = QS(I)*Q(I)*TAUFAC                                          380
      RTAU = RS(I)*R(I)*TAUFAC                                          390
      PS(I) = PTAU*SFUN(PTAU)*EXP(-14387.9/(P(I)*T))/P(I)**4            400
      QS(I) = QTAU*SFUN(QTAU)*EXP(-14387.9/(Q(I)*T))/Q(I)**4            410
      RS(I) = RTAU*SFUN(RTAU)*EXP(-14387.9/(R(I)*T))/R(I)**4            420
      SUM = SUM + PS(I) + QS(I) + RS(I)                                 430
310   CONTINUE                                                          440
      RADFAC = RAD/SUM                                                  450
     DO 320 I = 1, 100                                                  460
      PS(I) = PS(I)*RADFAC                                              470
      QS(I) = QS(I)*RADFAC                                              480
      RS(I) = RS(I)*RADFAC                                              490
320   CONTINUE                                                          500
      RETURN                                                            510
      END                                                               520
```

Subroutine ROTPOS

Purpose: ROTPOS computes the wavelength of lines in the P, Q and R branches of the infrared vibrational - rotational bands of linear molecules neglecting spin and Λ or ℓ doubling. It is used for bands not included in the compilation of McClatchey, et al.

Method: The line positions are computed using equations (IV, 19), (IV, 20) and (IV, 22) of Herzberg (1945). It should be noted that use of these simple formulae results is an error in line positions which may approach 0.1μm for large J values in transitions between doublet states.

Program Flow: Upon entry to ROTPOS, the auxiliary values BDIF, BSUM, RCENT and TRI used in the compution are computed from the band center position and the rotational constants. The initial J value is set to zero and provisions made for omitting alternate lines by setting DJ equal to ND. This latter starting condition means that the alternate lines omitted are always those with odd J values. The wavelengths of 100 lines in each branch in units μm are then computed in the DO range ending with statement 110. The wavelengths are printed out and control returned to the main program.

Mnemonic Variables:

AJ	Real variable in which current rotational quantum number J is stored.
BDIF	Real variable storing difference of rotational constants BVP and BVPP.
BSUM	Real variable storing sum of rotational constants.
DJ	Real variable storing increment of rotational quantum number.
RCENT	Real variable containing constant used in computing R branch line positions.
TRI	Real variable containing constant used in computing R branch line positions.
CENTER	Real variable giving the wave number of the band center.
BVP	Real variable giving the mean rotational constant of the upper vibrational state.
BVPP	Real variable giving the mean rotational constant of the lower vibrational state.
P	Real array in which wavelengths of the P branch are returned.
Q	Real array in which wavelengths of the Q branch are returned.
R	Real array in which wavelengths of the R branch are returned.
ND	Integer variable determining whether all lines in a band are present (ND = 1) or whether alternate lines are missing (ND = 2).

```
      SUBROUTINE ROTPOS (CENTER,BVP,BVPP,P,Q,R,ND)                      10
C      ROTPOS COMPUTES ROTATIONAL LINE POSITIONS                        20
C      CENTER IS BAND CENTER, CM**(-1)                                  30
C      BVP IS ROTATIONAL CONSTANT OF UPPER STATE                        40
C      BVPP IS ROTATIONAL CONSTANT OF LOWER STATE                       50
C      P,Q,R ARE ARRAYS RETURNING LINE POSITIONS                        60
C      ND .EQ. 1 FOR ALL LINES                                          70
C      ND .EQ. 2 FOR ALTERNATE LINES                                    80
      DIMENSION P(100),Q(100),R(100)                                   90
      BDIF = BVP - BVPP                                               100
      BSUM = BVP + BVPP                                               110
      RCENT = CENTER + BVP + BVP                                      120
      TRI = 3.0*BVP - BVPP                                            130
      AJ = 1.0                                                        140
      DJ = ND                                                         150
      DO 110 I = 1, 100                                               160
      P(I) = 1.0E+4/((BDIF*AJ - BSUM)*AJ + CENTER)                    170
      Q(I) = 1.0E+4/((BDIF*AJ + BDIF)*AJ + CENTER)                    180
      R(I) = 1.0E+4/((BDIF*AJ + TRI)*AJ + RCENT)                      190
      AJ = AJ + DJ                                                    200
  110 CONTINUE                                                        210
      RETURN                                                          220
    8 FORMAT (10F10.3)                                                230
   15 FORMAT (1X)                                                     240
      END                                                             250
```

```
      FUNCTION SIMP (A,H,K)                                              10
      DIMENSION A(3)                                                     20
      IF (K - 2)  100,110,120                                            30
  100 CONTINUE                                                           40
      SIMP = (8.0*A(2) +5.0*A(1) - A(3))*H/12.0                          50
      RETURN                                                             60
  110 CONTINUE                                                           70
      SIMP = (8.0*A(2) + 5.0*A(3) - A(1))*H/12.0                         80
      RETURN                                                             90
  120 CONTINUE                                                          100
      SIMP = (4.0*A(2) + A(1) + A(3))*H/3.0                             110
      RETURN                                                            120
      END                                                               130
```

```
      FUNCTION SFUN (TAU)                                                    10
      T = ABS(TAU)                                                           20
      IF (T .GT. 4.0)  GO TO 110                                             30
      X = (T - 2.0)*0.5                                                      40
      SFUN = (((((0.001061*X - 0.004288)*X + 0.013586)*X - 0.042491)         50
     1  *X + 0.114932)*X - 0.267057)*X + 0.556485                           60
      RETURN                                                                 70
  110 CONTINUE                                                               80
      IF (T .GT. 10.0)  GO TO 120                                            90
      X = (T - 7.0)/3.0                                                     100
      SFUN = (((((0.000347*X - 0.001167)*X + 0.003149)*X - 0.009545)        110
     1  *X + 0.028248)*X - 0.081972)*X + 0.247900                          120
      RETURN                                                                130
  120 CONTINUE                                                              140
      Y = ALOG(T)                                                           150
      X = 1.0/Y                                                             160
      SFUN = (((((0.067876*X - 0.268698)*X + 0.264563)*X                    170
     1  - 0.259167)*X + 0.325368)*X + 1.128379)*SQRT(Y)/T                  180
      RETURN                                                                190
      END                                                                   200
```

```
      FUNCTION QINT (A,H1,H2,K)                                          10
      DIMENSION A(3)                                                     20
      HS = H1 + H2                                                       30
      IF (K - 2) 100,110,120                                             40
  100 CONTINUE                                                           50
      QINT = ((A(2)*HS - A(1)*H2 - A(3)*H1)*H1/(6.0*H2*HS)               60
     1  + (A(1) + A(2))*0.5)*H1                                          70
      RETURN                                                             80
  110 CONTINUE                                                           90
      QINT = ((A(2)*HS - A(1)*H2 - A(3)*H1)*H2/(6.0*H1*HS)              100
     1  + (A(2) + A(3))*0.5)*H2                                         110
      RETURN                                                            120
  120 CONTINUE                                                          130
      QINT = (A(2)*HS - A(1)*H2 - A(3)*H1)*(H1**3 + H2**3)              140
     1  /(6.0*HS*H1*H2) + (A(1)*H1 + A(2)*HS + A(3)*H2)*0.5             150
      RETURN                                                            160
      END                                                               170
```

PLOTTING PROGRAM

The plotting program, TRYCRT, which was developed under the previous contract, has been revised and adapted to use the pen and paper plotter in addition to the microform plotter. The version presented here is designed to read the output file produced by the spectral radiance program, SPCTRA. It consists of the main program, TRYCRT, and six subroutines, LINSCA, LOGSCA, LINAXS, LOGAXS, GRID and NULINE. In addition, it uses the AFCRL/CDC 6600 system subroutines which are needed to produce plots.

Array IDENT contains information required by the computer installation to identify the plotted graphs. The CDC 6600 subroutines are initialized by calls to subroutines PLTID3 and PLOT. The number of graphs to be plotted, NCASES, number of characters, NXL, in x-axis caption, x-axis caption, XBCD, number of characters, NYL, in y-axis caption, and y-axis caption, YBCD, are read in from cards and printed on the line printer. The DO loop ending with statement 400 is then executed for each graph to be produced. Minimum wavelength, WVMIN, and maximum wavelength, WVMAX, are read. Plotting constants INF, NX and NY are computed. Wavelengths are read from the SPCTRA output tape (channel 2) and stored in array AVLAM. An appropriate subset is then stored in array X. These are then scaled by subroutine LINSCA. Radiance data are then read from channel 2 and stored in array SPCTRA. An appropriate subset is selected and stored in array Y. Values in Y are scaled for plotting by a call to subroutine LOGSCA. Axes are drawn on all four sides of the graph by calls to subroutines LINAXS and LOGAXS. Calls to subroutine GRID place vertical and horizontal grid lines on the graph. Radiance values are further scaled in the DO loop ending

with statement 150. Radiance values are placed on the graph by calls to subroutine NULINE in the loop ending with statement 200. After each set of curves on a graph is produced, a call to subroutine PLOT prepares for a new graph. The DO loop ending with statement 250 is repeated three times, the first and second cycles plotting vertical and horizontal endoatmospheric radiances while the third plots limb viewing radiances. When only limb viewing radiances are desired, statement 250 should be placed immediately after the statement "READ (2) SPCTRA" on line TCRT 540.

Subroutines LINSCA and LOGSCA were written to provide automatic scaling of data points in a manner which would utilize maximum space on the graph. In addition to scaling data points, these subroutines generate values KONX, DX, MINY and MAXY which provide data which is used in the axis drawing subroutines LINAXS and LOGAXS and the grid drawing subroutine GRID. The subroutine NULINE was written to replace the standard AFCRL/CDC 6600 subroutine LINE which draws straight lines between plots. NULINE permits points to be plotted a limited distance outside the graph boundaries if the data for a graph covers more than 9 orders of magnitude.

The following listing includes both the original main program using microfilm plotter and a main program TRYPEN which uses the pen plotter.

Reproduced from best available copy.

```
      PROGRAM TPYCRT (INPUT,OUTPUT,TAPE2,TAPE5=INPUT,                   TCRT  10
     1  TAPE6=OUTPUT,TAPE30)                                            TCRT  20
C     VERSION 2                                                         TCRT  30
      COMMON AVLAY,SECTRA,Y,Y                                           TCRT  40
      DIMENSION IDENT(3)                                                TCRT  50
      DIMENSION AVLAY(201),SECTRA(201,12)                               TCRT  60
      DIMENSION Y(201),Y(2412)                                          TCRT  70
      DIMENSION XBCD(7),YBCD(7)                                         TCRT  80
      DIMENSION JBEAM(12)                                               TCRT  90
      DIMENSION TEMT(6)                                                 TCRT 100
      DATA IDENT(2),IDENT(3) /10H   BEGGES ,10H 7023  D-4/              TCRT 110
      DATA JBEAM /15,15,14,13,16,15,14,13,15,15,14,13/                  TCRT 120
      DATA TEMT /10H(/5X,5HSTA,10HET,6X,3HEN,10HD,7X,5HTOT,             TCRT 130
     1  10HAL,6X,5HTT,10HMES/1X,3F1.5H0.3/)/                            TCRT 140
    6 FORMAT (1X,I5)                                                    TCRT 150
    8 FORMAT (1X,I2,1X,3A10)                                            TCRT 160
   10 FORMAT (1X,E9.3,1X,E9.3)                                          TCRT 170
   16 FORMAT (1H1)                                                      TCRT 180
      CALL SECOND (SEC0)                                                TCRT 190
      CALL DATE (IDENT(1))                                              TCRT 200
      NW = 6                                                            TCRT 210
      REWIND 2                                                          TCRT 220
      WRITE (6,16)                                                      TCRT 230
      SX = 0.0                                                          TCRT 240
      SY = 14.0                                                         TCRT 250
      NW = 2                                                            TCRT 260
      CALL CRTPLT (IDENT,1.0,17.0)                                      TCRT 270
      CALL PLOT (1.0,1.0,-3)                                            TCRT 280
       READ (5,6) NCASES                                                TCRT 290
      WRITE (6,6) NCASES                                                TCRT 300
       READ (5,8) NXL,XBCD                                              TCRT 310
      WRITE (6,8) NXL,XBCD                                              TCRT 320
       READ (5,8) NYL,YBCD                                              TCRT 330
      WRITE (6,8) NYL,YBCD                                              TCRT 340
      DO 400 NCS = 1, NCASES                                            TCRT 350
       READ (5,10) WVMIN,WVMAX                                          TCRT 360
```

Reproduced from best available copy.

```
      WRITE (6,10) WWMIN,WWMAX                                          TOPT 370
      TNS = 10.0*WWMIN - 40.0                                           TOPT 380
      IF (TNS .LT. 1)  TNS = 1                                          TOPT 390
      TNE = 10.0*WWMAX - 40.0                                           TOPT 400
      IF (TNE .GT. 201)  TNE = 201                                      TOPT 410
      NX = TNE - TNS + 1                                                TOPT 420
      NY = 12*NX                                                        TOPT 430
      DO 300 NP = 1, 2                                                  TOPT 440
      READ (2) AWLAM                                                    TOPT 450
      DO 110 I = 1, NX                                                  TOPT 460
      X(I) = AWLAM(TNS + I - 1)                                         TOPT 470
  110 CONTINUE                                                          TOPT 480
      CALL LTNSCA (X,NX,SX,XMIN,DX,KONX)                                TOPT 490
      GAP = SX/FLOAT(KONX/10)                                           TOPT 500
      SX = GAP*DX                                                       TOPT 510
      NGAP = MOD(KONX,10)                                               TOPT 520
      DO 250 NSPC = 1, 2                                                TOPT 530
      READ (2) SPCTRA                                                   TOPT 540
      IJ = 1                                                            TOPT 550
      DO 130 J = 1, 12                                                  TOPT 560
      DO 120 I = 1, NX                                                  TOPT 570
      Y(IJ) = SPCTRA(TNS + I - 1,J)                                     TOPT 580
      IJ = IJ + 1                                                       TOPT 590
  120 CONTINUE                                                          TOPT 600
  130 CONTINUE                                                          TOPT 610
      CALL LOGSCA (Y,NY,SY,YMIN,YMAX,0)                                 TOPT 620
      CALL STREAM (16)                                                  TOPT 630
      CALL LINAXS (SY,0.0,0.0,YPOS,-NYL,SY,                             TOPT 640
     1  90.0,YMIN,SY,GAP,NW,NGAP)                                       TOPT 650
      CALL LINAXS (0.0,0.0,0.0,YPOS,NYL,SY,                             TOPT 660
     1  00.0,YMIN,SY,GAP,NW,-NGAP)                                      TOPT 670
      CALL LOGAXS (SY,0.0,0.0,YPOS,NYL,SY,                              TOPT 680
     1  180.0,YMIN,YMAX,1)                                              TOPT 690
      CALL LOGAXS (SY,SY,YPOS,-NYL,SY,                                  TOPT 700
     1  180.0,YMIN,YMAX,0)                                              TOPT 710
      CALL GRID (IFIX(SY/GAP - 0.9),GAP,SY)                             TOPT 720
```

```
      CALL GRID (MINY + 1 -MAXY,SY/FLOAT(MAXY - MINY),SX)                TCRT 730
      DO 150 I = 1, NY                                                   TCRT 740
      Y(I) = SY - Y(I)                                                   TCRT 750
  150 CONTINUE                                                           TCRT 760
      DO 200 J = 1, 12                                                   TCRT 770
      CALL STREAM (JSTRM(J))                                             TCRT 780
      IJ = (J - 1)*NX + 1                                                TCRT 790
      CALL MULTNE (X(IJ),Y,NX,SY,SX)                                     TCRT 800
  200 CONTINUE                                                           TCRT 810
      CALL FRAME (1.5,1.5)                                               TCRT 820
  250 CONTINUE                                                           TCRT 830
  300 CONTINUE                                                           TCRT 840
      REWIND 2                                                           TCRT 850
  400 CONTINUE                                                           TCRT 860
      CALL ENDPLT                                                        TCRT 870
      CALL SECOND (SECB)                                                 TCRT 880
      SECC = SECB - SECA                                                 TCRT 890
      WRITE (NW,TEMT) SECA,SECB,SECC                                     TCRT 900
      STOP                                                               TCRT 910
      END                                                                TCRT 920
```

```
      PROGRAM TRYPEN (TAPE2,TAPE5,OUTPUT,TAPE6=OUTPUT)                  10
C                                                                       20
      COMMON X,Y                                                        30
      DIMENSION IDENT(3)                                                40
      DIMENSION X(1801),Y(1801)                                         50
      DIMENSION XBCD(7),YBCD(7)                                         60
      DIMENSION TFMT(6)                                                 70
      DATA IDENT /10H   FOWLER ,10H   2105   ,10H  SPEC  A /            80
      DATA TFMT /10H(/5X,5HSTA,10HRT,6X,3HEN,10HD,7X,5HTOT,             90
     1  10HAL,6X,5HTI,10HMES/1X,3F1,5H0.3/)/                           100
    6 FORMAT (1X,I5)                                                   110
    8 FORMAT (1X,I2,1X,7A10)                                           120
   10 FORMAT (1X,F9.3,1X,F9.3)                                         130
   12 FORMAT (1P6E11.3)                                                140
   16 FORMAT (1H1)                                                     150
      CALL SECOND (SECA)                                               160
      NW = 6                                                           170
      REWIND 2                                                         180
      WRITE (6,16)                                                     190
      SY = 7.0                                                         200
      SX = 5.5                                                         210
      NX = 1801                                                        220
      NY = NX                                                          230
      NH = 2                                                           240
      CALL PLTID3 (IDENT,120.0,12.0,1.0)                               250
      CALL PLOT (1.0,1.0,-3)                                           260
       READ (5,8) NXL,XBCD                                             270
      WRITE (6,8) NXL,XBCD                                             280
       READ (5,8) NYL,YBCD                                             290
      WRITE (6,8) NYL,YBCD                                             300
     DO 250 NGRPH = 1, 2                                               310
      READ (2,12) Y                                                    320
```

160

```
      XXX=398.0                                                          330
     DO 110 I =1, 1801                                                   340
      XXX = XXX + 2.0                                                    350
      X(I) = XXX                                                         360
  110 CONTINUE                                                           370
      CALL LINSCA (X,NX,SX,XMIN,DX,KONX)                                 380
      GAP = SX/FLOAT(KONX/10)                                            390
      GX = GAP*DX                                                        400
      NGAP = MOD(KONX,10)                                                410
      CALL LOGSCA (Y,NY,SY,MINY,MAXY,9)                                  420
     DO 200 J = 1, 1                                                     430
      CALL LINAXS (0.0,0.0,XBCD,-NXL,SX,                                 440
     1  0.0,XMIN,GX,GAP,NH,NGAP)                                         450
      CALL LINAXS (SX,SY,XBCD,-NXL,SX,                                   460
     1  180.0,XMIN,GX,GAP,NH,-NGAP)                                      470
      CALL LOGAXS (0.0,0.0,YBCD,NYL,SY,                                  480
     1  90.0,MINY,MAXY,1)                                                490
      CALL LOGAXS (SX,0.0,YBCD,-NYL,SY,                                  500
     1  90.0,MINY,MAXY,0)                                                510
      CALL GRID (MAXY - MINY - 1,SY/FLOAT(MAXY - MINY),SX)               520
      CALL GRID (-IFIX(SX/GAP - 0.9),GAP,SY)                             530
      CALL NULINE (X,Y,NX,SX,SY)                                         540
      CALL PLOT (9.0,0.0,-3)                                             550
  200 CONTINUE                                                           560
      CALL LINAXS (0.0,0.0,XBCD,-NXL,SX,                                 570
     1  0.0,XMIN,GX,GAP,NH,NGAP)                                         580
      CALL LINAXS (SX,SY,XBCD,-NXL,SX,                                   590
     1  180.0,XMIN,GX,GAP,NH,-NGAP)                                      600
      CALL LOGAXS (0.0,0.0,YBCD,NYL,SY,                                  610
     1  90.0,MINY,MAXY,1)                                                620
      CALL LOGAXS (SX,0.0,YBCD,-NYL,SY,                                  630
     1  90.0,MINY,MAXY,0)                                                640
```

```
C       CALL GRID (MAXY - MINY - 1,SY/FLOAT(MAXY - MINY),SX)                650
C       CALL GRID (-IFIX(SX/GAP - 0.9),GAP,SY)                              660
        DO 240 J = 1, 1                                                     670
        CALL NULINE (X,Y,NX,SX,SY)                                          680
  240   CONTINUE                                                            690
        CALL PLOT (9.0,0.0,-3)                                              700
  250   CONTINUE                                                            710
  300   CONTINUE                                                            720
        REWIND 2                                                            730
        CALL SECOND (SECB)                                                  740
        SECC = SECB - SECA                                                  750
        WRITE (NW,TFMT) SECA,SECB,SECC                                      760
        CALL ENDPLT                                                         770
        STOP                                                                780
        END                                                                 790
```

```
      SUBROUTINE LINSCA (X,N,S,XMIN,DX,KON)                              10
C                                                                        20
C     THE SCALING ALGORITHM GUARANTEES THAT THE PLOT WILL USE AT         30
C     LEAST SEVENTY PER CENT OF THE LENGTH OF A LINEAR AXIS.             40
C                                                                        50
      DIMENSION A(14),KA(14),KB(14),X(100)                               60
      DATA A /1.0,1.2,1.5,1.8,2.0,2.4,3.0,4.0,5.0,6.0,7.5,               70
     1  9.0,10.0,12.0/                                                  80
      DATA KA / 10,12,15,9,20,12,15,10,10,12,15,9,10,12 /                90
      DATA KB / 44,33,34,32,44,33,34,44,44,33,34,32,44,33 /             100
      XMX = X(1)                                                         110
      XMN = XMX                                                          120
      ADX = 0.0                                                          130
      DO 110 I = 2, N                                                    140
      XMX = AMAX1(XMX,X(I))                                              150
      XMN = AMIN1(XMN,X(I))                                              160
  110 CONTINUE                                                           170
      SX = XMX - XMN                                                     180
      K = 0                                                              190
  120 CONTINUE                                                           200
      IF (SX .GE. 0.99999)  GO TO 130                                    210
      SX = 10.0*SX                                                       220
      K = K - 1                                                          230
      GO TO 120                                                          240
  130 CONTINUE                                                           250
      IF (SX .LT. 10.0)  GO TO 140                                       260
      SX = 0.1*SX                                                        270
      K = K + 1                                                          280
      GO TO 130                                                          290
  140 CONTINUE                                                           300
      DO 150 I = 1, 14                                                   310
      IF (SX .GT. 1.0005*A(I))  GO TO 150                                320
```

```
      ST = A(I)*10.0**K                                                  330
      DX = ST/FLOAT(KA(I))                                               340
      XMIN = XMN - AMOD(DX + AMOD(XMN,DX),DX)                            350
      DX = ST/S                                                          360
      KON = KB(I)                                                        370
      IF ((XMX - XMIN)/DX .LT. S*1.0005)  GO TO 160                      380
  150 CONTINUE                                                           390
  160 CONTINUE                                                           400
      DO 220 I = 1, N                                                    410
      X(I) = (X(I) - XMIN)/DX                                            420
  220 CONTINUE                                                           430
  300 CONTINUE                                                           440
      RETURN                                                             450
      END                                                                460
```

```
      SUBROUTINE LOGSCA (X,N,S,MIN,MAX,K)                                10
C     IF K IS POSITIVE, K LOG CYCLES WILL BE PRODUCED.                   20
C     IF K IS NEGATIVE, INPUT VALUE OF MIN WILL BE USED.                 30
      DIMENSION X(100)                                                   40
      MINSAV = MIN                                                       50
      XX = ALOG10(ABS(X(1)) + 1.0E-99)                                   60
      X(1) = XX                                                          70
      XMAX = XX                                                          80
      XMIN = XX                                                          90
      DO 110 I = 2, N                                                   100
      XX = ALOG10(ABS(X(I)) + 1.0E-99)                                  110
      X(I) = XX                                                         120
      XMAX = AMAX1(XMAX,XX)                                             130
      XMIN = AMIN1(XMIN,XX)                                             140
  110 CONTINUE                                                          150
      MAX = XMAX                                                        160
      MIN = XMIN                                                        170
      IF (FLOAT(MAX) .LT. XMAX) MAX = MAX + 1                           180
      IF (FLOAT(MIN) .GT. XMIN) MIN = MIN - 1                           190
      IF (K .LT. 0)  MIN = MINSAV                                       200
      IF (K .EQ. 0)  GO TO 120                                          210
      MIN = MAX0(MIN,MAX - K)                                           220
  120 CONTINUE                                                          230
      FAC = S/FLOAT(MAX - MIN)                                          240
      XMIN = MIN                                                        250
      XLAST = -0.5                                                      260
      DO 140 I = 1, N                                                   270
      X(I) = (X(I) - XMIN)*FAC                                          280
      IF (X(I) .LT. -0.01)  X(I) = -0.5                                 290
      IF (XLAST .GE. 0.0)  GO TO 130                                    300
      XLAST = X(I)                                                      310
      IF (I .EQ. 1)  GO TO 140                                          320
```

```
      IF (X(I) .GE. 0.0)  X(I-1) = -0.01                                330
      GO TO 140                                                         340
  130 CONTINUE                                                          350
      XLAST = X(I)                                                      360
      IF (X(I) .LT. 0.0)  X(I) = -0.01                                  370
  140 CONTINUE                                                          380
      RETURN                                                            390
      END                                                               400
```

```
      SUBROUTINE LINAXS (X,Y,BCD,NC,S,THETA,XMIN,DX,GAP,NH,NG)                  10
      DIMENSION BCD(8)                                                         20
      DATA W /10H X 10     /                                                   30
      NJ = IABS(NG)                                                            40
      NSN = 1                                                                  50
      IF (NC .LT. 0)  NSN = -1                                                 60
      TH = THETA*0.017453294                                                   70
      SN = NSN                                                                 80
      NC = NC*NSN                                                              90
      N = S/GAP + 0.5                                                         100
      S2 = 0.5*S                                                              110
      CTH = COS(TH)                                                           120
      STH = SIN(TH)                                                           130
      CFAC = CTH*GAP/FLOAT(NJ + 1)                                            140
      SFAC = STH*GAP/FLOAT(NJ + 1)                                            150
      NABS = IABS(NH)                                                         160
      ADJ = FLOAT(NABS)                                                       170
      IF (NABS .GT. 3)  ADJ = 5.0                                             180
      SADJ = ADJ*STH                                                          190
      CADJ = ADJ*CTH                                                          200
      XA = X + 0.0875*SN*SADJ                                                 210
      YA = Y - 0.0875*SN*CADJ                                                 220
      XD = X + 0.05*SN*SADJ                                                   230
      YD = Y - 0.05*SN*CADJ                                                   240
      HEIGHT = 0.05*ADJ                                                       250
      CALL PLOT (XA,YA,3)                                                     260
      XB = X                                                                  270
      YB = Y                                                                  280
      DO 120 I = 1, N                                                         290
      DO 110 J = 1, NJ                                                        300
      CALL PLOT (XB,YB,2)                                                     310
      XC = XB + CFAC                                                          320
```

```
      YC = YB + SFAC                                                    330
      CALL PLOT (XC,YC,2)                                               340
      XD = XD + CFAC                                                    350
      YD = YD + SFAC                                                    360
      CALL PLOT (XD,YD,2)                                               370
      XB = XC                                                           380
      YB = YC                                                           390
  110 CONTINUE                                                          400
      CALL PLOT (XB,YB,2)                                               410
      XC = XB + CFAC                                                    420
      YC = YB + SFAC                                                    430
      CALL PLOT (XC,YC,2)                                               440
      XA = XA + GAP*CTH                                                 450
      YA = YA + GAP*STH                                                 460
      CALL PLOT (XA,YA,2)                                               470
      XD = XD + CFAC                                                    480
      YD = YD + SFAC                                                    490
      XB = XC                                                           500
      YB = YC                                                           510
  120 CONTINUE                                                          520
      IF (NG .LE. 0)  RETURN                                            530
      ABSV = XMIN + DX*FLOAT(N)                                         540
      IF (NH .LT. 0)  GO TO 160                                         550
      ADX = ABS(DX)                                                     560
      E = 0.0                                                           570
  130 CONTINUE                                                          580
      IF (ADX .EQ. 0.0)  GO TO 150                                      590
      IF (ADX .LT. 1.0)  GO TO 140                                      600
      ADX = ADX/10.0                                                    610
      E = E - 1.0                                                       620
      ABSV = 0.1*ABSV                                                   630
      GO TO 130                                                         640
```

```
  140  CONTINUE                                                          650
       IF (ADX .GE. 0.1)  GO TO 150                                      660
       ABSV = ABSV*10.0                                                  670
       ADX = ADX*10.0                                                    680
       E = E + 1.0                                                       690
       GO TO 140                                                         700
  150  CONTINUE                                                          710
       IF (DX .GE. 0.0)  GO TO 160                                       720
       ADX = -ADX                                                        730
  160  CONTINUE                                                          740
       XA = XB - (0.1*SN - 0.025)*SADJ - 0.0964*CADJ                     750
       YA = YB + (0.1*SN - 0.025)*CADJ - 0.0964*SADJ                     760
  170  CONTINUE                                                          770
       N = N + 1                                                         780
       CFAC = 0.0429*CADJ                                                790
       SFAC = 0.0429*SADJ                                                800
       DO 200 I = 1, N                                                   810
       IF (NH .LT. 0)  GO TO 180                                         820
       CALL NUMBER (XA,YA,HEIGHT,ABSV,THETA,3)                           830
       ABSV = ABSV - ADX                                                 840
       GO TO 190                                                         850
  180  CONTINUE                                                          860
       IBD = 1                                                           870
       IF (ABS(ABSV) .GE. 10.0)  IBD =                                   880
      1  IFIX(ALOG10(ABS(ABSV)) + 1.0E-9) + 1                           890
       XB = XA - CFAC*FLOAT(IBD - 1)                                     900
       YB = YA - SFAC*FLOAT(IBD - 1)                                     910
       CALL NUMBER (XB,YB,HEIGHT,ABSV,THETA,3)                           920
       ABSV = ABSV - DX                                                  930
  190  CONTINUE                                                          940
       XA = XA - GAP*CTH                                                 950
       YA = YA - GAP*STH                                                 960
```

```
200  CONTINUE                                                          970
     IF (NH .GE. 0)  GO TO 210                                         980
     ANC = MC                                                          990
     XA = X + (S2 - 0.035*ANC*ADJ)*CTH - (0.18*SN - 0.035)*SADJ       1000
     YA = Y + (S2 - 0.035*ANC*ADJ)*STH + (0.18*SN - 0.035)*CADJ       1010
     HEIGHT = 0.07*ADJ                                                1020
     CALL SYMBOL (XA,YA,HEIGHT,BCD,THETA,MC)                          1030
     RETURN                                                           1040
210  CONTINUE                                                         1050
     TNC = MC + 7                                                     1060
     XA = X + (S2 - 0.035*TNC*ADJ)*CTH - (0.1875*SN - 0.0375)*SADJ    1070
     YA = Y + (S2 - 0.035*TNC*ADJ)*STH + (0.1875*SN - 0.0375)*CADJ    1080
     HEIGHT = 0.07*ADJ                                                1090
     CALL SYMBOL (XA,YA,HEIGHT,BCD,THETA,MC)                          1100
     IF (E .EQ. 0.0)  RETURN                                          1110
     XA = XA + ((TNC - 6.0)*0.07*ADJ)*CTH                             1120
     YA = YA + ((TNC - 6.0)*0.07*ADJ)*STH                             1130
     CALL SYMBOL (XA,YA,HEIGHT,W,THETA,5)                             1140
     IF (E - 1.0) 220,250,230                                         1150
220  CONTINUE                                                         1160
     XA = XA + 0.35*CADJ - 0.035*SADJ                                 1170
     YA = YA + 0.35*SADJ + 0.035*CADJ                                 1180
     GO TO 240                                                        1190
230  CONTINUE                                                         1200
     XA = XA + 0.35*CADJ - 0.035*SADJ                                 1210
     YA = YA + 0.35*SADJ + 0.035*CADJ                                 1220
240  CONTINUE                                                         1230
     HEIGHT = 0.05*ADJ                                                1240
     CALL NUMBER (XA,YA,HEIGHT,E,THETA,-1)                            1250
250  CONTINUE                                                         1260
     RETURN                                                           1270
     END                                                              1280
```

```
      SUBROUTINE LOGAXS (X,Y,BCD,NC,S,THETA,MIN,MAX,NTEST)                  10
      DIMENSION BCD(8)                                                      20
      SN = 1.0                                                              30
      IF (NC .LT. 0)  SN = -1.0                                             40
      FAC = 1.0                                                             50
      TH = 0.017453294*THETA                                                60
      CTH = COS(TH)                                                         70
      STH = SIN(TH)                                                         80
      CFAC = CTH*S/FLOAT(MAX - MIN)                                         90
      SFAC = STH*S/FLOAT(MAX - MIN)                                        100
      XL = 0.1*SN*STH*FAC                                                  110
      YL = 0.1*SN*CTH*FAC                                                  120
      XM = 1.75*XL                                                         130
      YM = 1.75*YL                                                         140
      CALL PLOT (X + XM,Y - YM,3)                                          150
      XA = X                                                               160
      YA = Y                                                               170
      N = 0                                                                180
  140 CONTINUE                                                             190
      P = 2.0                                                              200
      Q = 1.0                                                              210
      XB = XA                                                              220
      YB = YA                                                              230
  150 CONTINUE                                                             240
      CALL PLOT (XB,YB,2)                                                  250
      PQLOG = ALOG10(P/Q)                                                  260
      XB = XB + CFAC*PQLOG                                                 270
      YB = YB + SFAC*PQLOG                                                 280
      CALL PLOT (XB,YB,2)                                                  290
      IF (P .EQ. 10.0)  GO TO 160                                          300
      P = P + 1.0                                                          310
      Q = Q + 1.0                                                          320
```

```
      CALL PLOT (XB + XL,YB - YL,2)                                      330
      GO TO 150                                                          340
  160 CONTINUE                                                           350
      CALL PLOT (XB + XM,YB - YM,2)                                      360
      N = N + 1                                                          370
      XA = XA + CFAC                                                     380
      YA = YA + SFAC                                                     390
      IF (N - MAX + MIN .LT. 0)  GO TO 140                               400
      IF (NTEST .EQ. 0)  GO TO 300                                       410
      FAC = 0.7*FAC                                                      420
      XP = MIN                                                           430
      XA = X                                                             440
      YA = Y                                                             450
  170 CONTINUE                                                           460
      XB = XA - ((0.325*SN - 0.1)*STH + 0.235675*CTH)*FAC                470
      YB = YA + ((0.325*SN - 0.1)*CTH - 0.235675*STH)*FAC                480
      BASE = 10.0                                                        490
      IF (XP .NE. 0.0)  GO TO 180                                        500
      XB = XB + 0.064175*CTH*FAC                                         510
      YB = YB + 0.064175*STH*FAC                                         520
      BASE = 1.0                                                         530
  180 CONTINUE                                                           540
      CALL NUMBER (XB,YB,0.15*FAC,BASE,THETA,-1)                         550
      IF (XP .LT. 0.0)  GO TO 200                                        560
      IF (XP .EQ. 0.0)  GO TO 230                                        570
      IF (XP .EQ. 1.0)  GO TO 230                                        580
  190 CONTINUE                                                           590
      XT = XB + (0.3143*CTH - 0.1*STH)*FAC                               600
      YT = YB + (0.3143*STH + 0.1*CTH)*FAC                               610
      GO TO 220                                                          620
  200 CONTINUE                                                           630
  210 CONTINUE                                                           640
```

```
      XT = XB + (0.3713*CTH - 0.125*STH)*FAC                            650
      YT = YB + (0.3713*STH + 0.125*CTH)*FAC                            660
  220 CONTINUE                                                          670
      CALL NUMBER (XT,YT,0.1*FAC,XP,THETA,-1)                           680
  230 CONTINUE                                                          690
      XA = XA + CFAC                                                    700
      YA = YA + SFAC                                                    710
      XP = XP + 1.0                                                     720
      IF (XP - 1.0 - FLOAT(MAX) .LT. 0.0)  GO TO 170                    730
      OC = FLOAT(NC)*SN                                                 740
      T = S/FAC                                                         750
      XB = X + ((T - OC*0.20 - 0.05725)*0.5*CTH -                       760
     1  (0.6*SN - 0.1)*STH)*FAC                                         770
      YB = Y + ((T - OC*0.20 - 0.05725)*0.5*STH +                       780
     1  (0.6*SN - 0.1)*CTH)*FAC                                         790
      ND = OC                                                           800
      CALL SYMBOL (XB,YB,0.2*FAC,BCD,THETA,ND)                          810
  300 CONTINUE                                                          820
      RETURN                                                            830
      END                                                               840
```

```
      SUBROUTINE GRID (NL,SPACE,S)                                      10
      NI = IABS(NL)                                                     20
      AX = 0.0                                                          30
      AY = 0.0                                                          40
      IF (SPACE .EQ. 0.0)  GO TO 200                                    50
      IF (NL .EQ. 0)  GO TO 200                                         60
      IF (NL .GT. 0)  GO TO 120                                         70
      BX = 0.0                                                          80
      BY = S                                                            90
      DX = ABS(SPACE)                                                  100
      DY = 0.0                                                         110
      GO TO 130                                                        120
  120 CONTINUE                                                         130
      BX = S                                                           140
      BY = 0.0                                                         150
      DX = 0.0                                                         160
      DY = ABS(SPACE)                                                  170
  130 CONTINUE                                                         180
      IT = 0                                                           190
     DO 150 I = 1, NI                                                  200
      AX = AX + DX                                                     210
      AY = AY + DY                                                     220
      BX = BX + DX                                                     230
      BY = BY + DY                                                     240
      IT = 1 - IT                                                      250
      IF (IT .LT. 0)  GO TO 200                                        260
      IF (IT .GT. 0)  GO TO 140                                        270
      CALL PLOT (BX,BY,3)                                              280
      CALL PLOT (AX,AY,2)                                              290
      GO TO 150                                                        300
  140 CONTINUE                                                         310
      CALL PLOT (AX,AY,3)                                              320
```

```
      CALL PLOT (BX,BY,2)                                              330
150   CONTINUE                                                         340
200   CONTINUE                                                         350
      RETURN                                                           360
      END                                                              370
```

```
      SUBROUTINE NULINE (X,Y,N,SX,SY)                                    10
      DIMENSION X(100),Y(100)                                            20
      DL = 0.1                                                           30
      SXA = SX + DL                                                      40
      SYA = SY + DL                                                      50
      XA = X(1)                                                          60
      YA = Y(1)                                                          70
      IPNA = 2                                                           80
      IF (XA .LT. -DL)  IPNA = 3                                         90
      IF (XA .GT. SXA)  IPNA = 3                                        100
      IF (YA .LT. -DL)  IPNA = 3                                        110
      IF (YA .GT. SYA)  IPNA = 3                                        120
      CALL PLOT (XA,YA,3)                                               130
      DO 150 I = 2, N                                                   140
      IPNB = IPNA                                                       150
      XA = X(I)                                                         160
      YA = Y(I)                                                         170
      IPNA = 2                                                          180
      IF (XA .LT. -DL)  IPNA = 3                                        190
      IF (XA .GT. SXA)  IPNA = 3                                        200
      IF (YA .LT. -DL)  IPNA = 3                                        210
      IF (YA .GT. SYA)  IPNA = 3                                        220
      IPN = 2                                                           230
      IF (IPNA + IPNB .NE. 4)  IPN = 3                                  240
      CALL PLOT (XA,YA,IPN)                                             250
  150 CONTINUE                                                          260
      RETURN                                                            270
      END                                                               280
```

Program BCKGND (BGND)

Purpose: The main program BCKGND (short for BACKGROUND) serves primarily to control the operation of subroutines which compute vibrational populations and radiative rates. BCKGND reads and checks the input data, and transfers control to Subroutine BANRAD which computes vibrational populations, local optical thicknesses and volume radiances for all included bands of a single molecular species. When control is returned, BCKGND computes and outputs integrated band radiances for limb viewing (exoatmospheric) and seven endoatmospheric viewing angles at selected tangent heights or endoatmospheric altitudes.

Program Flow: The initial step in operation of BCKGND is the reading of all input data. The number IGAS of molecular species including total number density is read in. Then in the DO range ending with statement 170, the following operations are performed. The title card, containing alphanumeric identification, an index check number and maximum altitude considered for the species is read in and its check number checked against the index of the DO range. If the check succeeds, altitude and number densities are read into arrays ALT and STORE until an altitude corresponding to the maximum altitude is found. When this altitude is found, the logarithm of input number densities is formed in the DO range ending with statement 130 and control is passed to Subroutine INTERP which interpolates to form values at 1 km intervals up to 160 km and at 5 rm intervals at higher altitudes. The interpolated values are converted

to number densities in the DO range ending with statement 160. In the DO range ending with statement 175, total atmosphere number densities are stored in array TOTDEN. Temperature data are then read in similarly in the DO range ending with statement 245 and interpolated values found by Subroutine INTERP. The complete set of number densities and temperatures is then pointed out. Next, vibrational level and band constants are read in for each species in the DO range ending with statement 331. The first card for each species contains an alphanumeric identification, number of vibrational levels NLEV, number of bands NBAN, molecular mass AMAS, and two vibrational energy transfer coefficients DEXT and DEXB. The alphabetic code is checked and if it is a valid code stored in array DATATY, these constants are transferred to the appropriate arrays. In the DO range ending with statement 313, constants for each vibrational level are read in, and card sequence numbers checked. In the DO range ending with statement 316, constants for each band are read in and card sequence numbers checked. For CO_2 and H_2O, tabular values of temperatures and associated vibrational energy exchange coefficients are read before the vibrational level and molecular band data.

The computation of molecular band radiances then proceeds in the following manner. The outer DO range index NLTE and ending with statement 900 controls the factors considered in a complete set of calculations for any of five possible sets. The corresponding DO statement may be varied depending on what factors are desired. See the description of Subroutine BANRAD for the effect

of different values of the index NLTE. Each cycle through the nested DO range with Index I and ending with statement 150 results in the computation of band radiance values for a single molecule. Action within this DO range is as follows.

Subroutine BANRAD is called and returns volume emission radiance values in array RAD and local optical thickness values in array TAU. Then for each band, line of sight band radiances are computed in the DO range ending with statement 840. Within this DO range, the value of NLTE is examined and appropriate headings are printed. Then trapezoidal rule integration is performed to determine the radiance arising along each of the eight lines of sight, including the modification necessary for the effects of optical thickness along the line of sight. The DO range ending with statement 830 controls the integration for each tangent height and endoatmospheric viewing level. In the current version, radiance values are computed at 5 km intervals of tangent height up to 150 km and at 25 km intervals above this level. The integration of radiance values is performed for each endoatmospheric viewing level in the DO range ending with statement 810. The integration begins at the viewing altitude. At each point on the line of sight, the contribution of the local volume emission rate to that received at the viewing point is computed by Subroutine SMI, which takes into account the effects of optical thickness between the emitting point and the viewing point. The equivalent integration for the limb viewing case is

performed in the DO ranges ending with statements 815 and 818. Upon completion of the integration for each tangent height and endoatmospheric viewing level, band radiances for that level are punched and printed. This punched output, together with that produced by Subroutine BANRAD, is used as input data to the special radiance program SPECTRA.

Tables V-2 through V-4 list the molecular bands in the program. These are followed by a sample set of input data and the program listing.

Table V-2

Water Vapor Band Data

Vibrational Transition		Band Center	Band Strength	Solar Flux
Upper State	Lower State	Wavelength (μm)	at 296 K (mol^{-1} cm^2 cm^{-1})	Band Center (photons cm^{-2} μm^{-1})
010	000	6.27	1.06(-17)*	5.60(15)
020	000	3.17	6.58(-20)	3.50(16)
100	000	2.73	3.62(-19)	5.10(16)
001	000	2.66	7.99(-18)	5.40(16)
020	010	6.42	8.61(-21)	5.70(15)
100	010	4.85	8.90(-23)	1.20(16)
001	101	4.63	4.90(-22)	1.50(16)
030	010	3.26	7.99(-23)	3.30(16)
110	010	2.75	1.50(-22)	5.00(16)
011	010	2.68	3.30(-21)	5.40(16)
030	020	6.60	7.00(-24)	4.90(15)
030	000	2.14	2.00(-22)	9.30(16)
110	000	1.91	1.83(-20)	1.20(17)
011	000	1.88	9.16(-19)	1.25(17)

* Number in parentheses is power of ten.

Table V-3

Carbon Dioxide Band Data

Vibrational Transition		Band Center	Band Strength at 296 K	Solar Flux Band Center
Upper State	Lower State	Wavelength (μm)	(mol^{-1} cm^2 cm^{-1})	(photons cm^{-2} μm^{-1})
01101	00001	14.98	8.26 (-18)	4.60 (14)
10002	01101	16.18	1.44 (-19)	3.70 (14)
02201	01101	14.98	6.49 (-19)	4.60 (14)
11102	10002	15.45	2.22 (-20)	4.20 (14)
1102	02201	16.74	5.21 (-21)	3.35 (14)
03301	02201	14.97	3.82 (-20)	4.60 (14)
10001	01101	13.87	1.85 (-19)	5.80 (14)
00011	10001	10.41	4.91 (-22)	1.31 (15)
00011	00001	4.62	9.60 (-17)	1.60 (16)

Table V-4

Ozone Band Data

Vibrational Transition		Band Center		Solar Flux
Upper State	Lower State	Wavelength (μm)	Band Strength, 296 K (mol^{-1} cm^2 cm^{-1})	(photons cm^{-2} μm^{-1})
010	000	14.27	4.13(-19)*	5.30(14)
001	000	9.60	1.29(-17)	1.70(15)
100	000	9.06	3.47(-19)	1.90(15)
101	000	4.74	1.33(-18)	1.20(16)
111	000	3.59	2.32(-20)	2.50(16)
003	000	3.29	1.10(-19)	3.20(16)
101	100	9.92	9.49(-21)	1.50(15)
011	010	9.79	4.23(-19)	1.60(15)
002	001	9.74	1.62(-19)	1.60(15)
003	002	10.29	1.66(-21)	1.40(14)
011	000	5.81	3.10(-20)	6.90(15)

* Number in parentheses is power of ten.

```
7     END
METHANE - CH4                1   120.
 60. 1.0E+10    70. 2.4E+09    80. 5.0E+08    90. 1.0E+08   100. 1.0E+07   1  1
110. 1.7E+05   120. 1.0E+04     .  . E+       .  . E+       .  . E+       1  2
CARBON DIOXIDE - CO2         2   200.
 60. 2.0E+12    65. 1.1E+12    70. 5.8E+11    75. 2.9E+11    80. 1.3E+11   2  1
 85. 5.3E+10    90. 2.1E+10    95. 8.1E+09   100. 3.3E+09   105. 1.4E+09   2  2
110. 6.6E+08   115. 3.1E+08   120. 1.7E+08   130. 5.3E+07   140. 2.4E+07   2  3
150. 1.3E+07   160. 8.4E+06   170. 5.9E+06   180. 4.3E+06   200. 2.5E+06   2  4
WATER VAPOR - H2O            3   200.
 60. 3.0E+10    65. 1.5E+10    70. 4.5E+09    75. 1.2E+08    80. 4.5E+07   3  1
 85. 4.2E+06    90. 1.9E+06    95. 9.6E+05   100. 2.9E+05   110. 4.8E+04   3  2
120. 8.0E+03   200. 5.0E+02     .  . E+       .  . E+       .  . E+       3  3
NITRIC OXIDE - NO            4   500.
 60. 1.6E+09    65. 1.6E+09    70. 1.3E+09    80. 2.2E+08    95. 7.5E+07   4  1
110. 2.3E+07   125. 1.3E+07   150. 7.4E+06   200. 1.9E+06   500. 1.1E+02   4  2
NITROUS OXIDE - N2O          5   150.
 60. 3.6E+09    70. 9.5E+08    80. 2.5E+08    90. 5.0E+07   100. 1.3E+07   5  1
110. 3.9E+06   120. 8.2E+05   130. 2.2E+05   140. 5.6E+04   150. 1.4E+04   5  2
OZONE - O3  PRE-DAWN         6   150.
 60. 1.0E+10    65. 2.0E+10    70. 3.0E+10    75. 2.0E+10    80. 1.0E+10   6  1
 85. 1.1E+07    86. 1.0E+07    87. 1.1E+07    88. 1.4E+07    90. 2.5E+07   6  2
 93. 3.6E+07    96. 3.6E+07   100. 1.9E+07   110. 1.6E+06   120. 8.8E+04   6  3
150. 1.3E+02     .  . E+       .  . E+       .  . E+       .  . E+       6  4
OZONE - O3 NOON              7   130.
 60. 1.0E+10    65. 4.8E+09    70. 2.0E+09    75. 7.6E+08    80. 1.1E+08   7  1
 85. 1.1E+07    86. 8.3E+06    87. 7.9E+06    90. 1.5E+07    92. 1.8E+07   7  2
 94. 1.7E+07    97. 1.0E+07   105. 1.8E+06   112. 1.9E+05   130. 1.3E+02   7  3
MOL NO DEN - ICAO ST ATM     8   500.
 60. 6.4E+15    65. 3.5E+15    70. 1.8E+15    75. 9.0E+14    80. 4.2E+14   8  1
 90. 6.5E+13    95. 2.5E+13   100. 1.0E+13   110. 2.1E+12   120. 5.2E+11   8  2
130. 1.7E+11   140. 7.5E+10   150. 4.1E+10   200. 7.8E+09   250. 2.5E+09   8  3
300. 9.5E+08   350. 4.2E+08   400. 2.0E+08   450. 1.0E+08   500. 5.3E+07   8  4
```

```
TEMP - ICAO STD. ATMOS.  1   500.
  60.   254.65    63.   244.65    80.   180.65    90.   180.65   100.   210.02   1  1
 110.   257.00   120.   343.49   130.   533.80   140.   714.22   150.   892.79   1  2
 160.  1022.23   170.  1105.51   180.  1156.12   190.  1205.50   200.  1235.35   1  3
 210.  1265.49   220.  1294.09   230.  1321.70   240.  1339.90   250.  1357.28   1  4
 260.  1373.84   270.  1389.59   280.  1404.54   290.  1418.70   300.  1432.11   1  5
 340.  1458.23   380.  1473.23   420.  1487.85   460.  1499.73   500.  1499.22   1  6
METHANE - CH4                    04    4     16.00       1.0000E-15      2.0000E-15
    11          0.0000E+00  3.4000E-01                                          1  1
    31          1.3062E+03  3.5000E-01                                          1  2
    31          3.0184E+03  3.5000E-01                                          1  3
    21     2    1.5260E+03  3.5000E-01                                          1  4
   201    11    1.8500E+02  3.1500E+15  1.6000E+16                              1  1
   301    11    3.2000E+02  3.1000E+16  4.0000E+13                              1  2
   401    11    2.4000E+00  4.9000E+15  3.5000E+15                              1  3
   302    11    1.0000E-02  6.8000E+15  1.8000E+15                              1  4
CARBON DIOXIDE - CO2             08   00    44.0
  .08  -12.70   .10  -13.58   .12  -14.21   .14  -14.74   .16  -15.16   .18  -15.40
    0.  -12.07 400.  -12.32 800.  -12.471200.  -12.471600.  -12.222000.  -11.92
  .09  -12.86   .11  -13.82   .13  -14.44   .15  -14.80   .17  -15.05   .19  -15.21
    12          0.0000E+00  3.9321E-01                                          2  1
    21          6.6736E+02  3.9094E-01                                          2  2
    12          1.2854E+03  3.9048E-01                                          2  3
    22          1.3351E+03  3.9166E-01                                          2  4
    21          1.9325E+03  3.9122E-01                                          2  5
    21          2.0033E+03  3.9236E-01                                          2  6
    12          1.3882E+03  3.9120E-01                                          2  7
    11          2.3491E+03  3.9020E-01                                          2  8
   201    12     8.258E-18  4.6000E+14                                          2  1
   302    12     1.436E-19  3.7000E+14                                          2  2
   402    12     6.468E-19  4.6000E+14                                          2  3
   503    12     2.220E-20  4.2000E+14                                          2  4
   504    12     5.208E-21  3.3500E+14                                          2  5
```

604	12	3.824E-20	4.6[illegible]E+14														2	6	
702	12	1.853E-19	5.8000E+14														2	7	
807	11	4.910E-22	1.3100E+15														2	8	
801	11	9.594E-17	1.6000E+16														2	9	
WATER VAPOR - 420																			
				9	14	18.00		5.0000E-11		2.0000E-15									
0.09	-12.31	0.11	-13.14	0.13	-13.99	0.15	-14.62	0.17	-15.29	0.19	-15.95								
11		0.0000E+00	3.0000E-02																
11		1.5967E+03	3.0000E-02														3	1	
11		3.1516E+03	3.0000E-02														3	2	
11		3.6571E+03	3.0000E-02														3	3	
11		3.7559E+03	3.0000E-02														3	4	
11		4.3667E+03	3.0000E-02														3	5	
11		5.2356E+03	3.0000E-02														3	6	
11		5.3331E+03	3.0000E-02														3	7	
201	11	1.0619E-17	5.6000E+15														3	8	
301	11	6.5810E-20	3.5000E+16														3	1	
401	11	3.6200E-19	5.1000E+16														3	2	
501	11	7.9940E-18	5.4000E+16														3	3	
302	11	9.5100E-21	5.2000E+15														3	4	
402	11	8.9000E-23	1.2000E+16														3	5	
502	11	4.9000E-22	1.5000E+16														3	6	
602	11	7.990[illegible]E-23	3.3400E+16														3	7	
702	11	1.5000E-22	5.0000E+16														3	8	
802	11	3.3600E-21	5.4000E+16														3	9	
603	11	7.0000E-24	4.9000E+15														3	10	
601	11	2.0600E-22	9.2000E+16														3	11	
701	11	1.8300E-20	1.2000E+17														3	12	
801	11	3.0600E-19	1.2500E+17														3	13	
NITRIC OXIDE - NO																	3	14	
				3	7	30.00		4.0000E-15		2.0000E-15									
11		0.0000E+00	1.6951E+06														4	1	
11	1	1.8780E+03	1.5732E+10														4	2	

11		3.7270E+03	1.6505E+06				4	3
201	11	1.2500E+02	8.7000E+15	3.3000E+15			4	1
301	11	2.1100E+00	5.4300E+16	1.5000E+13			4	2
302	11	3.1500E-02	8.4000E+15	3.0000E+15			4	3
NITROUS OXIDE - N2O			7 3	44.00	1.0000E-15	2.5000E-13		
11		0.0000E+00	4.1001E-01				5	1
21		5.8880E+02	4.0300E-01				5	2
31		1.1730E+03	4.0000E-01				5	3
41		1.7580E+03	4.0000E-01				5	4
51		2.3385E+03	4.0000E-01				5	5
11	1	2.2235E+03	4.0000E-01				5	6
11		1.2850E+03	4.0000E-01				5	7
201	12	3.3000E+01	3.4300E+14	2.5000E+16			5	1
701	11	2.6000E+02	3.0000E+15	1.8000E+16			5	2
601	11	1.8560E+03	1.6000E+16	1.3000E+14			5	3
OZONE PRE-DAWN - O3			9 11	48.00	1.0000E-14	1.0000E-14		
11		0.0000E+00	5.0000E-02				6	1
11		7.0098E+02	5.0000E-02				6	2
11		1.6421E+03	5.0000E-02				6	3
11		1.1032E+03	5.0000E-02				6	4
11		1.7220E+03	5.0000E-02				6	5
11		2.0692E+03	5.0000E-02				6	6
11		2.1104E+03	5.0000E-02				6	7
11		2.7852E+03	5.0000E-02				6	8
11		3.0412E+03	5.0000E-02				6	9
201	11	4.1300E-19	5.3000E+14				6	1
301	11	1.2900E-17	1.7100E+15				6	2
401	11	3.4700E-19	1.9000E+15				6	3
701	11	1.3300E-18	1.2000E+16				6	4
801	11	2.3200E-20	2.5000E+16				6	5
901	11	1.1000E-16	3.2300E+15				6	6
704	11	9.4900E-21	1.5000E+15				6	7
502	11	4.2300E-19	1.6000E+15				6	8
603	11	1.6200E-19	1.6000E+15				6	9
905	11	1.6580E-21	1.4300E+14				6	10
501	11	3.1000E-20	6.0000E+15				6	11
OZONE NOON - O3- O3			2 1	48.00	1.0000E-14	1.0000E-14		
11		0.0000E+00	3.0000E-01				7	1
11	2	1.0430E+03	3.0000E-01				7	2
201	11	3.6000E+02	1.6800E+15	2.3000E+16			7	1

3.8[illegible]E+01	4.[illegible]673E+01	4.[illegible]335E+01	4.6598E+01	4.9976E+01	5.3385E+01
5.7141E+01	6.1161E+01	6.5+64E+01	7.0070E+01	7.5000E+01	8.1862E+01
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
1.5130E+02	1.6491E+02	1.[illegible]00E+02	1.9036E+02	2.2081E+02	2.4456E+02
2.7[illegible]	3.[illegible]	3.[illegible]27E+02	3.6901E+02	4.[illegible]760E+02	4.5144E+02
5.0000E+02	5.6833E+02	6.4[illegible]0E+02	7.3428E+02	8.3462E+02	9.4868E+02
[illegible]	1.2257E+03	1.3[illegible]32E+03	1.5[illegible]E+03	1.8000E+03	2.0896E+03
2.4257E+03	2.8159E+03	3.2[illegible]E+03	3.7947E+03	4.4052E+03	5.1139E+03
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
1.2415E+04	1.3956E+04	1.5+65E+04	1.7261E+04	1.9266E+04	2.1503E+04
[illegible]	[illegible]	[illegible]	2.8387E+04	3.0021E+04	3.1749E+04
3.3576E+04	3.5[illegible]E+04	3.7[illegible]E+04	3.9714E+04	4.2000E+04	4.2291E+04
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
4.4383E+04	4.4691E+04	4.5[illegible]E+04	4.5099E+04	4.5198E+04	4.5298E+04
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
4.6[illegible]E+04	4.6[illegible]E+04	4.8[illegible]E+04	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
7.6[illegible]E+03	4.9[illegible]E+03	3.2[illegible]E+03	2.3933E+03	1.7899E+03	1.3387E+03
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
2.0427E+02	1.5[illegible]E+02	1.2337E+02	9.5875E+01	7.4509E+01	5.7904E+01
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
9.0205E+00	6.9[illegible]E+00	5.27[illegible]E+00	4.0387E+00	3.0897E+00	2.3636E+00
[illegible]	[illegible]	[illegible]	8.0958E-01	6.1934E-01	[illegible]
3.6247E-01	2.7720E-01	2.1213E-01	1.6228E-01	1.2415E-01	9.4976E-02
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
1.4560E-02	1.1142E-02	8.52+0E-03	6.5210E-03	4.9885E-03	3.8164E-03
2.9196E-03	2.2336E-03	1.7087E-03	1.3072E-03	1.[illegible]06E-03	

```
      PROGRAM BGND (INPUT,OUTPUT,PUNCH,TAPE1,TAPE2,TAPE3,                10
     1  TAPE4=PUNCH,TAPE5=INPUT,TAPE6=OUTPUT)                            20
      DIMENSION XID(25,4),XIDT(4),ALT(30),STORE(30),IALT(25)             30
      DIMENSION DATATY(25),CDEXT(25),CDEXB(25),NLEVEL(25),NBAND(25)      40
      DIMENSION LSC(25,25),LCC(25,25),RCM(25,25),BV(25,25),AMASS(25)     50
      DIMENSION ZANG(7),STAU(7),SRAD(8),RDN(8),RATIO(8)                  60
      DIMENSION LBC(25,25),LBU(25,25),STR(25,25),SFLUX(25,25),TFLUX(25,  70
     125,3)                                                              80
      DIMENSION CO2(18,2),WATVA(18,2)                                    90
      REAL LAMDA                                                        100
      COMMON /VBRCOM/ I,CO2,WATVA,CDEXT,CDEXB                           110
      COMMON /TEQCOM/ NLTE                                              120
      COMMON/KCEM/KCHEM                                                 130
      COMMON/INOUT/NR,NP,NW                                             140
      COMMON CONC(201,10),TAU(201,25),RAD(201,25),T(201),LAMDA(25),     150
     1TOTDEN(201)                                                       160
      DATA(ZANG(I),I=1,7)/0.0,20.0,40.0,60.0,70.0,80.0),90.0/           170
      DATA(DATATY(I),I=1,25)/8HMETHANE ,8HCARBON D,8HWATER VA,8HNITRIC O 180
     1,8HNITROUS ,8HOZONE PR,8HOZONE NO,18*8H         /                 190
      KCHEM = 1                                                         200
      NR = 5                                                            210
      NP = 3                                                            220
       NTX = 6                                                          230
      NW = 6                                                            240
      REWIND 1                                                          250
      REWIND 2                                                          260
   10 FORMAT (I2)                                                       270
      READ(NR,10) IGAS                                                  280
   20 FORMAT(1H1,22HNUMBER OF GAS PROFILES, I3)                         290
      WRITE(NW,20) IGAS                                                 300
      IGAS=IGAS+1                                                       310
      DO 170 J=1,IGAS                                                   320
```

```
 30 FORMAT (4A6,I3,F7.0)                                                  330
    READ(NR,30)(XID(J,K), K=1,4), ICHK,ALTM                               340
    IF(J.EQ.ICHK) GO TO 50                                                350
 40 FORMAT ( 7HID CARD, A6, 20HOUT OF ORDER FOR GAS, I3)                  360
    WRITE(NW,40)(XID(J,K),K=1,4), J                                       370
 50 DO 105 K=1,6                                                          380
    K1=(K-1)*5+1                                                          390
    K2=K1+4                                                               400
 60 FORMAT (5(F5.0,E8.1,2X),I2,I3)                                        410
    READ(NR,60)(ALT(L),STORE(L),L=K1,K2), ICHK, KCHK                      420
    IF(ICHK.EQ.J) GO TO 80                                                430
 70 FORMAT(30HDATA CARD OUT OF ORDER FOR GAS,I3, 2X, 8HCARD NO., I3)      440
    IFIVE = 5                                                             450
    WRITE (NW,70) J,IFIVE                                                 460
 80 IF(KCHK.EQ.K) GO TO 90                                                470
    WRITE(NW,70) J,K                                                      480
 90 DO 100 L=K1,K2                                                        490
    IF(ALT(L).LT.ALTM) GO TO 99                                           500
    KSET=L                                                                510
    GO TO 120                                                             520
 99 CONTINUE                                                              530
100 CONTINUE                                                              540
105 CONTINUE                                                              550
110 FORMAT(42HMAX ALT. NOT FOUND ON DATA CARDS - GAS NO., I3)             560
    WRITE(NW,110) J                                                       570
120 DO 130 L=1,KSET                                                       580
130 STORE(L)=ALOG(STORE(L))                                               590
    IF(ALTM.GT.150.) GO  TO 140                                           600
    IALT(J)=ALTM-59.                                                      610
    GO TO 150                                                             620
140 IALT(J)=91.+(ALTM-150.)/5.                                            630
150 CALL INTERP(ALT,STORE,KSET,1,J)                                       640
```

```
      L1=IALT(J)                                                        650
      DO 160 L=1,L1                                                     660
  160 CONC(L,J)=EXP(CONC(L,J))                                          670
  170 CONTINUE                                                          680
      DO 175 J=1,201                                                    690
  175 TOTOEN(J)=CONC(J,IGAS)                                            700
      READ(NP,30) (XIDT(K),K = 1,4), ICHK, KCHK                         710
      IF(ICHK.EQ.1)  GO TO 190                                          720
  180 FORMAT( 7HID CARD, 4A6,26HOUT OF ORDER FOR TEMP DATA)             730
      WRITE(NW,180) (XIDT(K),K=1,4)                                     740
  190 DO 245 K=1,6                                                      750
      K1=(K-1)*5+1                                                      760
      K2=K1+4                                                           770
      READ(NP,60)(ALT(L),STORE(L),L=K1,K2), ICHK, KCHK                  780
      IF(ICHK.EQ.1) GO TO 210                                           790
  200 FORMAT(31HTEMP DATA CARD OUT OF ORDER NO.,I3)                     800
      WRITE(NW,200)  K                                                  810
  210 IF(KCHK.EQ.K) GO TO 230                                           820
  220 WRITE(NW,230)  K                                                  830
  230 DO 240 L=K1,K2                                                    840
      IF(ALT(L).LT.ALTM) GO TO 240                                      850
      KSET=L                                                            860
      GO TO 250                                                         870
  240 CONTINUE                                                          880
  245 CONTINUE                                                          890
  250 FORMAT(37HMAX ALT. NOT FOUND ON TEMP DATA CARDS)                  900
      WRITE(NW,250)                                                     910
  260 CALL INTERP(ALT,STORE,KSET,2,1)                                   920
  261  FORMAT (1X,I5,1P9E12.4)                                          930
      WRITE(NW,261) (J,(CONC(J,I),I=1,IGAS), T(J),J = 1,201)            940
      IGAS=IGAS-1                                                       950
      DO 331 II=1,IGAS                                                  960
```

```
      READ(NR,310)TYPE,NLEV  ,NBAN ,AMAS ,DEXT,DEXB                          970
  310 FORMAT (A8,22X,I2,3X,I2,3X,F6.2,4X,2(E12.4,3X))                        980
      DO 311 J=1,25                                                          990
      IF(TYPE.EQ.DATATY(J)) GO TO 312                                       1000
  311 CONTINUE                                                              1010
      WRITE(NW,370)                                                         1020
  370 FORMAT(36HTHE GAS IS NOT LISTED IN THE PROGRAM)                       1030
C     STOP 1                                                              1040
  312 NSP=J                                                                 1050
      NLEVEL(NSP)=NLEV                                                      1060
      NBAND(NSP)=NBAN                                                       1070
      AMASS(NSP)=AMAS                                                       1080
      CDEXT(NSP)=DEXT                                                       1090
      CDEXB(NSP)=DEXB                                                       1100
      IF(NSP.EQ.2)READ(NR,320) (CO2(I,1),CO2(I,2),I=1,18)                   1110
  320 FORMAT (6(F5.2,F8.2))                                                 1120
      IF(NSP.EQ.3)READ(NR,320) (WATVA(I,1),WATVA(I,2),I=1,18)               1130
      ILEVL=0                                                               1140
      DO 313 IJ=1,NLEV                                                      1150
      ILEVL=ILEVL+1                                                         1160
      READ(NR,330)LSC(NSP,IJ),LCC(NSP,IJ),RCM(NSP,IJ),BV(NSP,IJ),JGAS,     1170
     1JLEVEL                                                                1180
  330 FORMAT (2I6,2E12.4,36X,2I3)                                           1190
      IF(NSP.EQ.JGAS) GO TO 314                                             1200
      WRITE(NW,340) NSP                                                     1210
  340 FORMAT(64HTHE CODE ON THE DATA DOES NOT AGREE WITH THE FIRST CARD     1220
     1WHICH IS,I3)                                                          1230
C     STOP 2                                                              1240
  314 IF(ILEVL.EQ.JLEVEL) GO TO 313                                         1250
      WRITE (NW,350) NSP                                                    1260
  350 FORMAT(42HTHERE IS A CARD OUT OF SEQUENCE IN GAS NO.,I3)              1270
C     STOP 3                                                              1280
```

```
  313 CONTINUE                                                                  1290
      ILEVL=0                                                                   1300
      DO 316 IJ=1,NBAN                                                          1310
      ILEVL=ILEVL+1                                                             1320
      READ(NP,360)LBC(NSP,IJ),LBU(NSP,IJ),STR(NSP,IJ),SFLUX(IJ,NSP),           1330
     1(TFLUX(IJ,NSP,J),J=1,3),JGAS,JLEVEL                                      1340
       STR(NSP,IJ) = 2.4793E+19*STR(NSP,IJ)                                     1350
  360 FORMAT (2I6,1P5E12.4,2I3)                                                 1360
      IF(NSP.EQ.JGAS) GO TO 322                                                 1370
      WRITE (NW,340) NSP                                                        1380
C     STOP 4                                                                    1390
  322 IF(ILEVL.EQ.JLEVEL) GO TO 316                                             1400
      WRITE (NW,350) NSP                                                        1410
C     STOP 5                                                                    1420
  316 CONTINUE                                                                  1430
  331 CONTINUE                                                                  1440
       WRITE (NP,378) T                                                         1450
  378  FORMAT (1P6E12.4)                                                        1460
      DO 900 NLTE = 3, 4                                                        1470
       NSPTOT = 4                                                               1480
       WRITE (NP,360) NSPTOT                                                    1490
      DO 850 I = 2, 7                                                           1500
       IF ((I .EQ. 6) .AND. (NLTE .EQ. 4))  GO TO 850                           1510
       IF ((I .EQ. 7) .AND. (NLTE .EQ. 3))  GO TO 850                           1520
       IF (I .EQ. 5)  GO TO 850                                                 1530
       WRITE (NP,360) I                                                         1540
       WRITE (NP,360) NLEVEL(I),NBAND(I),AMASS(I)                               1550
       NLEV = NLEVEL(I)                                                         1560
       NBND = NBAND(I)                                                          1570
      DO 381 IJ = 1, NLEV                                                       1580
       WRITE (NP,360) LSC(I,IJ),LCC(I,IJ),RCM(I,IJ),BV(I,IJ)                   1590
  381  CONTINUE                                                                 1600
```

```
      DO 382 IJ = 1, NBND                                                    1610
       WRITE (NP,360) LBC(I,IJ),LBU(I,IJ),STR(I,IJ)                          1620
382   CONTINUE                                                               1630
      JALT = (IALT(I) - 1)/5                                                 1640
      WRITE (NP,360) IALT(I),JALT                                            1650
      CALL BANRAD (NLEVEL(I),NBAND(I),IALT(I),CDEXT(I),CDEXB(I),             1660
     1  AMASS(I),LSC(I,1),LCC(I,1),LBC(I,1),LBU(I,1),BV(I,1),STR(I,1),       1670
     2  ROM(I,1),CONC(1,I),SFLUX(1,I),TFLUX(1,I,1))                          1680
      KK=IALT(I)                                                             1690
      JSET=NBAND(I)                                                          1700
      DO 340 J=1,JSET                                                        1710
700   FORMAT('H1,4A6,8HBAND NO.,I3,5X,7HLAMDA =,F6.2,7HMICRO M,4X,4A6//)     1720
      WRITE(NW,700)(XID(I,K),K=1,4),J,LAMDA(J),(XIDT(K),K=1,4)               1730
       IF (NLTE .GT. 1)  GO TO 707                                           1740
704   FORMAT(40H ASSUMES LOCAL THERMODYNAMIC EQUILIBRIUM/)                   1750
705   WRITE(NW,704)                                                          1760
      GO TO 715                                                              1770
706    FORMAT (28H LACK OF THERMAL EQUILIBRIUM ,I6/)                         1780
707   WRITE (NW,706) NLTE                                                    1790
710   FORMAT(118H EXOATMOSPHERIC VIEWING $$  - - - - - - - - - - - - E N D   1800
     1 O A T M O S P H E R I C   V I E W I N G - - - - - - - - - - - - )     1810
715   WRITE (NW,710)                                                         1820
720   FORMAT(118H TAN HT  LIMB BAND RAD  $$  ALTD  - - - - - - BAND RADI     1830
     1ANCE AS A FUNCTION OF ZENITH ANGLE (WATTS/CM2-STER)- - - - - - )       1840
      WRITE (NW,720)                                                         1850
730   FORMAT(118H (KM) (WATTS/CM2-STER) $$    (KM)  0 DEG - UP    20 DEG     1860
     1      40 DEG       60 DEG       70 DEG       80 DEG    90 DEG-HORZ  )  1870
      WRITE (NW,730)                                                         1880
740    FORMAT (F5.0,1PE14.4,0PF12.0,1X,1P7E12.4)                             1890
      KSET=JALT(I)-1                                                         1900
      DO 370 K=1,KSET,5                                                      1910
      IF(K.GT.91) GO TO 750                                                  1920
```

```
      HT=K+59                                                                    1930
      GO TO 760                                                                  1940
  750 HT=150+(K-91)*5                                                            1950
  760 D1=HT+6371.                                                                1960
      DO 810 L=1,7                                                               1970
      M1=K+1                                                                     1980
      PHI=ANG(L)*.017453292                                                      1990
      SAVTAU=0.0                                                                 2000
      STAU(L)=0.0                                                                2010
      SRAD(L)=0.0                                                                2020
      RANG=0.0                                                                   2030
      MSET=IALT(I)                                                               2040
      DO 790 M=M1,MSET                                                           2050
      IF(M.GT.91) GO TO 770                                                      2060
      D2=M+59+6371                                                               2070
      GO TO 780                                                                  2080
  770 D2=150+(M-91)*5+6371                                                       2090
  780 ANG=ASIN(SIN(PHI)*D1/D2)                                                   2100
      DS=SQRT(D1*D1+D2*D2-2.*D1*D2*COS(PHI-ANG))                                 2110
      DELS=(DS-RANG)*1.0E+05                                                     2120
      RANG=DS                                                                    2130
      STAU(L)=STAU(L)+DELS*(0.5*(TAU(M,J)+TAU(M-1,J)))                           2140
      SRAD(L)=SRAD(L)+DELS*0.5*(RAD(M,J)*SM1(MOD(LBJ(I,J),10),STAU(L))           2150
     1+RAD(M-1,J)*SM1(MOD(LBU(I,J),10),SAVTAU))                                  2160
      SAVTAU=STAU(L)                                                             2170
  790 CONTINUE                                                                   2180
  810 CONTINUE                                                                   2190
      SAVTAU=0.0                                                                 2200
      STAU(7)=0.0                                                                2210
      VRANG = 0.0                                                                2220
      SRAD(8)=0.0                                                                2230
      MSET=IALT(I)                                                               2240
```

```
      DO 815 M=K,MSET                                                    2250
      NSHELS = 2*(MSET-K) + 1                                            2260
      L=MSET-M+K                                                         2270
      IF(L.GT.91) GO TO 811                                              2280
      D2=L*59+6371                                                       2290
      GO TO 812                                                          2300
  811 D2=150+(L-91)*5+6371                                               2310
  812 DS=SQRT(D2*D2-D1*D1)                                               2320
      D3 = D2 - 6371.                                                    2330
      IF(L.LT.MSET) GO TO 813                                            2340
      RANG=DS                                                            2350
      SAVT1=1.0                                                          2360
      GO TO 815                                                          2370
  813 DELS=(RANG-DS)*1.0E+05                                             2380
      VRANG = VRANG + DELS                                               2390
      RANG=DS                                                            2400
      STAU(7)=STAU(7)+DELS*(0.5*(TAU(L,J)+TAU(L+1,J)))                   2410
      SAVT2=SM1(MOD(LBU(I,J),10),STAU(7))                                2420
      SRAD(8)=SRAD(8)+DELS*(0.5*(RAD(L,J)*SAVT2+RAD(L+1,J)*SAVT1))       2430
  819 SAVT1=SAVT2                                                        2440
  815 CONTINUE                                                           2450
      M1=K+1                                                             2460
      RANG=0.0                                                           2470
      MSET=IALT(I)                                                       2480
      DO 818 M=M1,MSET                                                   2490
      IF(M.GT.91) GO TO 816                                              2500
      D2=M*59+6371                                                       2510
      GO TO 817                                                          2520
  816 D2=150+(M-91)*5+6371                                               2530
  817 DS=SQRT(D2*D2-D1*D1)                                               2540
      D3 = D2 - 6371.                                                    2550
      DELS=(DS-RANG)*1.0E+05                                             2560
```

```
      VRANG = VRANG + DELS                                                    2570
      RANG=DS                                                                 2580
      STAU(7)=STAU(7)+DELS*(0.5*(TAU(M,J)+TAU(M-1,J)))                        2590
      SAVT2=SM1(MOD(LBU(I,J),10),STAU(7))                                     2600
      SRAD(8)=SRAD(8)+DELS*(0.5*(RAD(M,J)*SAVT2+RAD(M-1,J)*SAVT1))            2610
  820 SAVT1=SAVT2                                                             2620
  818 CONTINUE                                                                2630
      WRITE(NP,4444) HT,SRAD(8),SRAD(1),SRAD(7)                               2640
 4444 FORMAT (F6.1,1P3E17.4)                                                  2650
      WRITE(NW,740) HT,SRAD(8), HT,(SRAD(M), M= 1,7)                          2660
  830 CONTINUE                                                                2670
  840 CONTINUE                                                                2680
  850 CONTINUE                                                                2690
  900 CONTINUE                                                                2700
      REWIND 1                                                                2710
      REWIND 2                                                                2720
      STOP 77                                                                 2730
      END                                                                     2740
```

```
      SUBROUTINE INTERP(ALT,STORE,KSET,MODE,I)                          10
      DIMENSION ALT(30),STORE(30)                                       20
      COMMON CONC(201,10),TAU(201,25),RAD(201,25),T(201),LAMDA(25),     30
     1TOTDEN(201)                                                       40
      J=1                                                               50
      A=60.                                                             60
    8 DO 120 K=1,KSET                                                   70
   10 IF(A-ALT(K))80,20,20                                              80
   20 GO TO (30,40),MODE                                                90
   30 CONC(J,I)=STORE(K)                                               100
      GO TO 50                                                         110
   40 T(J)=STORE(K)                                                    120
   50 IF(K.LT.KSET) GO TO 60                                           130
      RETURN                                                           140
   60 IF(J.GE.91) GO TO 70                                             150
      A=A+1.                                                           160
      GO TO 110                                                        170
   70 A=A+5.                                                           180
      GO TO 110                                                        190
   80 FR=(A-ALT(K-1))/(ALT(K)-ALT(K-1))                                200
      GO TO (90,100), MODE                                             210
   90 CONC(J,I)=STORE(K-1)+FR*(STORE(K)-STORE(K-1))                    220
   91 J=J+1                                                            230
      IF(J.GT.91) GO TO 92                                             240
      A=A+1.                                                           250
      GO TO 10                                                         260
   92 A=A+5.                                                           270
      GO TO 10                                                         280
  100 T(J)=STORE(K-1)+FR*(STORE(K)-STORE(K-1))                         290
      GO TO 91                                                         300
  110 J=J+1                                                            310
  120 CONTINUE                                                         320
```

```
330
340
```

```
RETURN
END
```

Subroutine BANRAD

Purpose: BANRAD computes the population of vibrational levels of infrared active species, including the effects of collisional excitation; absorption of radiation from the lower atmosphere, sun and molecules radiating at high altitude. The values returned to the main program are volume radiance and unit optical thickness at each altitude for each band included in the computation.

Method: Subroutine BANRAD computes the population of vibrational levels of infrared active species, including the effects of collisional excitation and absorption radiation from the lower atmosphere, sun and molecules radiating at high altitude. The values returned to the main program are volume radiance and unit optical thickness at each altitude for each band included in the computation.

The collisional processes which are assumed to effect the vibrational state of a molecule are (1) tranlational-vibrational (T-V) interactions in which translation energy is transformed into vibrational energy or vice-versa during a collision, (2) intermolecular vibrational exchange (V-V) in which a molecule of one species exchanges a quantum of vibrational energy with a molecule of another species, and (3) intramolecular vibrational exchange in polyatomic molecules in which a collision transfers the energy in one mode of vibration into another mode within the same molecule.

When a molecule in the ground state collides with another molecule there is a finite probability P_{TV} that a portion of the relative translational energy will be transformed to vibrational energy if the translational energy is equal to or exceeds the vibrational energy. Similarly, if the molecule is in an excited vibrational state, there is a finite probability P_{VT} that the vibrational energy will be converted to relative translational energy during the collision. For a Boltzmann distribution of translational energies at a given temperature

$$P_{TV} = P_{VT} \frac{g_u}{g_l} \exp\left(-\frac{hc}{\lambda_o kT}\right)$$

where g_u and g_l are the statistical weights of the upper and lower vibrational levels, h is Planck's constant (6.6256×10^{-27} erg-sec), c is the velocity of light (2.9979×10^{10} cm/sec), k is the Boltzmann constant (1.38054×10^{-16} erg/K), T is absolute temperature (K) and λ_o is the wavelength (cm) corresponding to the energy difference of the two vibrational levels. Similar expressions hold for intermolecular and intramolecular V-V transfer.

The rate at which a vibrational level of a single molecule is excited (K_{TV}) or de-excited (K_{VT}) is given by the product of the collision frequency Z and the transition probability. The subroutine uses temperature dependent rate coefficients k_{TV}, k_{VT}, etc., such that if [M] is the concentration of the collision partner, $K_{VT} = k_{VT}[M]$, etc.

The radiative processes which are assumed to affect the vibrational state of a molecule are (1) absorption of solar radiation, (2) absorption of radiation from the lower atmosphere, (3) absorption of radiation emitted by molecules in the upper atmosphere and (4) spontaneous emission at a rate $A_{ul}(sec^{-1})$, the Einstein A coefficient. Because absorbing molecules are present between each source of radiation and the molecule for which the vibrational state is to be computed, radiative transfer functions must be used to take into account the effects of absorption. The model assumes that individual lines have a Doppler (thermal velocity broadened) spectral contour.

At a given altitude, there will be N_l molecules/cm^3 in the ground vibrational state of a molecule and N_u molecules/cm^3 in an upper state. Molecular nitrogen and oxygen are assumed to be the only collision partners effective in transfer of vibrational quanta, and their concentrations are used to determine total vibrational excitation rates K_{TV}, K_{VT}, etc. The treatment below refers to a two state molecule.

In unit volume, molecules in the ground state are excited to the upper state by:

(1) Collisional T-V excitation at a rate $N_l K_{TV}$

(2) Collisional V-V excitation at a rate $N_l K_{MV}$ (the subscript MV is used for excitation, VM for de-excitation, by transfer of vibrational quanta).

(3) Absorption of solar flux at a rate $N_1 I_o G' M_1^b(k_s)$ where G' is molecular cross section for absorption of radiative flux

(4) Absorption of lower atmospheric radiance at a rate $N_i N_{\lambda,t} G' N_{12}^b(k_t)$

(5) Absorption of radiance emitted at other levels of the atmosphere at a rate $N_1 \int N_{\lambda,D} G' N_{21}^b(k_D)\, dz$

Molecules in the upper state are de-excited by:

(1) Collisional V-T de-excitation at a rate $N_u K_{VT}$

(2) Collisional V-V de-excitation at a rate $N_u K_{VM}$

(3) Spontaneous radiative de-excitation at a rate $N_u A_{u1}$

Excitation rates are assumed to balance de-excitation rates and the equation

$$\frac{dN_u}{dt} = N_1(k_{TV} + K_{MV} + I_o G' M_1^b(k_s) + N_{\lambda,t} G' N_{12}^b(k_t)$$

$$+ \int N_{\lambda,D} G' N_{21}^b(k_D)\, dZ - N_1(K_{VT} + K_{VM} + A_{u1}) = 0$$

is solved by iteration to determine N_u and N_1 at all altitudes.

Program Flow:

The first step in the computation is determination of the thermal equilibrium populations of vibrational levels. This is performed in the DO range ending with statement 130. Each passage through this DO range computes populations for one altitude. Within this DO range are two others. The DO range ending with statement 110 computes the vibrational population of each level relative to the ground vibrational level, using the energies (wave-number) stored in WAVE ant the statistical weights stored temporarily in real array VIBPOP. The sum of relative populations is stored in SUM. The final ground state population is then computed by dividing total concentration by SUM, and stored in real array GNDSTE. The DO range ending with statement 120 then computes the vibrational population of each level by multiplying total concentration by relative concentration divided by SUM.

The DO range ending with statement 135 selects the vibrational level which is coupled with the first excited nitrogen vibrational level.

The DO range ending with statement 160 computes the rate at which a single molecule in a given level is excited or de-excited by collisions at each altitude.

The outer DO range ending with statement 180 computes wavelengths (BNDLAM), Einstein A coefficients (EA), rates for absorption of radiation from the sun (GDOWN) and from the earth and lower atmosphere (GUP) and the G factor used in computing radiative transfer within the upper atmosphere (GFAC). The inner DO range ending

with Statement 180 computes the unit optical thicknesses for each band at each altitude using Function TAUMAX, and stores the result in array BNDTAU. In addition, it computes the volume radiances and stores them in array BNDRAD.

At this point, if NLTE is 1, computation for thermal equilibrium is complete and control is transferred to Statement 900. If NLTE is not 1, the computation for non-equilibrium is performed iteratively in the DO range ending with Statement 400. The DO range ending with Statement 190 computes various integrated optical thicknesses for each band. The DO range ending with Statement 182 computes the total thickness between each two altitudes in the vertical direction and stores the values in array TTH.

The DO range ending with Statement 184 (1) computes the optical thickness between the base altitude and each other altitude, storing the result in array TTUP and (2) computes the total optical thickness between the top altitude and each other altitude, storing the results in array TTDOWN. If the computation is not for daytime conditions, the DO range ending with Statement 186 sets the values in TTDOWN equal to 0.0, for efficiency in the later computations. If the computation is for the effects of collisional excitation only, the DO range ending with Statement 188 sets the values in TTUP equal to 0.0

The computation of vibrational populations now begins. Control is transferred to

Statement 210 if I is 1, CH_4

Statement 220 if I is 2, CO_2

Statement 230 if I is 3, H_2O

Statement 240 if I is 4, NO

Statement 250 if I is 5, N_2O

Statement 230 if I is 6, pre-dawn Ozone

Statement 230 if I is 7, noon Ozone

Following each of these statements is a DO range in which vibrational populations are computed. A discussion of two statements in the computation for CO_2 will illustrate the computational details. First, look at the statement in the DO range immediately following Statement 221 that begins VIBPOP(NZ, 8) = which computes the population of the first level of the ν_3 vibrational mode. This vibrational level is coupled to the nitrogen first vibrational level by collision, to the upward flux from the earth and lower atmosphere, to solar flux, and to lower vibrational levels by radiation. Excitation by absorption of radiation from within the upper atmosphere is neglected for this level. A ground state molecule is excited by collision with nitrogen at the rate VVNSIG(NZ) (collision exchange), by absorption of solar flux at a rate GDOWN(9)*SM1(KB(9), TDOWN2(NZ)) (equals $I_o G' M_1^b(k_s)$) and by absorption of flux from the earth and lower atmosphere at a rate GUP(9)*SLM1(KB(9), TTUP2(NZ)) (equals $N_{\lambda,t} G' N_1^b(k_t)$). An excited molecule is de-excited by radiation at the rate EA(9) (other Einstein A coefficients are negligible), by vibrational exchange with nitrogen at a rate VVRSIG(NZ), and by transfer of vibrational energy to other modes within the CO_2 molecule at a rate VVDSIG(NZ). The vibrational population is then the ground state population times the ratio of excitation

and de-excitation rates. Second, look at the statement beginning POP2 = ..., which computes the vibrational population of level 2, the first excited level of the ν_2 vibrational mode. This level is excited from the ground state by collisional excitation at a rate TVS = TVSIG, by absorption of radiation from the lower atmosphere and earth at a rate GUP(1)*SLM1(KB(1), TTUP(NZ, 1)) (equals $N_{\lambda, t} G' N_1^b(k_t)$), by absorption of radiation from other points in the upper atmosphere in the 15.0 μm fundamental band at a rate 2π*GFAC(1)*(XFER(VIBPOP(1, 2), TTH(1, 1), NZ, 1, 1, KB(1)) + XFER(VIBPOP(NZ, 2), TTH(NZ, 1), NALT, NZ, 2, KB(1)))*EA(1) $\left(\text{equals } \int N_{\lambda, D} G' N_2^b(k_D) dz\right)$ and by absorption of solar radiation at the rate GDOWN(1)*SMI (KB(1), TTDOWN(NZ, 1)) (equals $I_o G' M_1(k_s)$). It is excited from levels 3, 4 and 7 by collisional excitation VTS and by radiation EA(2), EA(3) and EA(7). De-excitation occurs by radiation EA(1), and by collisional de-excitation to the ground state at a rate VTS = VTSIG(NZ), and by collisional excitation to higher states at a rate TVS. The approximate vibrational population is then the ratio of the sums of population rates times higher state populations divided by the sum of de-excitation rates for single molecule.

After each set of vibrational populations is completed, new values for volume band radiances are computed in the nested DO ranges ending with Statement 360.

After three iterations to compute vibrational populations control is returned to the main program.

Inputs: NLEVEL, NBAND, NALT, CDEXT, CDEXB, RM, LSC, LCC, I, LBC, LBU, BV, STR, WAVE, CONC, SFLUX, TFLUX, KCHEM, NR, NP, NW, NLTE, TEMP, BNDLAM, DENTOT, NBAND1.

Outputs: VIBPOP, BNDTAU, BNDRAD

Mnemonic Variables:

AA	intermediate f .ctor in computing vibrational populations
AAA	quantity used in NO chemiluminescence calculation in original Subroutine BANRAD application; it has no effect at present
BNDL	wavelength of current interest (μm)
BNDLAM	wavelength array (μm)
BNDRAD	volume radiance
BNDTAU	unit optical thickness
BV	rotational constant (cm^{-1})
CDEXB	rate coefficient at 300 K for vibrational de-excitation of lowest NO vibrational level ($cm^3\ sec^{-1}$)
CDEXT	rate coefficient at 300 K for vibrational exchange of NO with N_2 or O_2 for the most significant vibrational level ($cm^3\ sec^{-1}$)
CHEML	currently inactive
CONC	array of NO concentrations (cm^{-3})
CO2	real array used to store reduced temperature value and rate coefficients to generate vibrational energy exchange rates for CO_2: currently inactive
DENTOT	array of total number densities at altitude mesh (cm^{-3})

EA	array of Einstein A coefficients
FAC	used as temporary storage
FACTOR	temporary storage in computing vibrational exchange
GDOWN	real array in which are stored rates of absorption of solar radiation, $I_o G'$ for each band
GFAC	real array in which are stored values of G' for each band
GNDSTE	real array in which are stored ground state vibrational populations, equivalenced to VIBPOP(1, 1)
GUP	real array in which are stored rates of absorption from the lower atmosphere, $2\pi N^b_{\lambda,t} G'$ for each band
I	index of species of interest (equals 4 for NO)
ITER	integer used as index in iterative computation of vibrational level population
IX	index for altitude level
JF	index for final vibrational level in a band transition
JI	index for initial vibrational level in a band transition
JL	integer used to compute JI, JF
JX	index for vibrational level
KB	integer array used to store band types
KBN	index of band type
KCHEM	chemiluminescense index; currently has no effect
KL	integer array storing level types
KLN	current level type
LBC	integer array containing information on which levels produce a band
LBU	integer array containing band type information
LCC	integer array containing vibrational level coupling go N_2 or O_2

LSC	integer array containing level type information
NALT	number of altitudes for which results are to be returned
NB	index for band number
NBAND	number of bands for this species
NBAND1	equal to NBAND
NBC	temporary storage index
NBX	temporary storage index
NF	temporary storage index
NI	temporary storage index
NL	level number index
NLEVEL	number of vibrational levels for this species
NLTE	integer variable used to indicate conditions of computation, 1 for local thermodynamic equilibrium, 2 for collisional excitation only, 3 for night conditions, 4 for noon conditions, 5 to include high vibrational temperature for nitrogen
NMIN	integer variable storing (NALT - 1)
NP	output device index
NR	input device index
NW	output device index
NY, NZ	index for altitude level
POP2, POP3, POP4, POP5, POP6, POP7, POP8	real variables used as temperature storage for vibrational populations
POWER	real array in which are stored conversion factors between emission rates in photons cm^{-3} sec^{-1} and watts cm^{-3} $ster^{-1}$ for each band
RM	molecular mass for this molecule (g/mole)

SFLUX	array of solar flux values for each band
SNB	temporary storage of band strength ($cm^{-1}(atm\text{-}cm)^{-1}$)
STR	array of band strength for each band ($cm^{-1}(atm\text{-}cm)^{-1}$)
SUM	used to store partial sums
TDOWN2	real array equivalenced to TTDOWN(1, 9)
TEMP	array containing temperature profile (K)
TFLUX	array containing fluxes for lower altitudes for each band
TMP	temporary storage for temperature (K)
TNNV	nitrogen vibrational temperature (K)
TTDOWN	array storing values of total optical thickness from top of atmosphere for each band
TTH	array storing values of total optical thickness between adjacent altitudes for each band
TTUP	array storing values of total optical thickness from bottom of upper atmosphere for each band
TTUP2	array equivalenced to TTUP(9, 1)
TVS	used to store local value of TVSIG
TVSIG	array storing values of vibrational populations of each state for each altitude
VIBPOP	array storing values of vibrational populations of each state for each altitude
VTS	local value of VTSIG
VTSIG	array of vibrational de-excitation rates at each altitude
VVDSIG	array of estimated values of transfer of vibrational energy (downward)
VVNSIG	array of rates of transfer of vibrational energy from N_2 to this molecule
VVRSIG	array of rates of transfer of vibrational energy to N_2

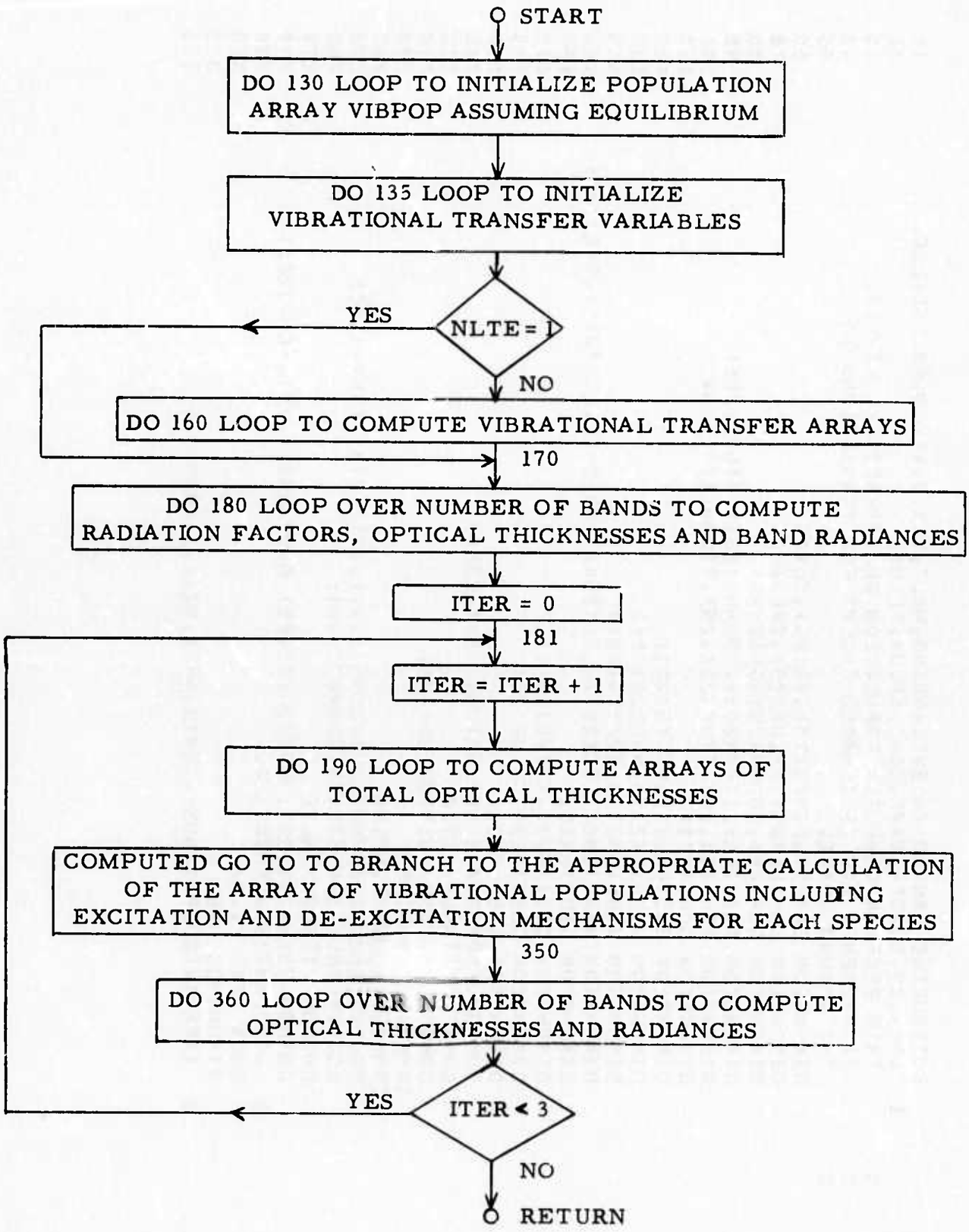
VVUSIG	array of estimated values of rate of transfer of vibrational exchange (upward)
VVWAVE	difference between N_2 vibrational energy and a molecular vibrational energy
WATVA	array storing reduced temperature values and rate coefficients to generate vibrational energy exchange rates for CO_2
WAVE	array containing energy level of each band (cm^{-1})
XYM	temporary storage variables
XYMM	temporary storage variable
XYN	temporary storage variable

Non-Library Externals:

EVAL, SIMP, SLM1, SM1, TAUMAX, XFER

Called By: BCKGND

FLOW CHART: BANRAD



```
      SUBROUTINE BANRAD (NLEVEL,NBAND,NALT,TEXCO,VEXCO,RM,LSC,LCC,
     1  LBC,LBU,BW,STR,WAVE,CONC,SFLUX,TFLUX)
C       THIS SECTION COMPUTES POPULATION OF VIBRATIONAL LEVELS.
C       AT PRESENT, TAU IS IGNORED EXCEPT FOR TRANSITIONS TO
C       THE GROUND STATE.
      DIMENSION EA(25),POWER(25),KB(25),KL(25)
      DIMENSION EAHALF(25),TFLUX(25),SFLUX(25)
      DIMENSION GUP(25),GDOWN(25),GFAC(25)
      DIMENSION TR2(201),TTH2(201),TDOWN2(201),TTUP2(201)
      DIMENSION TTH(201,25),TTUP(201,25),TTDOWN(201,25)
      DIMENSION CONC(201)
      DIMENSION TVSIG(201),VTSIG(201)
      DIMENSION VVUSIG(201),VVDSIG(201)
      DIMENSION VVMSIG(201),VVRSIG(201)
      DIMENSION WAVE(250),LSC(250),LCC(250),LBC(250),LBU(250),STR(250)
      DIMENSION GNDSTE(201)
      DIMENSION CO2(18,2),WATVA(18,2)
      DIMENSION CDEXT(25),CDEXB(25)
      DIMENSION CHEML(201)
      COMMON /VBRCOM/ I,CO2,WATVA,CDEXT,CDEXB
      COMMON /ITRCOM/ ITER
      COMMON /POPCOM/ VIBPOP(201,25)
     COMMON/KCEM/KCHEM
     COMMON/INOUT/NR,NP,NW
      EQUIVALENCE (TDOWN2(1),TTDOWN(1,9)),(TTUP2(1),TTUP(1,9))
      EQUIVALENCE (GNDSTE(1),VIBPOP(1,1))
      COMMON /TEQCOM/ NLTE
      COMMON TDEN(201,10),BNDTAU(201,25),BNDRAD(201,25),TEMP(201)
     1  ,BND_AM(25),DENTOT(201)
      DATA GFAC /1.0/
      RTFUN(NUS,NBS) =
     1  (XFER(VIBPOP(1,NUS),TTH(1,NBS),NZ,1,1,KB(NBS))
```

```
     2   + XFER(VIBPOP(NZ,NUS),TTH(NZ,NBS),NALT,NZ,2,<B(NBS)))*EA(NBS)          330
     3   +GDOWN(NBS)*SM1(KB(NBS),ZFAC*TTDOWN(NZ,NBS))                          340
     4   +GUP(NBS)*SLM1(KB(NBS),TTUP(NZ,NBS))                                  350
         AAA = 0.0                                                             360
         IF (NLTE .EQ. 4)  AAA = 2.0                                           370
         IF (KCHEM .EQ. 1)  READ(NR,50)  CHEML                                 380
  50     FORMAT (6E12.4)                                                       390
         IF (KCHEM .EQ. 1)  KCHEM = 0                                          400
        DO 130 NZ = 1, NALT                                                    410
         SUM = 0.0                                                             420
         TMP = TEMP(NZ)                                                        430
        DO 110 NL = 2, NLEVEL                                                  440
         JL = (NL - 1)*25 + 1                                                  450
        XYN = MOD(LSC(JL)/10,10)                                               460
         AA = EXP(-1.43879*WAVE(JL)/TMP)*XYN                                   470
         VIBPOP(NZ,NL) = AA                                                    480
         SUM = SUM + AA                                                        490
  110    CONTINUE                                                              500
         SUM = SUM + 1.0                                                       510
         GNDSTE(NZ) = CONC(NZ)/SUM                                             520
        DO 120 NL = 2, NLEVEL                                                  530
         VIBPOP(NZ,NL) = CONC(NZ)*VIBPOP(NZ,NL)/SUM                            540
  120    CONTINUE                                                              550
  130    CONTINUE                                                              560
        DO 135 NL = 2, NLEVEL                                                  570
         JL = (NL - 1)*25 + 1                                                  580
         IF (LOC(JL) .EQ. 1)  VVWAVE = 2331.- WAVE(JL)                         590
         IF (LOC(JL) .EQ. 2)  VVWAVE = 1556.4 - WAVE(JL)                       600
         FAC = 2331.0                                                          610
         IF (LOC(JL) .EQ. 2)  FAC = 1556.4                                     620
  135    CONTINUE                                                              630
         IF(NLTE.EQ.1) GO TO 170                                               640
```

```
      DO 160 NZ = 1,NALT                                                    650
       TEMPNZ = TEMP(NZ)                                                    660
       TEMP13 = EXP(ALOG(TEMPNZ)/3.0)                                       670
       IF (I .EQ. 3)  GO TO 136                                             680
       IF (I .NE. 2)  GO TO 140                                             690
       VTSIG(NZ) = 6.69E-10*EXP(-84.07/TEMP13)                              700
       VVRSIG(NZ) = 1.71E-6*EXP(-175.3/TEMP13)                              710
     1   + 6.37E-14*EXP(15.27/TEMP13)                                       720
       VVDSIG(NZ) = 1.0E-15 + 5.16E-11*EXP(-76.75/TEMP13)                   730
       VVNSIG(NZ) = VVRSIG(NZ)/EXP(1.43879/TEMP(NZ)*18.0)                   740
       VVJSIG(NZ) = VVDSIG(NZ)*VIBPOP(NZ,8)/VIBPOP(NZ,5)                    750
       GO TO 150                                                            760
  136  CONTINUE                                                             770
       FAC = 0.0                                                            780
       VVNSIG(NZ) = 4.0E-13                                                 790
       VTSIG(NZ) = 5.37E-10*EXP(-70.0/TEMP13)                               800
       VVRSIG(NZ) = VVNSIG(NZ)*EXP(-1.43879/TEMP(NZ)*1594.736)              810
       GO TO 150                                                            820
  140  CONTINUE                                                             830
       VTSIG(NZ) = CDEXT(I)                                                 840
       VVNSIG(NZ) = CDEXB(I)                                                850
       VVRSIG(NZ) = VVNSIG(NZ)*EXP(-1.43879/TEMP(NZ)*VVWAVE)                860
  150  CONTINUE                                                             870
       TVSIG(NZ) = VTSIG(NZ)/GNDSTE(NZ)*VIBPOP(NZ,2)                        880
       VTSIG(NZ) = VTSIG(NZ)*DENTOT(NZ)                                     890
       TVSIG(NZ) = TVSIG(NZ)*DENTOT(NZ)                                     900
       TNNV = TEMP(NZ)                                                      910
       IF (NLTE .LT. 5)  GO TO 154                                          920
       IF (NZ .GT. 41 .AND. NZ .LT. 51)  TNNV = TEMP(NZ)+FLOAT(NZ-41)       930
     1   *280.0                                                             940
       IF (NZ .GE. 51)  TNNV = 3000.0                                       950
  154  CONTINUE                                                             960
```

```
      FACTOR = EXP(-1.43879*FAC/TINV)                                    970
      VVNSIG(NZ) = VVNSIG(NZ)*DENTOT(NZ)*FACTOR/(1.0 + FACTOR)           980
      VVRSIG(NZ) = VVRSIG(NZ)*DENTOT(NZ)/(1.0 + FACTOR)                  990
160   CONTINUE                                                          1000
      DO 166 NZ = 1, NALT                                               1010
      SUM = 0.0                                                         1020
      TMP = 180.0                                                       1030
      DO 162 NL = 2, NLEVEL                                             1040
      JL = (NL - 1)*25 + 1                                              1050
      XYN = MOD(LSC(JL)/10,10)                                          1060
      AA = EXP(-1.43879*WAVE(JL)/TMP)*XYN                               1070
      VIBPOP(NZ,NL) = AA                                                1080
      SUM = SUM + AA                                                    1090
162   CONTINUE                                                          1100
      SUM = SUM + 1.0                                                   1110
      GNDSTE(NZ) = CONC(NZ)/SUM                                         1120
      DO 164 NL = 2, NLEVEL                                             1130
      VIBPOP(NZ,NL) = CONC(NZ)*VIBPOP(NZ,NL)/SUM                        1140
164   CONTINUE                                                          1150
166   CONTINUE                                                          1160
170   CONTINUE                                                          1170
      DO 175 IX = 1, NALT                                               1180
175   CONTINUE                                                          1190
      DO 180 NB = 1, NBAND                                              1200
      NBX = (NB - 1)*25 + 1                                             1210
      NBC = LBC(NBX)                                                    1220
      NI = MOD(NBC/100,100)                                             1230
      NF = MOD(NBC,100)                                                 1240
      JI = (NI - 1)*25 + 1                                              1250
      JF = (NF - 1)*25 + 1                                              1260
      KL(NB) = MOD(LBU(NBX)/10,10)                                      1270
      KB(NB) = MOD(LBU(NBX),10)                                         1280
```

```
      TWAVE = WAVE(JI) - WAVE(JF)                                       1290
      BNDLAM(NB) = 1.0E+4/TWAVE                                         1300
      IF (TFLUX(NB) .EQ. 0.0)  TFLUX(NB) =                              1310
     1  5.997E+6*TWAVE**4/(EXP(1.43879*TWAVE/TEMP(1)) - 1.0)            1320
      XYM = MOD(LSC(JF)/10,10)                                          1330
     XYMM = MOD(LSC(JI)/10,10)                                          1340
      EA(NB) = STR(NBX)*2.804/BNDLAM(NB)**2*XYM                         1350
     1  *EXP(1.43879*WAVE(JF)/300.0)/XYMM                               1360
      POWER(NB) = 1.58065E-20/BNDLAM(NB)                                1370
      GFAC(NB) = 3.72E-24*STR(NBX)*BNDLAM(NB)**2                        1380
     1  *EXP(1.43879*WAVE(JF)/296.0)                                    1390
      GUP(NB) = GFAC(NB)*TFLUX(NB)*12.4864                              1400
      IF (NLTE .LT. 3)  GUP(NB) = 0.0                                   1410
      GDOWN(NB) = GFAC(NB)*SFLUX(NB)                                    1420
      IF (NLTE .LT. 4)  GDOWN(NB) = 0.0                                 1430
      KBN = KB(NB)                                                      1440
      KLN = KL(NB)                                                      1450
      BNDL = BNDLAM(NB)                                                 1460
      FAC = POWER(NB)*EA(NB)                                            1470
      SNB = STR(NBX)*EXP(1.43879*WAVE(JF)/296.0)                        1480
     DO 180 NZ = 1, NALT                                                1490
      BNDTAU(NZ,NB) = TAUMAX(KLN,KBN,RM,BV,TEMP(NZ),BNDL,SNB,           1500
     1  VIBPOP(NZ,NF))                                                  1510
      IF (NLTE .GT. 1)  GO TO 180                                       1520
      BNDRAD(NZ,NB) = VIBPOP(NZ,NI)*FAC                                 1530
  180 CONTINUE                                                          1540
      IF (NLTE .EQ. 1)  GO TO 900                                       1550
      ITMAX = 4                                                         1560
      IF (I .EQ. 2)  ITMAX = 10                                         1570
     ITER = 0                                                           1580
  181 ITER = ITER + 1                                                   1590
      NMIN = NALT - 1                                                   1600
```

```
      DO 190 NB = 1, NBAND                                                    1610
       TTDOWN(NALT,NB) = 0.0                                                  1620
       TTUP(1,NB) = 0.0                                                       1630
       TTH(1,NB) = SIMP(BNDTAU(1,NB),1.0E+5,1)                                1640
      DO 182 NZ = 2, NMIN                                                     1650
       TTH(NZ,NB) = SIMP(BNDTAU(NZ - 1,NB),1.0E+5,2)                          1660
  182  CONTINUE                                                               1670
      DO 184 NZ = 2, NMIN                                                     1680
       NY = NALT - NZ + 1                                                     1690
       TTUP(NZ,NB) = TTUP(NZ - 1,NB) + TTH(NZ - 1,NB)                         1700
       TTDOWN(NY,NB) = TTDOWN(NY + 1,NB) + TTH(NY,NB)                         1710
  184  CONTINUE                                                               1720
       TTDOWN(1,NB) = TTDOWN(2,NB) + TTH(1,NB)                                1730
       TTUP(NALT,NB) = TTUP(NMIN,NB) + TTH(NMIN,NB)                           1740
       IF (NLTE .GT. 3)  GO TO 190                                            1750
      DO 186 NZ = 1, NALT                                                     1760
       TTDOWN(NZ,NB) = 0.0                                                    1770
  186  CONTINUE                                                               1780
       IF (NLTE .GT. 2)  GO TO 190                                            1790
      DO 188 NZ = 1, NALT                                                     1800
       TTUP(NZ,NB) = 0.0                                                      1810
  188  CONTINUE                                                               1820
  190  CONTINUE                                                               1830
  200  CONTINUE                                                               1840
       GO TO (210,220,230,240,250,260,260), I                                 1850
  210  CONTINUE                                                               1860
      DO 215 NZ = 1, NALT                                                     1870
       VIBPOP(NZ,3) = GNDSTE(NZ)*(GUP(2)*SLM1(KB(2),TTUP(NZ,2))               1880
     1  + GDOWN(2)*SM1(KB(2),TTDOWN(NZ,2)))/(EA(2) + EA(4))                   1890
       VIBPOP(NZ,2) = (EA(4)*VIBPOP(NZ,3) + GNDSTE(NZ)                        1900
     1  *(GUP(1)*SLM1(KB(1),TTUP(NZ,1))                                       1910
     2  + GDOWN(1)*SM1(KB(1),TTDOWN(NZ,1))                                    1920
```

```
     3   + TVSIG(NZ) + 0.75*VVNSIG(NZ)))                                    1930
     4   /(EA(1) + VTSIG(NZ) + 0.75*VVRSIG(NZ))                             1940
         VIBPOP(NZ,4) = GNDSTE(NZ)*(0.25*VVNSIG(NZ)                         1950
     1   + GUP(3)*SLM1(KB(3),TTUP(NZ,3))                                    1960
     2   + GDOWN(3)*SM1(KB(3),TTDOWN(NZ,3)))                                1970
     3   /(EA(3) + 0.25*VVRSIG(NZ))                                         1980
 215 CONTINUE                                                               1990
         GO TO 350                                                          2000
 220 CONTINUE                                                               2010
       NZ = 0                                                               2020
 221 NZ = NZ + 1                                                            2030
         VIBPOP(NZ,8) = GNDSTE(NZ)*(VVNSIG(NZ) + RTFUN(8,9))                2040
     2   /(VVRSIG(NZ) + EA(9) + VVDSIG(NZ))                                 2050
         TVS = TVSIG(NZ)                                                    2060
         VTS = VTSIG(NZ)                                                    2070
         POP3 = VIBPOP(NZ,3)                                                2080
         POP4 = VIBPOP(NZ,4)                                                2090
         POP7 = VIBPOP(NZ,7)                                                2100
         POP2 = ((TVS + RTFUN(2,1))*GNDSTE(NZ)                              2110
     2   + POP3*(EA(2) + VTS) + POP4*(EA(3) + VTS)                          2120
     3   + POP7*(EA(7) + VTS))/(VTS + EA(1) + TVS)                          2130
         VIBPOP(NZ,2) = POP2                                                2140
         POP8 = VIBPOP(NZ,8)                                                2150
         POP7 = ((TVS*0.25 + GUP(7)*SLM1(KB(7),TTUP(NZ,7))                  2160
     1   + GDOWN(7)*SM1(KB(7),TTDOWN(NZ,7)))*POP2                           2170
     2   + EA(8)*POP8)/(EA(7) + VTS)                                        2180
         VIBPOP(NZ,7) = POP7                                                2190
         POP5 = VIBPOP(NZ,5)                                                2200
         POP6 = VIBPOP(NZ,6)                                                2210
         POP3 = (POP2*(0.25*TVS + GUP(2)*SLM1(KB(2),TTUP(NZ,2))             2220
     1   + GDOWN(2)*SM1(KB(2),TTDOWN(NZ,2))) + POP5*(EA(4) + VTS))          2230
     2   /(EA(2) + VTS + TVS)                                               2240
```

```
      VIBPOP(NZ,3) = POP3                                                    2250
      POP4 = (POP2*(0.5*TVS + GUP(3)*SLM1(KB(3),TTUP(NZ,3))                  2260
     1  + GDOWN(3)*SM1(KB(3),TTDOWN(NZ,3))) + POP5*EA(5)                     2270
     2  + POP6*(EA(6) + VTS))/(EA(3) + VTS + TVS)                            2280
      VIBPOP(NZ,4) = POP4                                                    2290
      POP5 = (POP3*(TVS + GUP(4)*SLM1(KB(4),TTUP(NZ,4))                      2300
     1  + GDOWN(4)*SM1(KB(4),TTDOWN(NZ,4)))                                  2310
     2  + POP4*(GUP(5)*SLM1(KB(5),TTUP(NZ,5))                                2320
     3  + GDOWN(5)*SM1(KB(5),TTDOWN(NZ,5))) + POP8*VVDSIG(NZ))               2330
     4  /(EA(4) + EA(5) + VTS + VVUSIG(NZ))                                  2340
      VIBPOP(NZ,5) = POP5                                                    2350
      POP6 = (POP4*(0.5*TVS + GUP(6)*SLM1(KB(6),TTUP(NZ,6))                  2360
     1  + GDOWN(6)*SM1(KB(6),TTDOWN(NZ,6))))/(EA(6) + VTS)                   2370
      VIBPOP(NZ,6) = POP6                                                    2380
      GNDSTE(NZ) = CONC(NZ) - (POP8 +POP6 + POP5 + POP7 + POP4               2390
     1  + POP3 + POP2)                                                       2400
      IF (GNDSTE(NZ) .LE. 0.0)  GNDSTE(NZ) = 0.05*CONC(NZ)                   2410
  225 IF (NZ.LT.NALT) GO TO 221                                              2420
      GO TO 350                                                              2430
  230 CONTINUE                                                               2440
      DO 235 NZ = 1, NALT                                                    2450
      VIBPOP(NZ,2) = ((TVSIG(NZ) +VVRSIG(NZ)                                 2460
     1  + RTFUN(2,1))*GNDSTE(NZ) +VIBPOP(NZ,3)*EA(5))                        2470
     2  /(VTSIG(NZ) + EA(1) + VVNSIG(NZ))                                    2480
      VIBPOP(NZ,3) = ((TVSIG(NZ) + VVRSIG(NZ)                                2490
     1  + RTFUN(3,5))*VIBPOP(NZ,2) +                                         2500
     2  RTFUN(3,2)*GNDSTE(NZ) + VIBPOP(NZ,6)*EA(11))                         2510
     3  /(EA(2) + EA(5) + VTSIG(NZ) + VVNSIG(NZ))                            2520
      VIBPOP(NZ,4) = (RTFUN(4,3)*GNDSTE(NZ)                                  2530
     1  + RTFUN(4,6)*GNDSTE(NZ))/(EA(3) + EA(6))                             2540
      VIBPOP(NZ,5) = (RTFUN(5,4)*GNDSTE(NZ)                                  2550
     1  +  RTFUN(5,7)*VIBPOP(NZ,2))/(EA(4) + EA(7))                          2560
```

```
      VIBPOP(NZ,6) = (RTFUN(6,8)*VIBPOP(NZ,8)                           2570
     1  + RTFUN(6,12)*GNDSTE(NZ))/(EA(8) + EA(11)                      2580
     2  + EA(12))                                                      2590
      VIBPOP(NZ,7) = (RTFUN(7,9)*VIBPOP(NZ,2)                           2600
     1 + RTFUN(7,13)*GNDSTE(NZ))/(EA(9) + EA(13))                      2610
      VIBPOP(NZ,8) = (RTFUN(8,10)*VIBPOP(NZ,2)                          2620
     1  + RTFUN(8,14)*GNDSTE(NZ))/(EA(10) + EA(14))                    2630
  235 CONTINUE                                                          2640
      GO TO 350                                                         2650
  240 CONTINUE                                                          2660
      DO 245 NZ = 1, NALT                                               2670
      VIBPOP(NZ,2) = ((TVSIG(NZ) + VVNSIG(NZ)                           2680
     1  + GUP(1)*SLM1(KB(1),TTUP(NZ,1))                                2690
     8  + (XFER(VIBPOP(1,2),TTH(1,1),NZ,1,1,KB(1))                     2700
     3  + XFER(VIBPOP(NZ,2),TTH(NZ,1),NALT,NZ,2,KB(1)))*EA(1)          2710
     4  + GDOWN(1)*SM1(KB(1),TTDOWN(NZ,1)))*GNDSTE(NZ)                 2720
     5  + (EA(3) + VTSIG(NZ))*VIBPOP(NZ,3) + AAA*CHEML(NZ))            2730
     5  /(EA(1) + VTSIG(NZ) + VVRSIG(NZ) + TVSIG(NZ))                  2740
      VIBPOP(NZ,3) = ((TVSIG(NZ)                                        2750
     1  + GUP(3)*SLM1(KB(3),TTUP(NZ,3))                                2760
     2  + GUP(2)*SLM1(KB(2),TTUP(NZ,2))                                2770
     3  + GDOWN(2)*SLM1(KB(2),TTDOWN(NZ,2))                            2780
     4  + GDOWN(3)*SLM1(KB(3),TTDOWN(NZ,3)))*VIBPOP(NZ,2)              2790
     5  + AAA*CHEML(NZ))/(EA(2) + EA(3) + TVSIG(NZ))                   2800
      GNDSTE(NZ) = CONC(NZ) - VIBPOP(NZ,2) - VIBPOP(NZ,3)               2810
  245 CONTINUE                                                          2820
      GO TO 350                                                         2830
  250 CONTINUE                                                          2840
      DO 255 NZ = 1, NALT                                               2850
      VIBPOP(NZ,6) = GNDSTE(NZ)*(VVNSIG(NZ)                             2860
     1  + GDOWN(3)*SM1(KB(3),TTDOWN(NZ,3))                             2870
     2  + GUP(3)*SLM1(KB(3),TTUP(NZ,3)))                               2880
```

```
     3  /(VVRSIG(NZ)*1.5 + EA(3))                                          2890
        VIBPOP(NZ,5) = (VIBPOP(NZ,6)*VVRSIG(NZ)*0.5                        2900
     1  + TVSIG(NZ)*VIBPOP(NZ,4))/(VTSIG(NZ) + 5.0*EA(1))                  2910
        VIBPOP(NZ,4) = (VIBPOP(NZ,5)*(VTSIG(NZ) + 5.0*EA(1))               2920
     1  + TVSIG(NZ)*VIBPOP(NZ,3))                                          2930
     2  /(TVSIG(NZ) + VTSIG(NZ) + 4.0*EA(1))                               2940
        VIBPOP(NZ,3) = (VIBPOP(NZ,4)*(VTSIG(NZ) + 4.0*EA(1))               2950
     1 + TVSIG(NZ)*VIBPOP(NZ,2))/(TVSIG(NZ) + VTSIG(NZ) + 3.0*EA(1))       2960
        VIBPOP(NZ,7) = (VVNSIG(NZ)*VIBPOP(NZ,3)                            2970
     1  + GNDSTE(NZ)*(GUP(2)*SLM1(KB(2),TTUP(NZ,2))                        2980
     2  + GDOWN(2)*SM1(KB(2),TTDOWN(NZ,2))))/EA(2)                         2990
        VIBPOP(NZ,2) = (GNDSTE(NZ)*(TVSIG(NZ)                              3000
     1  + GUP(1)*SLM1(KB(1),TTUP(NZ,1))                                    3010
     2  + GDOWN(1)*SM1(KB(1),TTDOWN(NZ,1)))                                3020
     3  + VIBPOP(NZ,3)*(3.0*EA(1) + VTSIG(NZ)))                            3030
     4  /(VTSIG(NZ) + TVSIG(NZ) + EA(1))                                   3040
255  CONTINUE                                                              3050
     GO TO 350                                                             3060
260  CONTINUE                                                              3070
     DO 265 NZ = 1, NALT                                                   3080
        VIBPOP(NZ,2) = (TVSIG(NZ) + VVNSIG(NZ)                             3090
     1  + GUP(1)*SLM1(KB(1),TTUP(NZ,1))                                    3100
     2  + (XFER(VIBPOP(1,2),TTH(1,1),NZ,1,1,KB(1))                         3110
     3  + XFER(VIBPOP(NZ,2),TTH(NZ,1),NALT,NZ,2,KB(1)))*EA(1)              3120
     4  + GDOWN(1)*SM1(KB(1),TTDOWN(NZ,1)))*GNDSTE(NZ)                     3130
     5  /(VTSIG(NZ) + EA(1) + VVRSIG(NZ))                                  3140
265  CONTINUE                                                              3150
     GO TO 350                                                             3160
350  CONTINUE                                                              3170
     DO 360 NB = 1, NBAND                                                  3180
        FAC = POWER(NB)*EA(NB)                                             3190
        NBX = (NB - 1)*25 + 1                                              3200
```

```
      NBC = LBC(NBX)                                                    3210
      NI = MOD(NBC/100,100)                                             3220
      NF = MOD(NBC,100)                                                 3230
      KLN = KL(NB)                                                      3240
      KBN = KB(NB)                                                      3250
      BNDL = BNDLAM(NB)                                                 3260
      JF = (NF - 1)*25 + 1                                              3270
      SNB = STR(NBX)*EXP(1.43879*WAVE(JF)/300.))                        3280
     DO 360 NZ = 1, NALT                                                3290
      BNDRAD(NZ,NB) = VIBPOP(NZ,NI)*FAC                                 3300
      BNDTAU(NZ,NB) = TAUMAX(KLN,KBN,PM,BV,TEMP(NZ),BNDL,SNB,           3310
     1  VIBPOP(NZ,NF))                                                  3320
 360  CONTINUE                                                          3330
     DO 850 NX = 1, NALT                                                3340
 850  CONTINUE                                                          3350
 400  CONTINUE                                                          3360
      IF (ITER .LT. ITMAX)  GO TO 181                                   3370
 900  CONTINUE                                                          3380
      IF (NLTE .LE. 2)  GO TO 920                                       3390
     DO 910 NZ = 1, NALT                                                3400
      WRITE (NP,8) (VIBPOP(NZ,NL), NL = 1, NLEVEL)                      3410
 910  CONTINUE                                                          3420
 920  CONTINUE                                                          3430
      RETURN                                                            3440
   8  FORMAT (1P8E10.4)                                                 3450
  15  FORMAT (1X)                                                       3460
  16  FORMAT (3X,1P10E12.4)                                             3470
  26  FORMAT (6X,7I8)                                                   3480
  36  FORMAT (1H1)                                                      3490
      END                                                               3500
```

Function XFER

Purpose: Function XFER is used to compute the excitation of a vibrational level caused by molecules radiating in a given band.

Method: Function XFER computes the contribution to the source function from the radiation received in a given vibration-rotation band by a molecule at a given altitude. A single entry to Function XFER computes this number for radiation received from molecules located either below or above this altitude. Equation (2-60) is summed over each altitude interval involved.

Program Flow: Function XFER assumes that the values in RAD(1) and TTH(1) are the values corresponding to the lowest altitude over which the integration is to be performed. Thus when called from Subroutine BANRAD to determine the effect of molecules above a given altitude with index NZ, the calling argument list has the form (VIBPOP(NZ, 2), TTH(NZ, 1), NALT, NZ, 2, KB(1)) for a band transition between vibrational levels 1 and 2 and band number 1.

Upon entry to Function XFER, the number of altitude intervals is found from the difference of NTOP and NBOT, and stored in NINT. If NTOP = NBOT, the value 0.0 is returned by Function XFER.

Otherwise, K is tested and control transferred to the appropriate part of the function.

When K is one, XFER computes the radiative contribution to excitation from layers below the altitude level for which the computation is made. When K is two, the contribution from layers above is computed.

The computation from a single level is found in each passage through the loop ending with statement 120 (if K is one) or statement 220 (if K is two). Before entry to either loop, RA is set to one, TB to zero, and EA3 to zero. RFAC, the ratio of ground state to upper state vibrational populations is computed and SUM is set to one-half. Within the appropriate loop, TB is incremented by the optical thickness obtained from array TTH. $1/2\ N_{12}(\tau)$ for the band is obtained from function SLM1 and the value stored in EB2. The band function $1/2L_3(\tau)$ is computed from the value of $L_o(\tau)$ obtained by a call to function SL3 and $1/2N_{12}(\tau)$ through use of the recurrence relation

$$\frac{1}{2} L_3(\tau) = \frac{1}{4} \left[L_o(\tau) + \tau N_{12}(\tau) \right]$$

Equation (2-60) is then applied and the result added to the value in SUM.

Mnemonic Variables:

EA3	Temporary storage for value of the band function L_3 for the nearer of two adjacent integration points
EB2	Temporary storage for value of the band function $1/2\ N_{12}$ for the farther away of two adjacent integration points
EB3	Temporary storage for value of the band function L_3 for the farther away of two adjacent integration points
K	integer parameter used to choose appropriate section of this function
KB	integer parameter used to choose appropriate sections of functions SLM1 and SL3
NBOT NTOP	indices in altitude array of top and bottom altitudes in layer of interest
NF	index used to select values from array VIBPOP
NINT	(NTOP -- NBOT), number of intervals
NPL	(NINT + 1)
NY	index for altitude level
NZ	index for altitude level
RA	relative vibrational population of nearer altitude point
RAD	real array containing vibrational population of an upper state
RB	relative vibrational population of farther altitude point
RFAC	ratio of ground state population to upper vibrational level population
SUM	immediate result of integration

TB	optical distance from level for which calculation is being made to farther of two levels
TM	increment of optical thickness between two adjacent levels
TTH	real array storing incremental optical thicknesses
VIBPOP	real array storing vibrational populations of ground and excited levels of a molecule

Non-Library Externals: SLM1, SL3

Called by: BANRAD

```
      FUNCTION XFER (RAD,TTH,NTOP,NBOT,K,KB)                          10
      COMMON /POPCOM/ VIBPOP(201,25)                                  20
      DIMENSION RAD(201),TTH(201),HOLD(201)                           30
      NINT = NTOP - NBOT                                              40
      NN = NINT/2                                                     50
      KK = MOD(NINT,2)                                                60
      NPL = NINT + 1                                                  70
      IF (NINT .GT. 0)  GO TO 100                                     80
      XFER = 0.0                                                      90
      RETURN                                                         100
  100 CONTINUE                                                       110
      RA = 1.0                                                       120
      TA = 0.0                                                       130
      TB = 0.0                                                       140
      EA3 = 0.0                                                      150
      IF (K .GT. 1)  GO TO 200                                       160
      NF = NTOP                                                      170
      RFAC = VIBPOP(NF,1)/RAD(NF)                                    180
      SUM = 0.5*RA                                                   190
     DO 120 NZ = 1, NINT                                             200
      NY = NINT - NZ + 1                                             210
      TM = TTH(NY)                                                   220
      TB = TB + TM                                                   230
      EB2 = SLM1(KB,TB)                                              240
      EB3 = (SL3(KB,TB) + 2.0*TB*EB2)/4.0                            250
      NF = NF - 1                                                    260
      RB = RFAC*RAD(NY)/VIBPOP(NF,1)                                 270
      SUM = SUM + (RB - RA)*(EB3 - EA3)/TM                           280
      RA = RB                                                        290
      EA3 = EB3                                                      300
  120 CONTINUE                                                       310
      XFER = ABS(SUM - RB*EB2)/RFAC                                  320
```

```
      RETURN                                                            330
200   CONTINUE                                                          340
      NF = NBOT                                                         350
      RFAC = VIBPOP(NF,1)/RAD(1)                                        360
      SUM = 0.5*RA                                                      370
      DO 220 NZ = 2, NPL                                                380
      TM = TTH(NZ - 1)                                                  390
      TB = TR + TM                                                      400
      EB2 = SLM1(KB,TB)                                                 410
      EB3 = (SL3(KB,TB) + 2.0*TB*EB2)/4.0                               420
      NF = NF + 1                                                       430
      RB = RFAC*RAD(NZ)/VIBPOP(NF,1)                                    440
      SUM = SUM + (RB - RA)*(EB3 - EA3)/TM                              450
      RA = RB                                                           460
      EA3 = EB3                                                         470
220   CONTINUE                                                          480
      XFER = ABS(SUM - RB*EB2)/RFAC                                     490
      RETURN                                                            500
      END                                                               510
```

Function SL3

Purpose: This function yields the total absorption of radiation from a continuum source passing through a line absorbing medium. The result of this computation is used by function XFER to compute the band function $L_3^b(\tau)$.

Method: SL3 computes the function $L^b(\tau)$ for parallel and perpendicular transitions in linear molecules. The algorithm uses a polynomial approximation to values of $L^b(\tau)$ computed for a reduced rotational temperature $hcB_v/kT = 0.002$. The values computed by SL3 are within five percent of the correct values for any reduced rotational temperature encountered in running the program.

Program Flow: Upon entry, the absolute value of the argument TAU is defined as T. Some logic then is used to choose the appropriate polynomial curve fit to the function which is computed and return is executed.

Inputs: K, TAU

Output: SLM1

Mnemonic Variables:

K	input parameter used to choose appropriate branch of the function, K equals 1 for parallel band, K equals 2 for perpendicular band.
SL3	computed value of $L^b(\tau)$.

T	absolute value of argument of TAU
TAU	input variable, optical depth k_t.
X	argument used to compute curve fit
Y	natural logarithm of T.

Non-Library Externals: None

Called by: XFER

```
      FUNCTION SL3 (K,TAU)                                                  10
     T = ABS (TAU)                                                          20
      IF (K .EQ. 2)  GO TO 200                                              30
      IF (T .GT. 4.0)  GO TO 110                                            40
      X = (T - 2.0)*0.5                                                     50
      SL3 = ((((((0.0004110*X - 0.0018496)*X + 0.0058378)*X                 60
     1  - 0.0246060)*X + 0.0792553)*X - 0.2301023)*X + 0.6569377)*T         70
      RETURN                                                                80
  110 CONTINUE                                                              90
      IF (T .GT. 10.0)  GO TO 120                                           100
      X = (T - 7.0)/3.0                                                     110
      SL3 = ((((((0.0001916*X - 0.0007254)*X + 0.0023078)*X                 120
     1  - 0.0080348)*X + 0.0274865)*X - 0.0942082)*X + 0.3539295)*T         130
      RETURN                                                                140
  120 CONTINUE                                                              150
      Y = ALOG(T)                                                           160
      IF (T .GT. 100.0)  GO TO 130                                          170
      X = 9.210340371976/Y - 3.0                                            180
      SL3 = ((((((0.0000869*X - 0.0001530)*X - 0.0001490)*X                 190
     1  + 0.0013943)*X - 0.0063060)*X + 0.0837063)*X + 1.1415564)*Y         200
      RETURN                                                                210
  130 CONTINUE                                                              220
      X = 9.210340371976/Y - 1.0                                            230
      SL3 = (((((((((0.007142 - 0.006222*X) + 0.005211)*X                   240
     1  - 0.004896)*X - 0.005477)*X + 0.007743)*X - 0.015900)*X             250
     2  + 0.122528)*X + 0.940111)*Y                                         260
      RETURN                                                                270
  200 CONTINUE                                                              280
      IF (T .GT. 4.0)  GO TO 210                                            290
      X = (T - 2.0)*0.5                                                     300
      SL3 = ((((((0.0002431*X - 0.0010991)*X + 0.0041830)*X                 310
     1  - 0.0160764)*X + 0.0581769)*X - 0.2013270)*X + 0.7188944)*T         320
```

```
      RETURN                                                            330
  210 CONTINUE                                                          340
      IF (T .GT. 10.0)  GO TO 220                                       350
      X = (T - 7.0)/3.0                                                 360
      SL3 = ((((((0.0001163*X - 0.0004699)*X + 0.0015637)*X             370
     1  - 0.0065345)*X + 0.0255562)*X - 0.1003592)*X + 0.4282944)*T     380
      RETURN                                                            390
  220 CONTINUE                                                          400
      Y = ALOG(T)                                                       410
      X = 4.605170185988O9/Y - 1.0                                      420
      SL3 = ((((((((((0.035431 - 0.035086*X)*X + 0.052211)*X            430
     1  - 0.056895)*X - 0.046818)*X + 0.045072)*X + 0.009249)*X         440
     2  - 0.020887)*X - 0.062059)*X + 0.163319)*X + 1.399769)*Y         450
      RETURN                                                            460
      END                                                               470
```

Function TAUMAX

Purpose: The purpose of this function is to compute the optical thickness of the strongest line in an infrared vibrational - rotational band.

Method: Function TAUMAX computes the optical thickness k_o at the center of the strongest line of a vibrational-rotational band of a linear molecule, assuming Doppler line shape.

Program Flow: The optical thickness at the center of a line with Doppler shape is

$$k_o = 5.3612 \times 10^{-18} \sqrt{\frac{M}{T}} \; N_{lc} \lambda_o S_{lu}^{line}$$

where M is the molecular mass (gm/mole), T is the absolute temperature (k), N_{lc} is the columnar number density (molecules/cm^{-2}), λ_o is the line center wavelength (μm) and S_{lu}^{line} is the line strength in absorption (cm^{-1} $(atm\text{-}cm)^{-1}$) at 300K. Function TAUMAX computes the reduced rotational temperature hcB_v/kT and transmits this as an argument to Function SJMAX which computes the fraction of total band strength in the strongest line $S_{J_{max}} / S_{lu}^{band}$. TAUMAX then computes the optical thickness at the center of the strongest line, using $S_{J_{max}}$ in place of S_{lu}^{line} in the above expression.

Inputs: KL, KB, RM, BV, T, FLAM, S, DEN

Output: TAUMAX

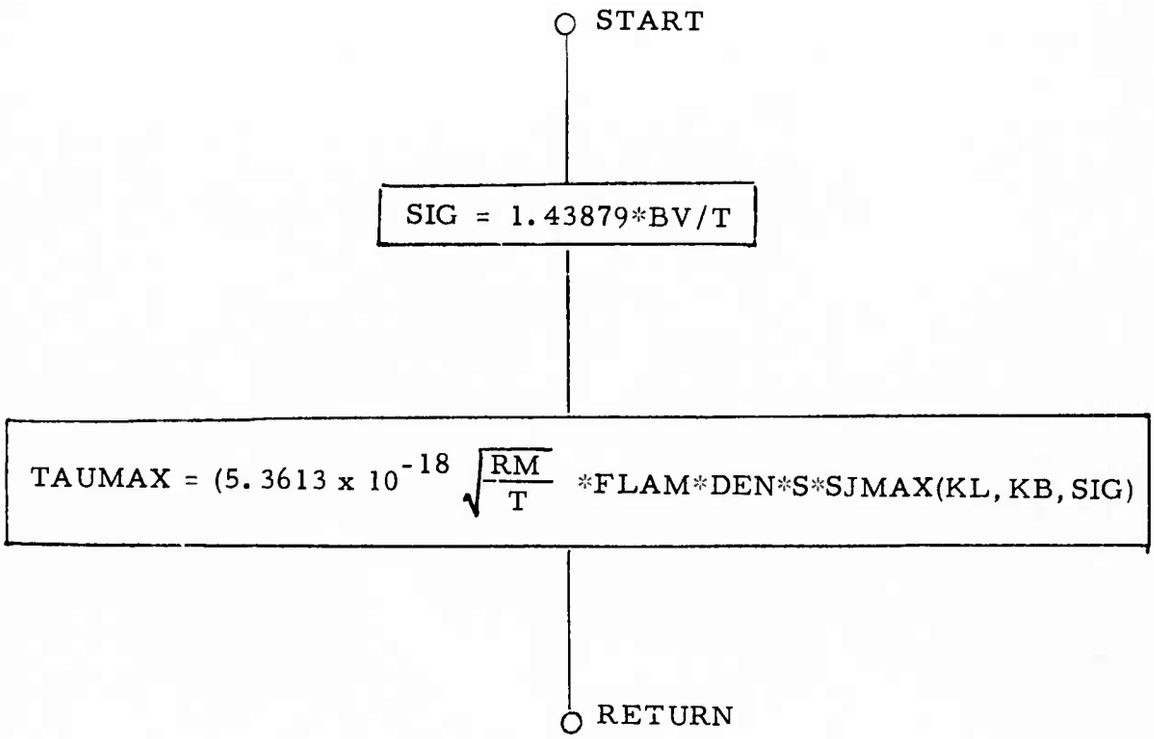
Mnemonic Variables:

BV	rotational constant (cm^{-1})
DEN	columnar number density of molecule (cm^{-2})
FLAM	wavelength (μm)
KB	integer parameter
KL	integer parameter
RM	mass of molecule (gm/mole)
S	band strength ($cm^{-1}(atm\text{-}cm)^{-1}$)
SIG	the quantity hcB_v/kT
T	temperature (K)
TAUMAX	value of optical thickness returned by function.

Non-Library Externals: SJMAX

Called By: BANRAD

FLOW CHART: TAUMAX



```
      FUNCTION TAUMAX (KL,KB,RM,BV,T,FLAM,S,DEN)                        10
      SIG = 1.43879*BV/T                                                20
      TAUMAX = 5.3613E-18*SQRT(RM/T)*FLAM*DEN*S*SJMAX(KL,KB,SIG)        30
      RETURN                                                            40
      END                                                               50
```

Function SJMAX

Purpose: The purpose of this function is to compute relative line strengths.

Method: Function SJMAX computes the relative strength of the strongest line in a parallel or perpendicular band of a linear molecule, assuming that all parallel bands are $^1\Sigma - {}^1\Sigma$ transitions and all perpendicular bands are $^1\Pi - {}^1\Sigma$ transitions. The effect of neglecting the actual transition involved is negligible for the uses of Function SJMAX.

Program Flow: Function SJMAX computes the J values of the strongest line in the band, computes the strength from the appropriate Hönl-London formula, and divides this by the rotational partition function returned by Function QJPART.

Inputs: KL, KB, SIG

Output: SJMAX

Mnemonic Variables:

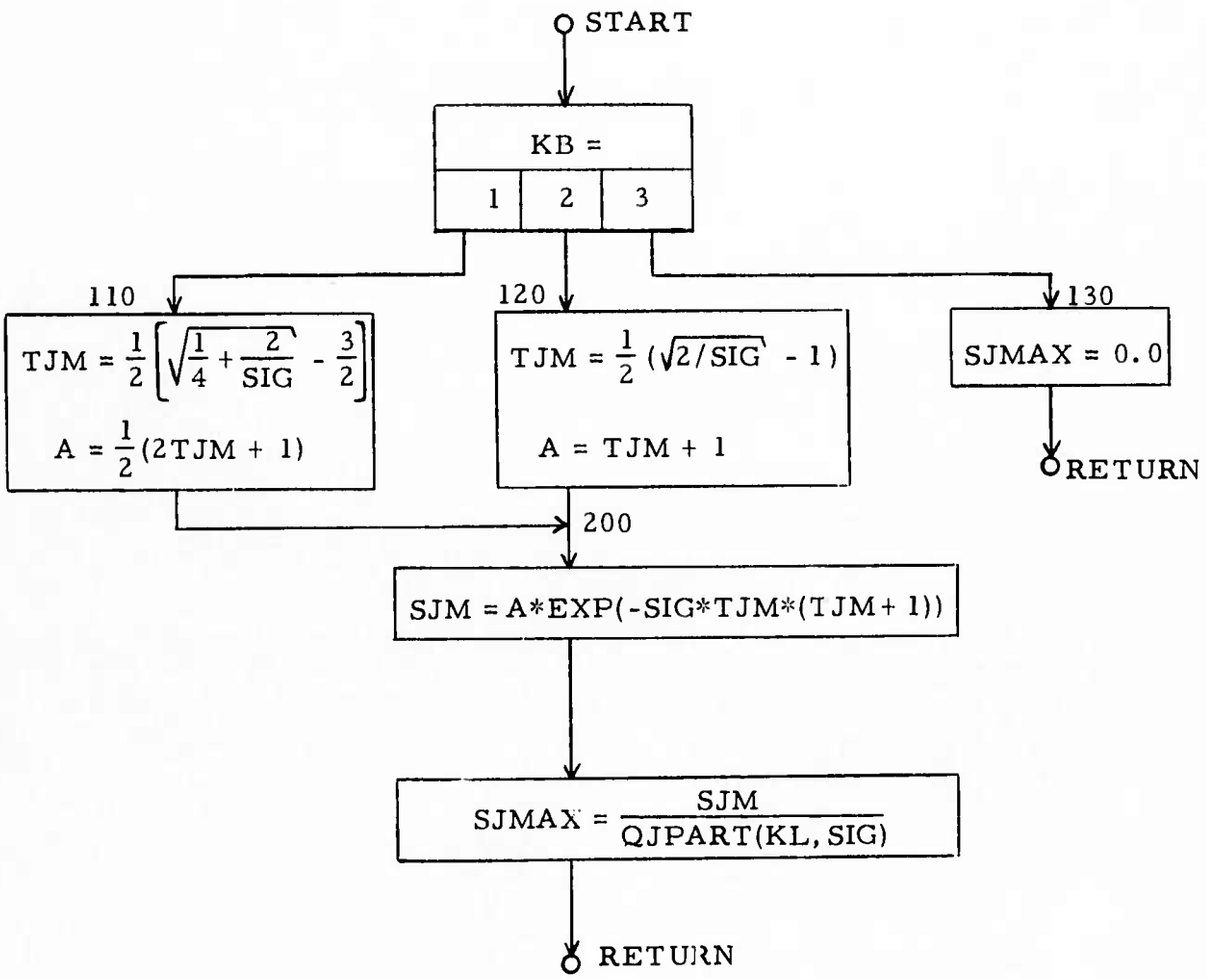
A	intermediate result in calculation
KB	parameter to choose type of transition (KB = $1\Sigma - \Sigma$ transition) (KB = $2\Pi - \Sigma$ transition)
KL	parameter passed to QJPART

SIG	the quantity $(hcB_v)/kT)$
SJM	strength of strongest line in band
SJMAX	relative strength of strongest line in band
TJM	effective rotational quantum number

Non-Library Externals: QJPART

Called By: TAUMAX

FLOW CHART: SJMAX



```
      FUNCTION SJMAX (KL,KB,SIG)                                        10
C       SJMAX COMPUTES THE RELATIVE STRENGTH OF THE STRONGEST LINE      20
C       IF KB = 1, SIGMA-SIGMA TRANSITION                               30
C       IF KB = 2, SIGMA-PI TRANSITION                                  40
      GO TO (110,120,130), KB                                           50
  110 CONTINUE                                                          60
      TJM = 0.5*(SQRT(0.25 + 2.0/SIG) - 1.5)                            70
      A = 0.5*(TJM + TJM + 1.0)                                         80
      GO TO 200                                                         90
  120 CONTINUE                                                         100
      TJM = 0.5*(SQRT(2.0/SIG) - 1.0)                                  110
      A = TJM + 1.0                                                    120
      GO TO 200                                                        130
  130 CONTINUE                                                         140
      SJMAX = 0.0                                                      150
      RETURN                                                           160
  200 CONTINUE                                                         170
      SJM = A*EXP(-SIG*TJM*(TJM + 1.0))                                180
      SJMAX = SJM/QJPART(KL,SIG)                                       190
      RETURN                                                           200
      END                                                              210
```

Function QJPART

Purpose: The purpose of this function is to compute rotational partition functions for linear molecules. Refer to Subroutine BANRAD for definition of symbols.

Method: Function QJPART computes the rotational partition function of a single vibrational level of a rigid linear molecule, assuming that the energy E_J of a rotational state with quantum number J is

$$E_J = hcB_v J(J+1)$$

Provision is made for computing the rotational partition functions for vibrational levels in which rotational states with only even or odd J values are present.

Program Flow: If K = 1 all rotational levels are assumed populated. If K = 2, only even rotational levels are assumed populated. If K = 3, only odd rotational level are assumed populated.

The computation uses polynomial approximates to the partition function

$$Q_J = \Sigma_J (2J+1) \exp(-E_J/kT)$$

with SIG as the parameter. The range of applicability is limited to values of SIG less than 0.2, and the largest value of SIG encountered for the molecules presently included is 0.015.

Inputs: K, SIG

Outputs: QJPART

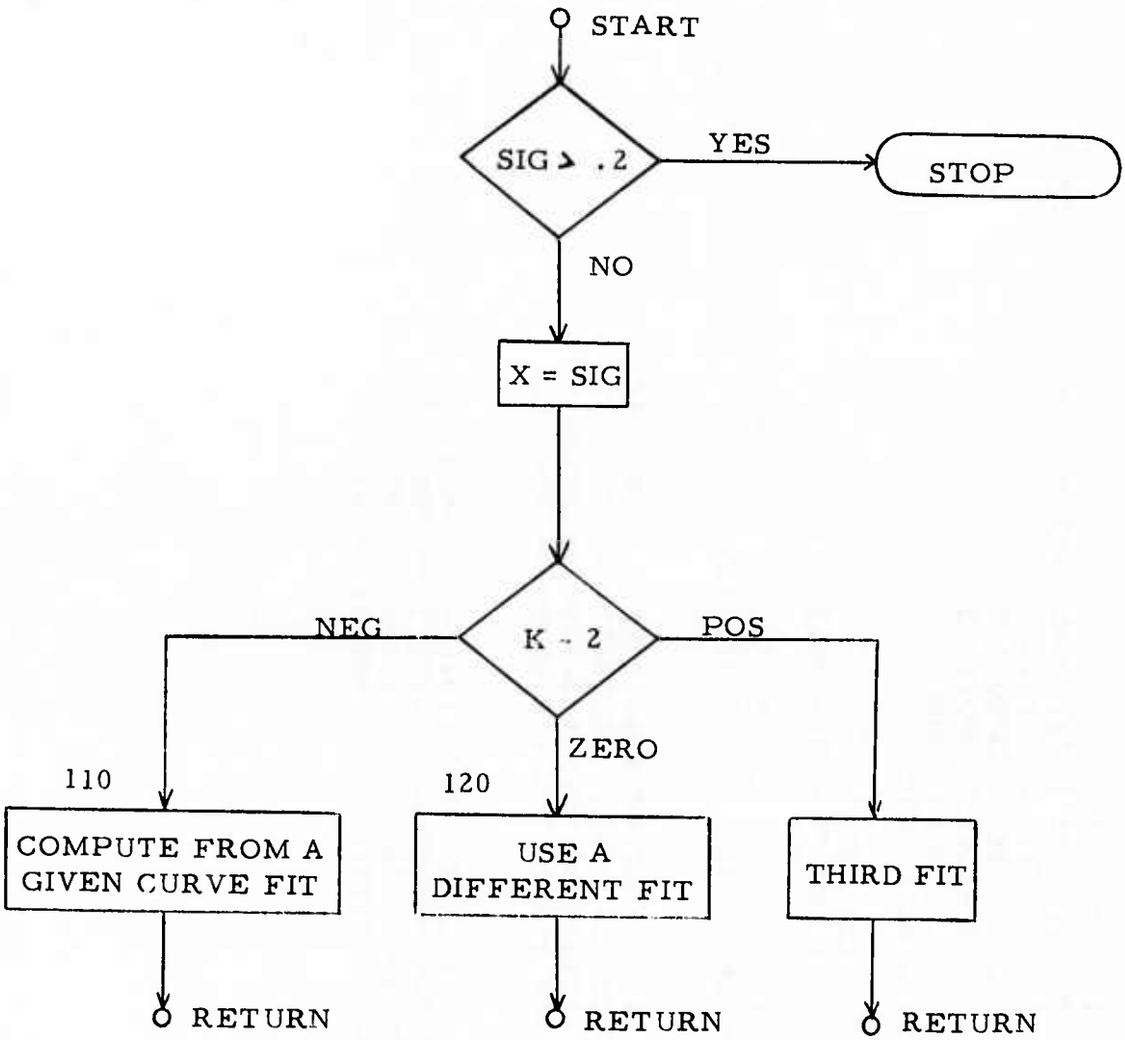
Menemonic Variables:

K	parameter used to choose appropriate fit
QJPART	value of rotational partition function
SIG	the quantity (hcB_v/kT)
X	the value of SIG, defined for convenience in the curve fit

Non-Library Externals: None

Called By: SJMAX

FLOW CHART: QJPART



```
      FUNCTION QJPART (K,SIG)                                                  10
C      QJPART COMPUTES THE ROTATIONAL PARTITION FUNCTION FOR A                 20
C      RIGID ROTATOR, SIG .LE. 0.2.                                            30
C        K = 1, ALL  LEVELS POPULATED                                          40
C        K = 2, EVEN LEVELS POPULATED                                          50
C        K = 3, ODD  LEVELS POPULATED                                          60
      IF (SIG .GT. 0.2)  STOP                                                  70
      X = SIG                                                                  80
      IF (K - 2)  110,120,130                                                  90
C      COEFFICIENTS ARE FROM RUN DALTE-EV                                     100
  110 CONTINUE                                                                 110
      QJPART = ((((0.0038650023285?*SIG + 0.012571833547?)*X                   120
     1  + 0.0666756164530)*X + 0.333333133379)*X + 1.00000000017)/SIG          130
      RETURN                                                                   140
  120 CONTINUE                                                                 150
      QJPART = (((((((62.5879214103*SIG - 29.5697177693)*SIG                   160
     1  + 5.49692670096)*SIG - 0.502572235608)*SIG                             170
     2  + 0.0297378193546)*SIG + 0.0328350909505)*SIG                          180
     3  + 0.166670220770)*SIG + 0.499999999456)/SIG                            190
      RETURN                                                                   200
  130 CONTINUE                                                                 210
      QJPART = (((((((29.5703117876 - 62.5874596502*SIG)*SIG                   220
     1  - 5.49579373669)*SIG + 0.505750942701)*SIG                             230
     2  - 0.0170397551676)*SIG + 0.0338315889063)*SIG                          240
     3  + 0.166663112387)*SIG + .500000000544)/SIG                             250
      RETURN                                                                   260
      END                                                                      270
```

Function SIMP

Purpose: The purpose of this function is to compute an integral using Simpson's rule.

Method: Function SIMP computes the integral of a function between any two of three equally spaced points. Simpson's rule integration is used.

Program Flow: Upon entry, the value of K is tested to choose the appropriate branch of the function with the following results:

$K < 2$ SIMP = (h/12) (8A (2) + 5A (1) - A (3)

$K = 2$ SIMP = (h/12) (8A (2) + 5A (3) - A (1)

$K > 2$ SIMP = (h/3) (4A (2) + A (1) + A (3))

Inputs: A, H, K

Output: SIMP

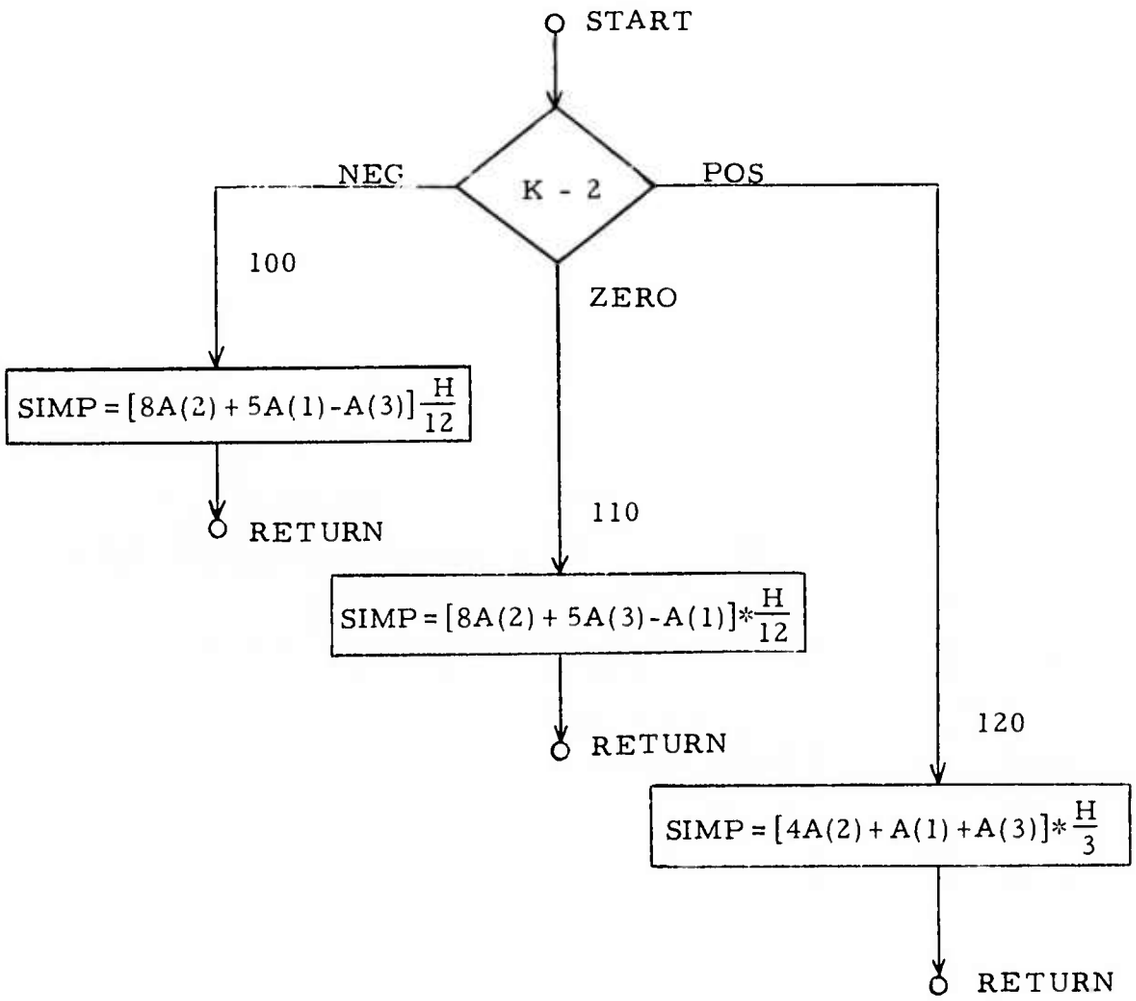
Mnemonic Variables:

A	array of values to be integrated
H	integration step size
K	parameter to choose method desired

Non-Library Externals: None

Called By: BANRAD, XFER

FLOW CHART: SIMP



```
      FUNCTION SIMP (A,H,K)                                          10
      DIMENSION A(3)                                                 20
      IF (K - 2)  100,110,120                                        30
  100 CONTINUE                                                       40
      SIMP = (8.0*A(2) +5.0*A(1) - A(3))*H/12.0                      50
      RETURN                                                         60
  110 CONTINUE                                                       70
      SIMP = (8.0*A(2) + 5.0*A(3) - A(1))*H/12.0                     80
      RETURN                                                         90
  120 CONTINUE                                                      100
      SIMP = (4.0*A(2) + A(1) + A(3))*H/3.0                         110
      RETURN                                                        120
      END                                                           130
```

Function SM1

Purpose: This function yields a factor needed to compute the true rate of absorption at an optical depth, k_o, of radiation (from a spectrally constant source) in a resonant medium. Refer to Subroutine BANRAD for further explanation and definition of symbols used below.

Method: SM1 computes the function $M_1^b(k_o)$ for parallel and perpendicular transitions in linear molecules. The algorithm employs a polynomial approximation to values of $M_1^b(k_o)$ computed at a reduced rotational temperature $hcB_v/kT = 0.002$.

Program Flow: Upon entry the absolute value of the argument TAU is defined as T. Some logic then is used to choose the appropriate polynomial curve fit to the function which is computed and return is executed.

Inputs: K, TAU

Output: SM1

Mnemonic Variables:

K	input parameter used to choose appropriate branch of the function; K equals 1 for parallel band, K equals 2 for perpendicular band
SM1	rate of absorption at optical thickness k_o, relative to zero optical thickness
T	absolute value of argument TAU
TAU	input variable - optical depth k_o

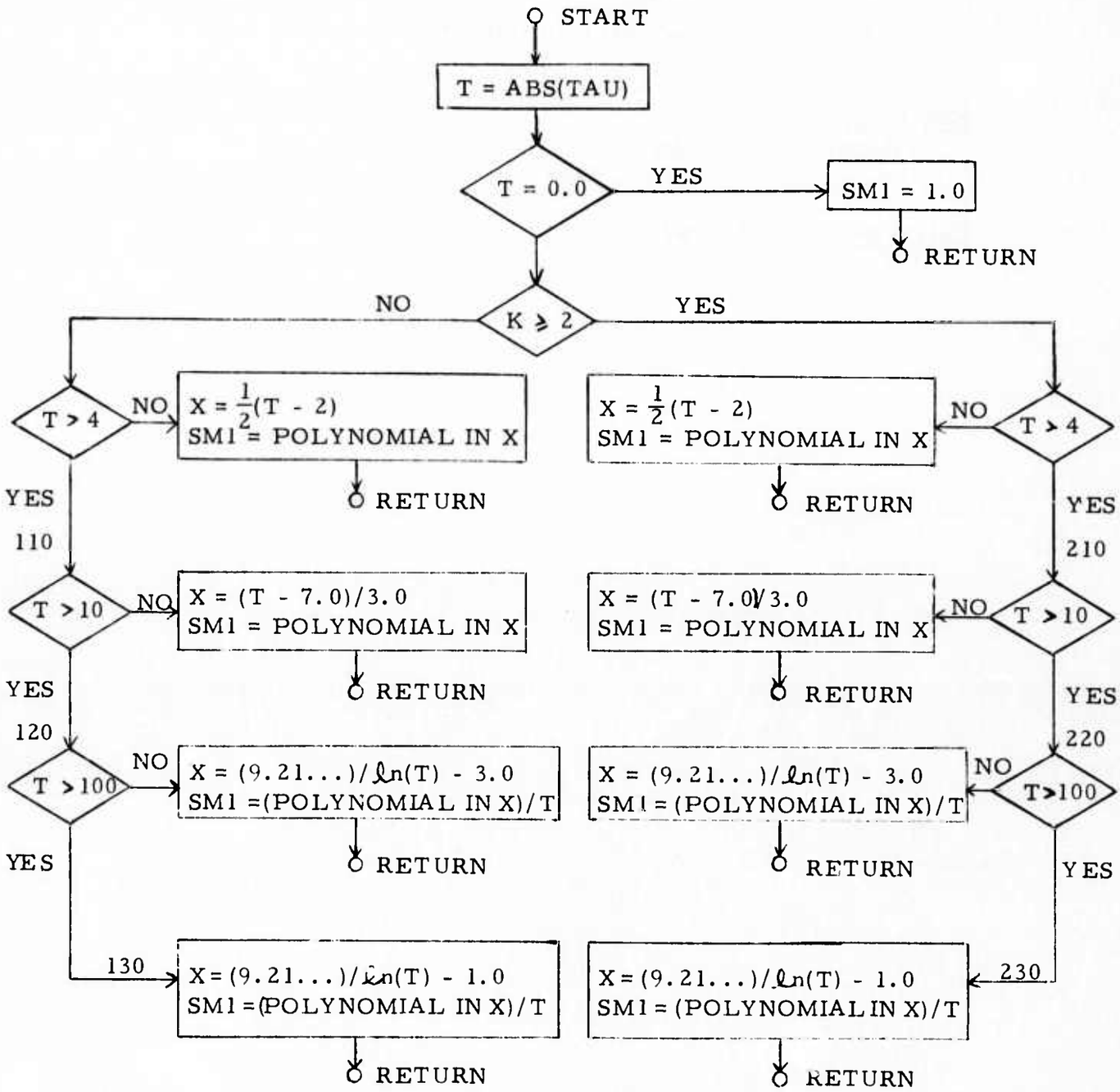
Mnemonic Variables (Cont'd)

X	argument used to compute curve fit
Y	natural logarithm of T

Non-Library Externals: None

Called By: BANRAD

FLOW CHART: SM1



```
      FUNCTION SM1 (K,TAU)                                                  10
      T = ABS(TAU)                                                          20
      IF (T .EQ. 0.0)  GO TO 300                                            30
      IF (K .GE. 2)  GO TO 200                                              40
      IF (T .GT. 4.0)  GO TO 110                                            50
      X = (T - 2.0)*0.5                                                     60
      SM1 = (((((2.35490E-03*X - 8.90535E-03)*X + 2.56223E-02)*X            70
     1  - 7.07877E-02)*X + 1.63735E-01)*X - 3.01749E-01)*X + 0.426845       80
      RETURN                                                                90
  110 CONTINUE                                                             100
      IF (T .GT. 10.0)  GO TO 120                                          110
      X = (T - 7.0)/3.0                                                    120
      SM1 = (((((5.92297E-04*X - 1.79043E-03)*X + 4.05109E-03)*X           130
     1  - 1.04749E-02)*X + 2.59109E-02)*X - 6.01708E-02)*X + 0.134123      140
      RETURN                                                               150
  120 CONTINUE                                                             160
      IF (T .GT. 100.0)  GO TO 130                                         170
      Y = ALOG(T)                                                          180
      X = 9.210340371976/Y - 3.0                                           190
      SM1 = ((((((-6.24516E-04*X - 6.79378E-04)*X + 2.98590E-03)*X         200
     1  -1.28190E-03)*X - 6.31784E-03)*X + 3.78904E-02)*X                  210
     2  + 8.30441E-01)/T                                                   220
      RETURN                                                               230
  130 CONTINUE                                                             240
      Y = ALOG(T)                                                          250
      X = 9.210340371976/Y - 1.0                                           260
      SM1 = ((((((6.12038E-04*X - 2.93434E-04)*X - 4.36163E-04)*X          270
     1  + 1.70037E-02)*X - 3.80088E-04)*X + 3.23441E-02)*X                 280
     2  + 8.17407E-01)/T                                                   290
      RETURN                                                               300
  200 CONTINUE                                                             310
      IF (T .GT. 4.0)  GO TO 210                                           320
```

```
      X = (T - 2.0)*0.5                                                      330
      SM1 = (((((1.39071E-03 *X - 5.29904E-03)*X + 1.58249E-02)*X            340
     1 - 4.7+039E-02)*X + 1.26175E-01)*X - 2.86333E-01)*X + 0.517573         350
      RETURN                                                                 360
  210 CONTINUE                                                               370
      IF (T .GT. 10.0)  GO TO 220                                            380
      X = (T - 7.0)/3.0                                                      390
      SM1 = (((((3.76420E-04*X - 1.25384E-03)*X + 3.41168E-03)*X             400
     1  - 1.'5360E-02)*X + 3.07476E-02)*X - 8.14705E-02)*X + 0.1941308       410
      RETURN                                                                 420
  220 CONTINUE                                                               430
      IF (T .GT. 100.0)  GO TO 230                                           440
      Y = ALOG(T)                                                            450
      X = 9.210340371976/Y - 3.0                                             460
      SM1 = (((-7.53415E-03*X - 8.87124E-03)*X + 6.52199E-02)*X              470
     1  + 1.304245)/T                                                        480
      RETURN                                                                 490
  230 CONTINUE                                                               500
      Y = ALOG(T)                                                            510
      X = 9.210340371976/Y - 1.0                                             520
      SM1 =(((((1.99067E-03 - 5.82519E-04*X)*X + 1.99569E-03)*X              530
     1  + 1.2 174E-03)*X + 4.76165E-02)*X + 1.18445)/T                       540
      RETURN                                                                 550
  300 CONTINUE                                                               560
      SM1 = 1.0                                                              570
      RETURN                                                                 580
      END                                                                    590
```

Function SLM1

Purpose: This function yields the factor used to obtain the actual absorption of radiation from an extended black body surface passing through a resonant medium. Refer to Subroutine BANRAD for further explanation and definition of symbols used below.

Method: SLM1 computes the function $\frac{1}{2\pi} N_1^b(k_t)$ for parallel and perpendicular transitions in linear molecules. The algorithm uses a polynomial approximation to values of $\frac{1}{2\pi} N_1^b(k_t)$ computed for a reduced rotational temperature $hcB_v/kT = 0.002$. The values computed by SLM1 are within five percent of the correct values for any reduced rotational temperature encountered in running the program.

Program Flow: Upon entry the absolute value of the argument TAU is defined as T. Some logic then is used to choose the appropriate polynomial curve fit to the function which is computed and return is executed.

Inputs: K, TAU

Output: SLM1

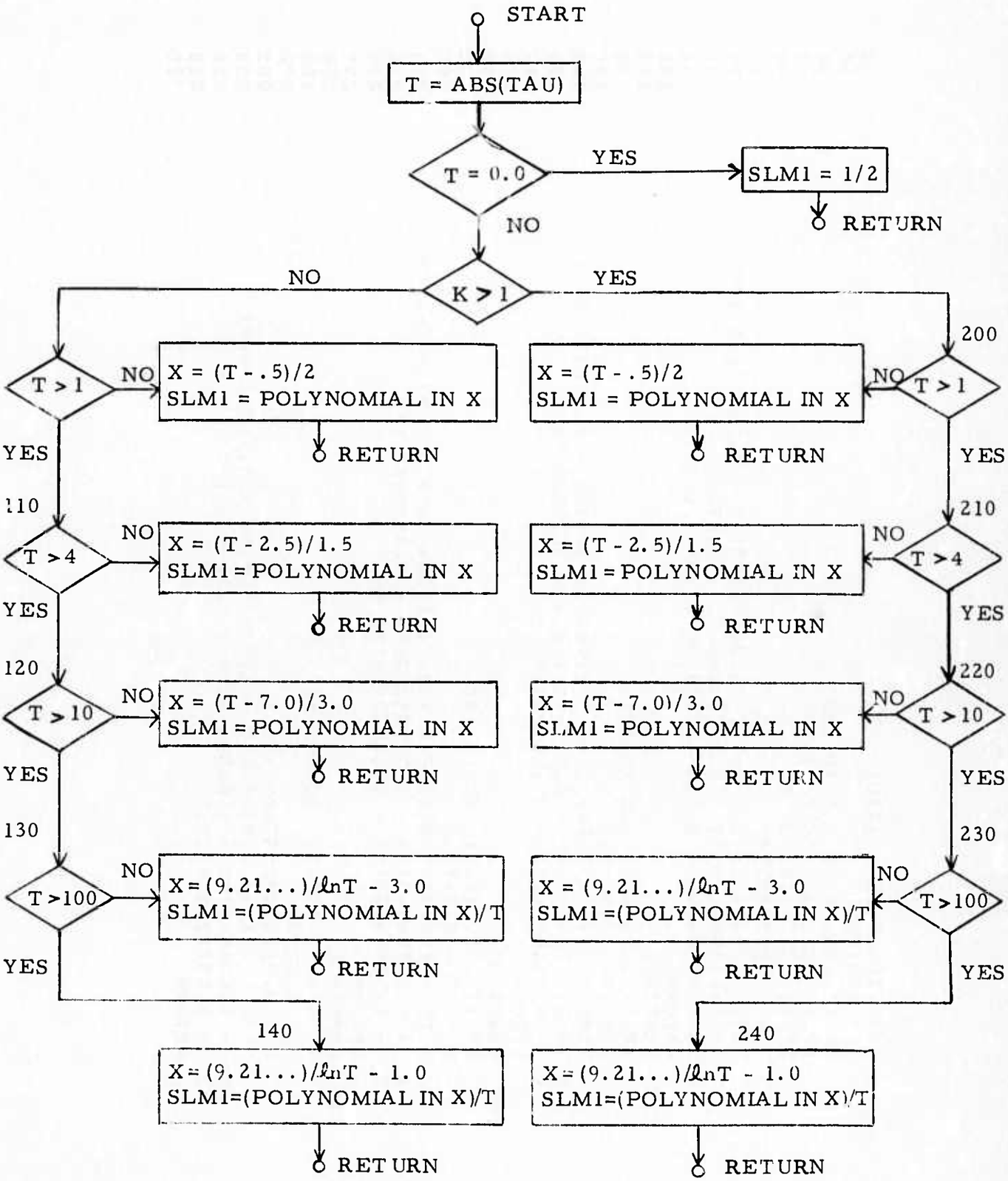
Mnemonic Variables:

K	input parameter used to chocse appropriate branch of the function, K equals 1 for parallel band, K equals 2 for perpendicular band
SLM1	computed value of $\frac{1}{2\pi} N_1^b(k_t)$
T	absolute value of argument TAU
TAU	input variable, optical depth k_t
X	argument used to compute curve fit
X	natural logarithm of T

Non-Library Externals: None

Called By: BANRAD

FLOW CHART: SLM1



```
      FUNCTION SLM1 (K,TAU)                                              10
      T = ABS(TAU)                                                       20
      IF (T .EQ. 0.0)  GO TO 300                                         30
      IF (K .GT. 1)  GO TO 200                                           40
      IF (T .GT. 1.0) GO TO 110                                          50
      T = 25.26760337*T                                                  60
      SLM1 = ((((((((5.54E-16*T - 1.3747E-13)*T + 3.11017E-11)*T         70
     1  - 6.434609E-9)*T + 1.24762E-6)*T - 2.57525E-4)*T                 80
     2  - 0.08264246 + 0.01984964*ALOG(T))*T + 1.0)*0.5                  90
      RETURN                                                            100
  110 CONTINUE                                                          110
      IF (T .GT. 4.0)  GO TO 120                                        120
      X = (T - 2.5)/1.5                                                 130
      SLM1 = ((((((((3.00591E-3 - 1.41466E-3*X)*X - 3.88085E-3)*X       140
     1  + 9.48736E-3)*X - 0.0239407)*X + 0.0523088)*X                    150
     2  - 0.1045984)*X + 0.1862743)*0.5                                  160
      RETURN                                                            170
  120 CONTINUE                                                          180
      IF (T .GT. 10.0)  GO TO 130                                       190
      X = (T - 7.0)/3.0                                                 200
      SLM1 = ((((((((5.08470E-4 - 1.90181E-4*X)*X - 3.27594E-4)*X       210
     1  + 2.24274E-3)*X - 5.54298E-3)*X + 0.01290804)*X                  220
     2  - 0.02939935)*X + 0.06552257)*0.5                                230
      RETURN                                                            240
  130 CONTINUE                                                          250
      IF (T.GT. 100.0)  GO TO 140                                       260
      Y = ALOG(T)                                                       270
      X = 9.210340371976/Y - 3.0                                        280
      SLM1 = ((((((  7.836E-4*X - 1.5181E-3)*X + 9.0128E-4)*X           290
     1  - 5.498E-4)*X - 1.0632E-3)*X + 0.0149364)*X                      300
     2  + 0.4378273)*0.5/T                                               310
      RETURN                                                            320
```

```
  140  CONTINUE                                                          330
       Y = ALOG(T)                                                       340
       X = 9.2103403719761/Y - 1.0                                       350
       SLM1 = ((((5.0746E-3 - 4.681E-4*X)*X - 1.5848E-3)*X               360
      1  + 0.0147526)*X + 0.4079074)*0.5/T                              370
       RETURN                                                            380
  200  CONTINUE                                                          390
       IF (T .GT. 1.0)  GO TO 210                                        400
       T = 26.0729185*T                                                  410
       SLM1 = (((((((2.71E-16*T - 6.8372E-14)*T + 1.58314E-11)*T         420
      1  - 3.39873E-9)*T + 7.01818E-7)*T - 1.61594E-4)*T                430
      2  - 0.0356453 + 0.01494384*ALOG(T))*T + 1.0)*0.5                 440
       RETURN                                                            450
  210  CONTINUE                                                          460
       IF (T .GT. 4.0)  GO TO 220                                        470
       X = (T - 2.5)/1.5                                                 480
       SLM1 = (((((((2.32E-3 - 1.1203E-3*X)*X - 3.0489E-3)*X             490
      1  + 7.9345E-3)*X - 0.0213281)*X + 0.0521586)*X                   500
      2  - 0.1213660)*X + 0.2508902)*0.5                                510
       RETURN                                                            520
  220  CONTINUE                                                          530
       IF (T .GT. 10.0)  GO TO 230                                       540
       X = (T - 7.0)/3.0                                                 550
       SLM1 = ((((((4.727E-4*X - 1.2459E-3)*X + 2.5233E-3)*X             560
      1  - 6.7323E-3)*X + 0.0174769)*X - 0.0421443)*X                   570
      2  + 0.0958415)*0.5                                               580
       RETURN                                                            590
  230  CONTINUE                                                          600
       IF (T .GT. 100.0)  GO TO 240                                      610
       Y = ALOG(T)                                                       620
       X = 9.2103403719761/Y - 3.0                                       630
       SLM1 = ((((9.1E-5*X - 1.9096E-3)*X - 1.8059E-3)*X                 640
```

```
     1  + 0.0263129)*X + 0.6392594)*0.5/T                                  650
        RETURN                                                             660
  240   CONTINUE                                                           670
        Y = ALOG(T)                                                        680
        X = 9.21034037976/Y - 1.0                                          690
        SLM1 = ((((6.444E-4*X + 3.453E-4)*X - 5.099E-4)*X                  700
     1  + 0.0216100)*X + 0.591052)*0.5/T                                   710
        RETURN                                                             720
  300   CONTINUE                                                           730
        SLM1 = 0.5                                                         740
        RETURN                                                             750
        END                                                                760
```

REFERENCES

Abels, L.L. and Shaw, J. H., J. Mol. Spectrosc. 20, 11-28 (1966).

Ackerman, M., Aeronomica Acta A, No. 69, Brussels (1970).

Ackerman, M., Biaume, F. and Nicolet, M., Can. J. Chem. 47, 1834-1840 (1969).

Anding, D., "Calculations of Atmospheric Spectral Radiance for Slant Paths from Several Altitudes", Report No. 8416-20-F(IV), Vol. IV, Willow Run Laboratories of the Institute of Science and Technology, The University of Michigan, Ann Arbor (1968).

Anderson, J.G., J. Geophys. Res. 76, 4634-4652 (1971a).

Anderson, J.G., J. Geophys. Res. 76, 7820-7824 (1971b).

Arnold, J.O., Whiting, E.E., and Lyle, G.C., J. Quant, Radiat. Transfer 9, 775-798 (1969).

Bainbridge, A.E., and Heidt, L.E., Tellus 18, 221-225 (1966).

Barth, C.A., J. Geophys. Res. 69, 3301-3303 (1964).

Barth, C.A., Planet. Space Sci. 14, 623 (1966a).

Barth, C.A., Ann. Geophys. 22, 198-207 (1966b).

Basco, N., Callear, A.B., and Norrish, R.G.W., Proc. Roy. Soc. A260, 459-474 (1961).

Basco, N., Callear, A.B., and Norrish, R.G.W., Proc. Roy. Soc. A269, 180-187 (1962).

Bates, D., and Hays, P., Planet. Space Sci. 15, 189-197 (1967).

Bates, D.R., and Moiseiwitsch, B.L., J. Atmos. Terr. Phys. 8, 305-308 (1956).

Bates, D.R., and Nicolet, M., Planet. Space Sci. 13, 905-909 (1965).

Bates, D.R., and Witherspoon, A.E., Mon. Not. Roy. Astr. Soc. 112, 101-124 (1952).

Bates, D.R., and Nicolet, M., J. Geophys. Res. 55, 301 (1950).

Bauer, E., Kummler, R. and Bortner, M.H., App. Optics 10, 1861-1869 (1971).

Bauer, S. H., and Tsang, S. C., Phys. Fluids 6, 182-189 (1963).

Baulch, D. L., Drysdale, D. D., and Lloyd, A. C., "Critical Evaluation of Rate Data for Homogeneous, Gas-Phase Reactions of Interest in High-Temperature Systems", High Temperature Reaction Rate Data, No. 1, The University, Leeds, England, May (1968a).

Baulch, D. L., Drysdale, D. D., and Lloyd, A. C., "Critical Evaluation of Rate Data for Homogeneous, Gas Phase Reactions of Interest in High-Temperature Systems", High Temperature Reaction Rate Data, No. 2, , The University, Leeds, England, November (1968b).

Baulch, D. L., Drysdale, D. D., Horne, D. G., and Lloyd, A. C., "Critical Evaluation of Rate Data for Homogeneous, Gas Phase Reactions of Interest in High-Temperature Systems", High Temperature Reaction Rate Data, No. 4, The University, Leeds, England, December (1968).

Baulch, D. L., Drysdale, D. D.,, and Horne, D. G., "Critical Evaluation of Rate Data for Homogeneous, Gas Phase Reactions of Interest in High Temperature Systems, High Temperature Reaction Rate Data, No. 5, The University, Leeds, England, July (1970).

Bérces, T., and Förgeteg, S., Trans. Far. Soc. 66, 633-639 (1970a).

Bérces, T., and Förgeteg, S., Trans. Far. Soc. 66, 640-647 (1970b).

Bérces, T., and Förgeteg. S., Trans. Far. Soc. 66, 648-655 (1970c).

Biondi, M. A., Can. J. Chem. 47, 1711-1719 (1969).

Black, G., and Eckstrom, D. J., "Excitation and De-excitation of Vibration in N_2 by Oxygen Atoms", Final Report Contract DAHCO4-70-C-0036 ARPA Order No. 1482, SRI, Menlo Park, Calif., October (1971).

Bowman, M. R., and Thomas, L., and Geisler, J. E., J. Atm. Terr. Phys. 32, 1661-1674 (1970).

Breig, E. L., J. Chem. Phys. 51, 4539-4537 (1969).

Brennan, W., and Niki, H., J. Chem. Phys. 42, 3725-3726 (1965).

Bulirsch, R., and Stoer, J., Num. Math. 8, 1-13 (1966).

Campbell, I. M., and Thrush, B. A., Trans. Far. Soc. 64, 1265-1274 (1968).

Ceschino, F., and Kuntzmann, J., Numerical Solution of Initial Value Problems, Prentice-Hall, Englewood Cliffs. N. J. (1966).

Chapman, S., and Cowling, T. G., The Mathematical Theory of Non-Uniform Gases, Cambridge University Press (1960).

Clark, T. C., Garnett, S. H. and Kistiakowski, G. B., J. Chem. Phys. 52, 4692-4698 (1970.

Clark, I. D., and Wayne, R. P., Proc. Roy. Soc. A316, 539-550 (1970).

Clyne, M. A. A., and Thrush, B. A., Proc. Roy. Soc. A269, 404-418 (1962).

Clyne, M. A. A., and Thrush, B. A., Proc. Roy. Soc. A275, 559 (1963).

Colegrove, F. D., Johnson, F. S., and Hanson, W. B., J. Geophys. Res. 71, 2227-2236 (1966).

Corbin, V. L., Dalgarno, A., Degges, T. C., House, F. B., Lilienfeld, P., Ohring, G., and Oppel, G. E., Atmospheric Radiance Models for Limb-Viewing Geometry in the Five- to Twenty-Five Micron Spectral Region, AFCRL-69-0552 (1969).

Dalgarno, A., Planet. Space Sci. 10, 19-28 (1963).

Dalgarno, A., Rev. Mod. Phys. 39, 858-861 (1967).

Dalgarno, A., Phil. Trans. Roy. Soc. A264, 153-160 (1969).

Dalgarno, A., Ann. Geophys. 26, 601-607 (1970).

Davis, P. J., and Rabinowitz, P., Numerical Integration, Blaisdell, Waltham, Mass. (1967).

Dalgarno, A., and McElroy, M. B., Planet. Space Sci. 14, 1321-1329 (1966).

Dean, A. M., and Kistiakowsky, J. Chem. Phys. 54, 1718-1725 (1971).

Degges, T. C., Applied Optics 10, 1856-1860 (1971).

Degges, T. C., A High Altitude Radiance Model, AFCRL 72-0273 (1972).

Detwiler, C. R., Garrett, D. L., Purcell, J. D., and Tousey, R., Ann. Geophys. 17, 263-272 (1961).

Donahue, T. M., J. Geophys. Res. 71, 2237-2242 (1966).

Donovan, R. J., and Hussain, D., Chem. Rev. 70, 489-516 (1970).

Dunkin, D. B., Feasenfeld, F. C., Schmeltekopf, A. I., and Ferguson, E. E., J. Chem. Phys. 49, 1365-1371 (1968).

Fehsenfeld, F. C., Schmeltekopf, A. L., Dunkin, D. B., and Ferguson, E. E., "Compilation of Reaction Rate Constants Measured in the ESSA Flowing Afterglow System to August 1969", ESSA Technical Report ERL 135-AL3, ESSA, Boulder, Colo., Sept. (1969).

Fehsenfeld, F. C., Dunkin, D. B., and Ferguson, E. E., Planet. Space Sci. 18, 1267-1269 (1970).

Feinberg, R. M., and Camac, M., Quant. Spectrosc. Radiat. Transfer 7, 581-590 (1967).

Ferguson, E. E., Rev. Geophys. Space Phys. 9, 997-1008 (1971).

Fisher, E. R., and Bauer, E., "The Quenching of $O(^1D)$ by N_2", presented at the DASA Symposium on Physics and Chemistry of the Upper Atmosphere, Philadelphia, Pa., 24-26 June (1970).

Fite, W. L., Can. J. Chem. 47, 1797-1807 (1969).

Foner, S. N., and Hudson, R. L., Chem. Phys. 36, 2681 (1962).

Freedman, E., and Daiber, J. W., J. Chem. Phys. 34, 1271 (1961).

Garnett, S. H., Kistiakowski, G. B., and O'Grady, B. V., J. Chem. Phys. 51, 84-91 (1969).

Garvin, D., and Gevantman, L. H., "Chemical Kinetics Data Survey III. Selected Rate Constants for Chemical Reactions in Atmospheric Chemistry," NBS Report 10867, June 1972.

Garvin, D., and Hampson, R. F., Editors, Chemical Kinetics Data Survey VII. Tables of Rate and Photochemical Data for Modelling of the Stratosphere (Revised). NBSIR 74-430, National Bureau of Standards, Washington, D. C., January (1974).

Gear, C. W., Numerical Initial Value Problems in Ordinary Differential Equations, Prentice-Hall, Englewood Cliffs, N. J. (1971).

Geissler, J. E., and Dickinson, R. E., J. Atmos. Terr. Phys. 30, 1505-1521 (1968).

Gille, J.C., and Elingson, R.G., Appl. Op. 7, 471-474 (1968).

Gilmore, F.R., Bauer, E., and McGowan, J.W., J. Quant. Spectrosc. Radiat. Transfer 9, 157-183 (1969).

Goldan, P.D., Schmeltekopf, A.L., Fehsenfeld, F.C., Schiff, H.I., and Ferguson, E.E., J. Chem. Phys. 44, 4095-4103 (1966).

Goody, R.M., Atmospheric Radiation, I, Theoretical Basis, Oxford University (1964).

Hall, T.C., Jr., and Blacet, F.E., J. Chem. Phys. 20, 1745 (1952).

Hampson, R.F., Brown, R.L., Garvin, D., Herron, J.T., Huie, R.E., McKinley, J.D., and Tsang, W., "Chemical Kinetics Data Survey I. Rate Data for Twelve Reactions of Interest for Stratospheric Chemistry", NBS Report 10692, January (1972a).

Hampson, R.F., Braun, W., Garvin, D., Herron, J.T., Huie, R.E., Kurylo, M.J., Laufer, A.H., Okabe, H., and Tsang, W., "Chemical Kinetics Data Survey II. Photochemical and Rate Data for Fifteen Gas Phase Reactions of Interest for Stratospheric Chemistry", NBS Report 10828, April (1972b).

Hays, P.B., and Olivero, J.J., Planet. Space Sci. 18, 1729-1733 (1970).

Herron, J.T., Int. J. Chem. Kin. 1, 527-539 (1969).

Herron, J.T., and Klein, F.S., J. Chem. Phys. 40, 2731 (1964).

Herzberg, G., Molecular Spectra and Molecular Structure II. Infrared and Raman Spectra of Polyatomic Molecules, D. van Nostrand Co., Inc., New York (1945).

Herzfeld, K.F., and Litovitz, T.A., Absorption and Dispersion of Ultrasonic Waves, Academic Press, New York (1959).

Hesstvedt, E., Tellus 17, 341-349 (1965).

Hesstvedt, E., Geophys. Norv. 27, No. 4, 1-35 (1968).

Hoare, D.E., and Patel, M., Trans. Far. Soc. 65, 1325-1333 (1969).

Houghton, J., Quart. J. Roy. Met. Soc. 95, 1-20 (1969).

Hudson, R.D., Rev. Geophys. Space Sci. 9, 305-406 (1971).

Hudson, R.D, and Carter, V.L., Can. J. Chem. 47, 1840-1844 (1969).

Hudson, R.D., Carter, V.L., and Breig, E.L., J. Geophys. Res. 74, 4079-4086 (1969).

Hunt, B.G., J. Atmos. Terr. Phys. 27, 133-144 (1965).

Hunt, B.G., J. Geophys. Res. 71, 1385-1398 (1966).

Hunten, D.M., and McElroy, M.B., Rev. Geophys. 4, 303-328 (1966).

Hunten, D.M., and Strobel, D.F., "Production and Escape of Terrestrial Hydrogen", J. Atm. Sci. 31, 305-317 (1974).

Hushfar, F., Rogers, J.W., and Stair, A.T., Jr., Appl. Optics 10, 1843-1847 (1971).

Ivanov, V.V., "Approximate Solution to the Equation of Radiative Transfer in Line Frequencies", Soviet Astr. 16, 91-94 (1972).

Ivanov, V.V., and Shcherbakov, V.T., Astrophysics 1, 10-16 (1965a).

Ivanov, V.V., and Shcherbakov, V.T., Astrophysics 1, 17-22 (1965b).

Ivanov, V.V., and Shcherbakov, V.T., Astrophysics 1, 10-21 (1965).

Jaffe, S., and Klein, F.S., Trans. Far. Soc. 62, 3135-3141 (1966).

James, T.C., and Kumer, J.B., "Fluorescence of CO_2 near 4.3 Microns: Application to Daytime Limb Radiance Calculations", J. Geophys. Res. 78, 8320-8329 (1973).

Johnson, F.S., "Solar Radiation", Ch. 4 in F.S. Johnson, ed., Satellite Environment Handbook, Stanford University Press, Stanford (1965).

Johnson, R., Brown, H.L., and Biondi, M.A., J. Chem. Phys. 52, 5080-5084 (1970).

Johnston, H.S., "Gas Phase Reaction Kinetics of Neutral Oxygen Species", NSRDS-NBS 20, National Bureau of Standards (1968).

Kasner, W.H., and Biondi, M.A., Phys. Rev. 137, A317-A329 (1965).

Kaufman, F., Ann. Geophys. 20, 106-114 (1964).

Kaufman, F., Can. J. Chem. 47, 1917-1924 (1969).

Keneshea, T. J., "A Technique for Solving the Reaction Rate Equations in the Atmosphere", Environmental Research Papers No. 263, AFCRL-67-0221 (1967).

Keneshea, T. J., and Zimmerman, S. P., J. Atm. Sci. 27, 831-840 (1970).

Kiefer, J. H., and Lutz, R. W., "The Effect of Oxygen Atoms on the Vibrational Relaxation of Oxygen", Eleventh Symposium (International) on Combustion, pp 67-74, The Combustion Institute, Pittsburgh (1967).

King, J. I. F., "The Source Function for an Equilibrium Gray Atmosphere", Astrophys. J. 124, 406-411 (1956).

Krassovsky, V. I., in The Airglow and the Aurorae (ed. E. B. Armstrong and A. Dalgarno), p. 193, Pergamon Press, Oxford (1955).

Kuhn, W. R., and London, J., J. Atmos. Sci. 26, 189-204 (1969).

Kumer, J. B., and James, T. C., "CO_2 and N_2 Vibrational Temperatures in the $50 \lesssim Z \lesssim 130$ km Altitude Range", J. Geophys. Res. 79, 638-648 (1974).

Lane, N. F., and Dalgarno, A., J. Geophys. Res. 74, 3011-3012 (1969).

Lin, C. L., and Kaufman, F., "Deactivation and Reaction of $N(^2D)$", presented at the DASA Symposium on the Physics and Chemistry of the Upper Atmosphere, Stanford Research Institute, Menlo Park, Calif. 23-24 June (1969).

Liu, S. C., and Donahue, T. M., "Realistic Model of Hydrogen Constituents in the Lower Atmosphere and Escape Flux from the Upper Atmosphere", J. Atmos. Sci. 31, 2238-2242 (1974).

Malkmus, W., J. Opt. Soc. Am. 58, 1214-1217 (1968).

Malkmus, W., "Random Lorentz Band Model with Exponential Tailed S^{-1} Line-Intensity Distribution Function", J. Opt. Soc. Am. 57, 323-329 (1967).

Malkmus, W., "Random Band Models with Lines of Pure Doppler Shape", J. Opt. Soc. Am. 58, 1214-1217 (1968).

Markov, M. N., Appl. Opt. 8, 887-891 (1969).

McClatchey, R. A., Fenn, R. W., Selby, J. E. A., Garing, J. S., and Volz, F. E., "Optical Properties of the Atmosphere", Environmental Research Papers, No. 331, AFCRL-70-0527 (1970).

Meira, L. G., Jr., J. Geophys. Res. 202-212 (1971).

Michels, H., Harris, R., Generosa, J., and Rockwood, S., "Configuration Interaction Calculations of Electronic Transition Probabilities in Aluminum Oxide", paper presented at the DASA Symposium on Physics and Chemistry of the Upper Atmosphere, Philadelphia, Pa. 24-26 June 1970.

Millikan, R. C., and White, D. R., Chem. Phys. 39, 3209-3212 (1963).

Mitchell, A. C. G., and Zemansky, M. W., Resonance Radiation and Excited Atoms, Cambridge University Press, Cambridge (1961).

Morgan, J. E., Phillips, L. F., and Schiff, H. I., Disc. Far. Soc. 33, 118-127 (1962).

Morgan, J. E., and Schiff, H. I., Can. J. Chem. 41, 903-912 (1963).

Nicholls, R. W., J. Res. NBS68A, 535-540 (1964a).

Nicholls, R. W., Ann. Geophys. 20, 144-181 (1964b).

Nicolet, M., J. Geophys. Res. 70, 679-689 (1965a).

Nicolet, M., J. Geophys. Res. 70, 691-701 (1965b).

Nicolet, M., Ann. Geophys. 26, 531-546 (1970).

Norton, R. B., and Barth, C. A., J. Geophys. Res. 75, 3903-3909 (1970).

Offerman, D., and van Zahn, U., J. Geophys. Res. 76, 2520-2522 (1971).

Ogg, R. A., Jr., and Sutphen, W. T., J. Chem. Phys. 21, 2078 (1953).

Ogg, R. A., Jr., and Sutphen, W. T., Disc. Far. Soc. 17, 47 (1954).

Oppenheim, U. P., Yair Avia, and Aharon Goldman, Appl. Opt. 8, 1305-1307 (1967).

Pearce, J. B., J. Geophys. Res. 74, 853-851 (1969a).

Pearce, J.B., J. Quant, Spectrosc. Radiat. Transfer 9, 1593-1602 (1969b).

Penner, S.S., Quantitative Molecular Spectroscopy and Gas Emissivities, Addison-Wesley Pub. Co., Reading, Mass. (1959).

Roble, R.G., and Hays, P.B., Planet. Space Sci., 22, 1337-1340 (1970).

Schiff, H.I., Can. J. Chem. 47, 1903-1916 (1969).

Schofield, K., Planet. Space Sci. 15, 643-670 (1967).

Schumb, W.C., Gatterfield, C.N., and Wentworth, R.L., Hydrogen Peroxide, Reinhold, N.Y. (1955).

Schurin, B., and Clough, S.A., J. Chem. Phys. 38, 1855-1857 (1963).

Schuring, B., and Ellis. R.E., J. Chem. Phys. 45, 2528-2532 (1966).

Shahin, M.M., J. Chem. Phys. 47, 4392-4398 (1967).

Shimazaki, T., and Laird, A.R., J. Geophys. Res. 75, 3221-3235 (1970).

Simpson, C.J.S.M., Bridgman, K.B., and Chandler, T.R.D., J. Chem. Phys. 49, 509-522 (1968).

Slanger, T.G., Wood, J. and Black, G., J. Geophys. Res. 76, 8430-8433 (1971).

Snelling, D.R., and Bair, E.J., J. Chem. Phys. 47, 228 (1967).

Spencer, N.W., Brace, L.H., Garignan, G.R., Taeusch, D.R., and Niemann, H., J. Geophys. Res. 70, 2665-2698 (1965).

Stair, A.T., Jr., Ulwick, J.C., Baker, K.D., and Baker, D.J., "Rocketborne Observations of Atmospheric Infrared Emissions in the Aurora Region", to be published in Atmospheres of the Earth and Planets, ed. B.M. McCormac, D. Reidel, (1975).

Stephenson, J.C., Wood, R.E., and Moore, C.B., J. Chem. Phys. 54, 3097-3102 (1971).

Strobel, D.F., J. Geophys. Res. 76, 2441-2453 (1971a).

Strobel, D.F., J. Geophys. Res. 76, 8384-8393 (1971b).

Strobel, D. F., J. Geophys. Res. 77, 1337-1339 (1972).

Strobel, D. F., Hunten, D. M., and McElroy, M. B., J. Geophys. Res. 75, 4307-4321 (1970).

Taylor, R. L., and Bitterman, S., Rev. Mod. Phys. 41, 26-47 (1969).

Taylor, R. L., Camac, M. and Feinberg, R. M., "Measurements of Vibration-Vibration Coupling in Gas Mixtures", Eleventh Symposium (International) on Combustion, pp 49-64, The Combustion Institute, Pittsburgh (1967).

Thomas, R. J., "Total Mixing Ratios", Planet. Space Sci. 22, 175-177 (1974).

van Tright, C., "Asymptotic Expansions of the Integrated Absorptance for Simple Spectral Lines and Lines with Hyperfine Structure and Isotope Shifts", J. Opt. Soc. Am. 58, 669-678 (1968).

U.S. Committee on Extension to the Standard Atmosphere, U.S. Standard Atmosphere, 1962, Government Printing Office, Wash., D. C. (1962).

U.S. Committee on Extension to the Standard Atmosphere, U.S. Standard Atmosphere Supplements, 1966, Government Printing Office, Wash., D. C. (1966).

Varanasi, P., and Penner, S.S., J. Quant. Spectrosc. Radiat. Transfer 7, 279-282 (1967).

Walker, J. C. G., Planet. Space Sci. 16, 321-327 (1968).

Walker, J. C. G., Stolarski, R. S., and Nagy, A. F., Ann. Geophys. 25, 831-839 (1969).

Wark, D. Q., Yamamoto, G., and Lienesch, J. H., J. Atmos. Sci. 19, 369-384 (1962).

Watanabe, K., Zelikoff, M. and Inn, E. C. Y., "Absorption Coefficients of Several Atmospheric Gases", Air Force Cambridge Research Center Report 53-23 (1953).

Weill, G. M., "NI(^{4}S-^{2}D) Radiation in the Night Airglow and Low Latitude Aurora", in B. M. McCormac and A. Omholt, eds., Atmospheric Emissions, van Nostrand Reinhold, New York (1969).

Weller, C.S., and Biondi, M.A., Phys. Rev. 172, 198-206 (1968).

Wofsy, S.C., McConnell, J.C., and McElroy, M.B., "Atmospheric CH_4, CO, and CO_2," J. Geophys. Res. 77, 4477-4493 (1972).

Wofsy, S.C., and McElroy, M.B., "On Vertical Mixing in the Upper Stratosphere and Lower Mesosphere", J. Geophys. Res. 78, 2619-2624 (1973).

Wong, E.L., and Potter, A.E., J. Chem. Phys. 43, 3371-3382 (1965).

Wray, K.L., and Teare, J.D., J. Chem. Phys. 36, 2582-2596 (1962).

Yardley, J.T., J. Chem. Phys. 49, 2816-2821 (1968).

Zelikoff, M., Watanabe, K., and Inn, E.C.Y., J. Chem. Phys. 21, 1643 (1953).

APPENDIX

CHEMISTRY PROGRAM SNAPS2

The chemistry program developed by Degges (1972) is listed here for reference purposes. Its main use has been to estimate diurnal variations in chemical abundances, given initial molecular species concentrations.

The reaction rates originally used are listed in Table A-1. The new values of Garvin and Hampson (1974) are listed in Table A-2 when the values vary by fifty per cent or more.

The chemistry program consists of a main program, SNAPS2, and 6 subroutines; ROMTST, TKEL, SOLLY, CONCEN, COLUMN, and SUNPHI. The main program performs the chemical computations, including the effects of vertical transport. ROMTST embodies the extrapolation method of Bulirsch and Stoer (1966). TKEL is used to compute temperatures at each altitude. SOLLY computes the solar zenith angle and fraction of the solar disk visible at each altitude. COLUMN integrates the number densities of molecular nitrogen, oxygen and ozone above a given altitude in the direction of the sun. CONCEN is called by COLUMN to obtain number densities required for the integration. SUNPHI computes the attenuation of solar flux by absorption between a given altitude and the assumed top of the atmosphere in the solar direction and returns the photodissociation rates required for the chemistry.

The operation of the main program is described briefly below.

Preceding page blank

Table A-1

Reaction Rates for Neutral Species

Reactants and products are in lowest electronic state unless indicated explicitly.
Where two rates are given, (a) is for the reaction as written, (b) is for the reverse.

	Reaction	Rate Coefficient, c.g.s. units	Source	Error Estimate
R1a	$O + O + M \rightarrow O_2 + M$	$3.8 \times 10^{-30} T^{-1} \exp(-171/T)$	Johnston (1968)	High by up to 50%
R1b		$4.57 \times 10^{-5} T^{-1} \exp(-59700/T)$	Johnston (1968)	30%
R2a	$O + O_2 + M \rightarrow O_3 + M$	$4.63 \times 10^{-35} \exp(1060/T)$	Johnston (1968)	30%
R2b		$1.65 \times 10^{-9} \exp(-11920/T)$	Johnston (1968)	30%
R3a	$O + O_3 \rightarrow 2O_2$	$2.0 \times 10^{-11} \exp(-2410/T)$	Johnston (1968)	30%
R3b		$2.12 \times 10^{-11} \exp(-50700/T)$	Johnston (1968)	30%
R4	$H + O_3 \rightarrow OH + O_2$	2.6×10^{-11}	Kaufman (1969)	20% at 300 K
R5a	$O + OH \rightarrow H + O_2$	2.06×10^{-11}	Schofield (1967)	50%
R5b		$4.19 \times 10^{-10} \exp(-8450/T)$	Schofield (1967)	50%
R6	$O + HO_2 \rightarrow OH + O_2$	$\geq 10^{-11}$	Kaufman (1969)	
R7	$H + HO_2 \rightarrow H_2O + O$	$2.0 \times 10^{-10} \exp(-2000/T)$	Bates and Nicolot (1950)	An estimate
R8	$H + O_2 + M \rightarrow HO_2 + M$	$3.0 \times 10^{-32} (273/T)^{1.3}$	Schofield (1967)	30%

	Reaction	Rate Coefficient, c.g.s. units	Source	Error Estimate
R9	$OH + O_3 \rightarrow HO_2 + O_2$	$\leq 5.0 \times 10^{-13}$	Kaufman (1969)	
R10	$HO_2 + O_3 \rightarrow OH + 2O_2$	10^{-14}	Hunt (1964)	An estimate
R11a	$OH + OH \rightarrow H_2O + O$	$9.3 \times 10^{-12} \exp(-390/T)$	Baulch, et al. (1968b)	20% at 300 K
R11b		$9.3 \times 10^{-11} \exp(-9000/T)$	Baulch, et al. (1968b)	20% at 300 K
R12	$OH + HO_2 \rightarrow H_2O + O_2$	$\geq 10^{-11}$	Kaufman (1969)	See Text
R13	$H + HO_2 \rightarrow H_2 + O_2$	2.0×10^{-13}	Clyne and Thrush (1963)	30%
R14	$O + H_2 \rightarrow OH + H$	$7.0 \times 10^{-11} \exp(-5100/T)$	Wong and Potter (1965)	30%
R15	$HO_2 + HO_2 \rightarrow H_2O_2 + O_2$	1.5×10^{-12}	Kaufman (1969)	30%
R16	$OH + H_2O_2 \rightarrow H_2O + HO_2$	4.0×10^{-13}	Foner and Hudson (1962)	50%
R17	$O + H_2O_2 \rightarrow OH + HO_2$	10^{-15}	Foner and Hudson (1962)	50%
R18	$H + H_2O_2 \rightarrow H_2 + HO_2$	10^{-13}	Foner and Hudson (1962)	50%
R19a	$OH + H_2 \rightarrow H_2O + H$	$*3.6 \times 10^{-11} \exp(-2600/T)$	Baulch, etal. (1968b)	35% at 300K
R19b		$*1.4 \times 10^{-10} \exp(-10100/T)$	Baulch, et al. (1968b)	35% at 300 K
R20	$O + H + M \rightarrow OH + M$	$*8.0 \times 10^{-33}$	Bates and Nicolet (1950)	Estimate
R21	$O + OH + M \rightarrow HO_2 + M$	$*1.4 \times 10^{-31}$	Kaufman (1964)	30%

	Reaction	Rate Coefficient, c.g.s. units	Source	Error Estimate
R22	$H + H + M \rightarrow H_2 + M$	$*1.2 \times 10^{-32} (273/T)^{0.7}$	Kaufman (1969)	30%
R23	$H + OH + M \rightarrow H_2O + M$	$*2.5 \times 10^{-31}$	Kaufman (1964)	50%
R24	$H + HO_2 \rightarrow 2OH$	$* \geq 3 \times 10^{-12}$	Kaufman (1969)	
R25	$H + HO_2 \rightarrow H_2 + O_2$	$* \geq 3 \times 10^{-12}$	Kaufman (1969)	
R26	$H + OH \rightarrow H_2 + O$	$*3.0 \times 10^{-11} \exp(-4170/T)$	Kaufman (1969)	30%
R27	$O(^1D) + N_2 \rightarrow O(^3P) + N_2$	5.0×10^{-11}	Donovan and Husain (1970)	50%
R28	$O(^1D) + O_3 \rightarrow O_2 + O_2$	5.0×10^{-11}	Snelling and Bair (1969)	[illegible]
R29	$O(^1D) + H_2 \rightarrow OH + H$	1.0×10^{-11}	Donovan and Husain (1970)	Uncertain
R30	$O(^1D) + H_2O \rightarrow OH + OH$	3.0×10^{-11}	Donovan and Husain (1970)	Uncertain
R31	$O(^1D) + H_2O \rightarrow OH + OH$	$*10^{-11}$	Estimate	
R32	$O(^1D) + CH_4 \rightarrow$ Products	10^{-11}	Donovan and Husain (1970)	Uncertain
R33	$CO + OH \rightarrow CO_2 + H$	$1.0 \times 10^{-12} \exp(-540/T)$	Baulch, et al. (1968a)	30%
R34	$CO + O + M \rightarrow CO_2 + M$	$1.4 \times 10^{-33} \exp(-1250/T)$	Baulch, et al. (1968a)	Extremely uncertain
R35	$CO + HO_2 \rightarrow CO_2 + OH$	$*2.5 \times 10^{-17}$	Schofield (1967)	Uncertain
R36	$CH_4 + OH \rightarrow H_2O + CH_3$	$1.2 \times 10^{-10} \exp(-2960/T)$	Schofield (1967)	30%
R37	$O + CH_4 \rightarrow OH + CH_3$	$3.5 \times 10^{-11} \exp(-4550/T)$	Herron (1969)	50%

	Reaction	Rate Coefficient, c.g.s. units	Source	Error Estimate
R38	$CH_3 + O_2 \rightarrow CH_2O + OH$	$*5.0 \times 10^{-11} \exp(-5000/T)$	Dean and Kistiakowski (1971)	Uncertain
R39	$CH_2O + OH \rightarrow H_2O + HCO$	$*5.3 \times 10^{-10} \exp(-2120/T)$	Schofield (1967)	50%
R40	$CH_2O + HO_2 \rightarrow H_2O_2 + HCO$	$*7.5 \times 10^{-15}$	Schofield (1967)	Uncertain
R41	$HCO + O_2 \rightarrow CO + HO_2$	$*1.0 \times 10^{-13}$	Estimate	Uncertain
R42a	$N + O_2 \rightarrow NO + O$	$1.1 \times 10^{-14} T \exp(-3140/T)$	Baulch, et al. (1969)	30% at 300 K
R42b		$2.6 \times 10^{-15} T \exp(-19500/T)$	Baulch, et al. (1969)	30%
R43a	$N + NO \rightarrow N_2 + O$	$5.[illegible] \times 10^{-11} \exp(-168/T)$	Baulch, et al. (1969)	Factor of 2
R43b		$2.3 \times 10^{-10} \exp(-37500/T)$	Baulch, et al. (1969)	Factor of 2
R44a	$NO + NO \rightarrow N_2 + O_2$	$1.3 T^{-5/2} \exp(-42800/T)$	Freedman and Daib[illegible] (1961)	Highly uncertain
R44b		$15.2 T^{-5/2} \exp(-64300/T)$	Wray and Teare (1962)	Highly uncertain
R45	$N + N + M \rightarrow N_2 + M$	$1.4 \times 10^{-33} \exp(-500/T)$	Schiff (1969)	Factor of 2
R46a	$N + O + M \rightarrow NO + M$	$1.1 \times 10^{-32} (273/T)^{1/2}$	Baulch, et al. (1969)	50%
R46b		$3.8 \times 10^{-7} T^{-1/2} \exp(-74500/T)$	Baulch, et al. (1969)	Factor of 5
R47	$O + NO \rightarrow NO_2 + h\nu$	$6.5 \times 10^{-17} (T/296)$	Baulch, et al. (1970)	Factor of 2
R48	$O + NO_2 \rightarrow NO + O_2$	$1.6 \times 10^{-11} \exp(-300/T)$	Baulch, et al. (1970)	20%

	Reaction	Rate Coefficient, c.g.s. units	Source	Error Estimate
R49	$N + NO_2 \rightarrow N_2O + O$	$*7.8 \times 10^{-12}$	Schiff (1969)	30% at 300 K
R50	$N + NO_2 \rightarrow N_2 + 2O$	2.3×10^{-12}	Schiff (1969)	Factor of 2 at 300 K
R51	$N + NO_2 \rightarrow N_2 + O_2$	$*1.8 \times 10^{-12}$	Schiff (1969)	Factor of 2 at 300 K
R52	$N + NO_2 \rightarrow 2NO$	5.9×10^{-12}	Schiff (1969)	30% at 300K
R53a	$NO + O + M \rightarrow NO_2 + M$	$2.9 \times 10^{-33} \exp(940/T)$	Baulch, et al. (1970)	20%
R53b		$1.8 \times 10^{-8} \exp(-32800/T)$	Baulch, et al. (1970)	25% at 1500 K
R54a	$NO + NO + O_2 \rightarrow NO_2 + NO_2$	$6.6 \times 10^{-39} \exp(530/T)$	Baulch, et al. (1970)	50%
R54b		$6.6 \times 10^{-12} \exp(-13500/T)$	Baulch, et al. (1970)	50%
R55	$O_3 + NO \rightarrow NO_2 + O_2$	$9.5 \times 10^{-13} \exp(-1240/T)$	Schiff (1969)	30%
R56	$N + O_2(^1\Delta_g) \rightarrow NO + O$	$*3.0 \times 10^{-15}$	Clark and Wayne (1969)	30%
R57	$N + O_3 \rightarrow NO + O_2$	$*3.0 \times 10^{-11} \exp(-1200/T)$	Nicolet (1965)	An estimate
R58	$N(^2D) + O_2 \rightarrow NO + O$	6.0×10^{-12}	Lin and Kaufman (1969)	20% at 300 K
R59	$N(^2D) + N_2O \rightarrow NO + N_2$	$*3.6 \times 10^{-13} T^{1/2} \exp(-400/T)$	Slanger, et al. (1971)	50%
R60	$N + OH \rightarrow NO + H$	$*6.8 \times 10^{-11}$	Campbell and Thrush (1968)	20% at 320 K
R61	$N(^2D) + O \rightarrow N(^4S) + O$	2.0×10^{-13}	Weill (1969)	An estimate

*An asterisk preceding the rate coefficient indicates that the reaction is not included in the present version of the chemistry program.

Table A-2

Revised Values for Reaction Rates

From Garvin and Hampson (1974)

R5a	4.2×10^{-11}
R6	$8.0 \times 10^{-11} \exp(-500/T)$
R8	$6.7 \times 10^{-33} \exp(+290/T)$
R9	$1.6 \times 10^{-12} \exp(-1000/T)$
R10	$1.0 \times 10^{-13} \exp(-1250/T)$
R12	should vary between 2×10^{-11} and 2×10^{-12}
R13	$4.2 \times 10^{-11} \exp(-350/T)$
R14	$3.0 \times 10^{-14} \exp(-4480/T)$
R15	$3.0 \times 10^{-11} \exp(-500/T)$
R16	$1.7 \times 10^{-11} \exp(-910/T)$
R17	$2.8 \times 10^{-12} \exp(-2125/T)$
R18	$2.8 \times 10^{-12} \exp(-1900/T)$
R29	2.9×10^{-16}
R30	3.5×10^{-10}
R33	1.4×10^{-13}
R36	$3.0 \times 10^{-12} \exp(-1770/T)$
R47	6.5×10^{-17}
R48	9.1×10^{-12}
R57	5.7×10^{-13}

Table A-3

Photodissociation Processes Considered for Chemistry Program

	Reaction	Source of Cross Sections
J1	$O_2 + h\nu \rightarrow O(^3P) + O(^3P)$	Hudson, Carter and Breig (1969) Ackerman (1970)
J2	$O_2 + h\nu \rightarrow O(^1D) + O(^3P)$	Ackerman (1970)
J3	$O_3 + h\nu \rightarrow O(^3P) + O_2$	Ackerman (1970)
J4	$O_3 + h\nu \rightarrow O(^1D) + O_2$	Ackerman (1970)
J5	$H_2O + h\nu \rightarrow OH + H$	Watanabe, et al. (1953)
J6	$H_2O_2 + h\nu \rightarrow OH + OH$	Schumb, et al. (1955)
J7	$HO_2 + h\nu \rightarrow OH + O(^3P)$	Assumed same as H_2O_2
J8	$NO + h\nu \rightarrow NO^+ + e$	Donahue (1966)
J9	$NO_2 + h\nu \rightarrow NO + O(^3P)$	Hall and Blacet (1952)
J10	$NO_2 + h\nu \rightarrow N(^4S) + O_2$	Strobel (1971b)
J11	$N_2O + h\nu \rightarrow NO + N(^4S)$	Zelikoff, et al. (1953)
J12	$N_2 + h\nu \rightarrow 2N(^4S)$	Hudson and Carter (1969)
J13	$CO_2 + h\nu \rightarrow CO + O$	Watanabe, et al. (1953)
J14	$CH_4 + h\nu \rightarrow CH_3 + H$	Watanabe, et al. (1953)

Solar fluxes for Lyman-α (1216 Å) and wavelengths longer than 1170 Å were taken from Ackerman (1970). Selection of cross section data was aided by the review of Hudson (1971).

Program Flow: Execution of Program SNAPS begins with the initialization of constants used by the program. The values of eddy diffusion coefficients used at altitudes above 40 km are stored in array EDCOEF. The peak value set by the DATA statement occurs at 104 km. The eddy diffusion coefficient profile is lowered by 9 km in the DO loop ending with statement 90 and the low altitude eddy diffusion coefficients in array EDCF are transferred to arrad EDCOEF in the DO loop ending with statement 95. This permits use of separate sets of high and low altitude diffusion coefficients by changing the appropriate data statements, and easy variation of the altitude of maximum turbulent transport. The data described above is then read in. The information on each data card is written to printer output channel 6 as it is read. Columns 13 through 80 of the second through fourteenth data cards are printed in addition to the number placed in columns 2 through 12 so that comments may be added. The first 14 data cards are printed on a page numbered -1. The input concentrations require about 10 pages to print, the first of which is numbered page 0. Further execution is determined by the input data. A brief flow chart is given on the next page.

The outermost control DO loop ending with statement 450 is executed IHRMAX times. Within this loop, the computer clock is called and computer running time since the beginning of the program is stored in variable SECA. Variable locations used in the DO loop ending with statement 110 are set to zero and this loop is entered and executed for each altitude between NBASE and NTOP.

The mean molecular mass RMWT at each altitude is computed by the statements associated with DO loop 102. Temperatures are computed by calls to function TKEL. The graviational acceleration at each altitude is computed and local mixing scale heights H_{av} are computed. Coefficients for the eddy diffusion contributions to g and β terms in Equation (3-4) are determined from a finite difference approximation, and stored in arrays BTED, QJEDL and QJEDU. Factors needed for computation of molecular diffusion contributions are computed and stored in arrays AFAC, BFAC, CFAC, DFAC and RDTEMP. The molecular concentrations at each altitude above the base altitude are scaled to be consistent with the finite difference scheme in the DO loop ending with statement 104. An effective molecular scale temperature is used for the local scale height. Before exit from DO loop 110, N_2 and O_2 concentrations and sample computed values are written to the printer output channel for altitudes between but not including those associated with NBASE and NTOP.

The DO loop ending with statement 400 is entered and executed MMAX times. Each execution advances the integration time by DLTIME seconds. The time at the end of the step is computed in floating and fixed point form. If the value of NPRNT permits, a new page is started and the time and column headings are printed. Molecular diffusion coefficients are computed for the two lowest altitudes in the statements associated with DO loop 130.

The DO loop ending with statement 350 is entered and executed for each altitude. Molecular diffusion coefficients are computed for the next higher altitude, and the molecular diffusion contributions to the β and g terms are computed and combined with the eddy diffusion contributions in the DO loop ending with statement 140. The solar zenith angle CHI and the fraction ETA of the sun's disc visible are computed by Subroutine SOLLY. If the sun is visible, column counts of N_2, O_2, O and O_3 in the solar direction are computed in Subroutine COLUMN and photodissociation rates are computed in Subroutine SUNPHI for the time at the midpoint of the time interval of integration. Reaction rate coefficients are computed for the temperature at the given altitude.

The integration for a global time increment DLTIME at a single altitude is performed in the DO loop ending with statement 300. The operations within this loop are repeated either until convergence to relative accuracy EPS is achieved or until MLMAX repetitions have been performed with no convergence. Each time through this loop, a new number of local time intervals, MMF, is computed to conform to the extrapolation procedure of Subroutine ROMTST. Values of the inital concentrations at the integration altitude in array CONSP are placed in array DCX. The DO loop ending with statement 290 then integrates over the MMF local time steps. The values in array DCX are copied into array TCX and the time increment DTIME is set to one-half the length of the local

increment. An index ITER is set to one and appropriate third body concentrations are computed. The algorithm of Equation 2-10 is then applied for each species in the DO loop ending with statement 255, the results being placed in array RCX. ITER is incremented and the results in array RCX are placed in array TCX. The time increment is doubled and the DO loop ending with statement 255 is again executed, using the intermediate results from the first execution to compute the α, β and g terms. The results of this second cycle, computed in array RCX, are transferred to array DCX and the statements in the DO loop ending with statement 290 repeated until the global time interval DLTIME has been covered. Subroutine ROMTST is called to test the values in array DCX with previous results produced in the DO range ending with statement 290. The improved concentration values are returned in array TCX. If convergence is achieved, control is transferred to statement 310. Otherwise, the DO loop ending with statement 300 is repeated. If this process does not result in convergence in MLMAX tries, the end point values in array RCX are placed in array TCX in the DO loop ending with statement 315. The endpoint results for the next lower altitude, which have been stored in array SCX, are written into array CONSP and the contents of array TCX are placed in array SCX. Production rates for OH by two different reactions are computed and stored in QCOHA and QCOHB. "Rectangular" sums of total column counts are incremented in DZOHA and DZOHB. These values are computed to

provide a rough check on the hydrogen compound chemistry and may be compared with airglow observations. If the value of NPRNT permits, the altitude, QCOHA, QCOHB, DZOHA, DZOHB and the concentrations of 16 molecules are written to the printer output channel. The statements in the DO loop ending with statement 350 are then repeated until concentrations at all altitudes between (but not including) NBASE and NTOP have been integrated over the global time step DLTIME.
When all altitudes have been integrated, a summary of number of iterations is sent to the printer output channel and the time value TIME is incremented by DLTIME.

After the integration has been advanced MMAX times with time increment DLTIME, all concentrations are sent to the punch output channel by the statements in DO loop 420 and to the printer channel by the statements in DO loop 440. The internal computer clock is again accessed and the starting, end, and elapsed time required in DO loop 450 are sent to the printer output channel. Program execution ceases after this outer DO loop has been executed IHRMAX times.

The scaling of concentrations performed in the DO loop ending with statement 104 permits use of output data from runs at another latitude or with a different temperature profile to be used as input data. While there may be some difficulty in reaching convergence at altitudes below 90 km in the first time step, this is not a serious problem. When using a global time step DLTIME of 600 seconds and a value of 0.01 for EPS, computation of chemical changes

over a day for 150 altitudes requires on the order of 20 minutes running time on the CDC 6600. This is longer than would be required by the differencing methods usually employed (e.g., Shimazaka and Lairo, 1970) but the error control compensates for this. It is not necessary to alter time increments at dawn or sunset since the convergence algorithm of Subroutine ROMTST automatically forces the use of smaller time subdividions of the global time interval DLTIME.

Inputs:

The necessary input data to the chemistry program is a set of 14 data cards containing identification and variable parameters and a set of 603 cards containing concentrations of N_2, O_2, $O(^3P)$, O_3, OH, H, H_2O, HO_2, H_2O_2, H_2, CH_4, CO_2, CO, $O(^1D)$, $N(^4S)$, NO, NO_2, N_2O, $N(^2D)$ and Ar at one km intervals between ground level and 200 km. The data is read in the following order:

1. IDENT — 60 characters of identifying comments which are used in output page headings. The first character should be a blank.
2. DTHETA — Co-latitude, degrees, of the computation location. A value of zero corresponds to the north pole.
3. DPHI — East longitude, degrees, of computation location.
4. SOLDEC — Solar declination angle, degrees, used to determine time of year.
5. GREF — Surface graviational acceleration, cm/sec^2, at computation location.

6.	REFF	Effective value of earth radius, km, for use in computing graviational accelerations at points above ground level.
7.	EPS	Relative error permitted for each time step in the integration.
8.	MLMAX	Maximum number of times integration is to be attempted for each time step to obtain relative error EPS.
9.	NBASE	Index of base altitude for computations, altitude in km plus one.
10.	NTOP	Index of highest altitude in computations, altitude in km plus one, restricted to 201 or less in current program.
11.	IHRMAX	Number of times at which complete output is required.
12.	MMAX	Number of integration steps between complete output points.
13.	DLTIME	Integration interval, seconds; the total simulated time for a given run, in seconds, is IHRMAX * MMAX * DLTIME.
14.	TIME	Initial time, seconds, measured from noon; AM values are negative, PM values are positive.
15.	CONSP	The 603 cards containing concentrations and identifying indices; concentrations are read into array CONSP.

Outputs: A complete set of data, identical to the input data, is written on tape 4 at the end of each major time step.

Mnemonic Variables:

Real Arrays:

AFAC — Stores values of numerical approximation to $\frac{1}{T}\frac{\partial T}{\partial z}$.

AMWT — Stores values of molecular weights, gm/mole; values are set by data statement.

BFAC — Stores values of the coefficient of molecular concentration in the numerical approximation to $-\left[\frac{\partial^2 n}{\partial z^2} + \frac{1}{T}\frac{\partial^2 T}{\partial z^2} n - \frac{1}{T^2}\left(\frac{\partial T}{\partial z}\right)^2\right]$

BTED — Stores values of coefficients used in computing contribution to β terms from eddy diffusion.

BTEMD — Stores values of contribution to β term from both eddy and molecular diffusion for each molecule at a given altitude during integration.

CARD — Used as temporary storage for comments in columns 13 through 80 of first 14 data cards.

CFAC — Stores values of numerical approximation used to compute $\frac{1}{H^2}\frac{\partial H}{\partial z}$ for diffusive scale heights.

CMD — Stores values of molecular diffusion coefficients for each molecular species at a given altitude during integration.

CMDL — Stores values of CMD at next lower altitude.

CMDU — Stores values of CMD at next higher altitude.

COL — Stores values of integrated column counts computed by Subroutine COLUMN.

CONSP	Stores values of molecular number densities; initial values are read in from cards; values are altered at each integration step.
DCX	Stores initial number densities for one altitude during local time step in integration.
DFAC	Stores values of g/RT used in computing atmospheric scale heights and contribution of molecular diffusion.
EDCF	Stores values of eddy diffusion coefficients for altitudes below 40 km, set by DATA statement.
EDCOEF	Stores values of eddy diffusion coefficients, initially contains high altitude values set by DATA statement.
QJEDL	Stores coefficients of eddy diffusion contribution to g terms from next lower altitude.
QJEDU	Stores coefficients of eddy diffusion contributions to g terms from next higher altitude.
QJEMD	Stores values of contribution to g term from eddy and molecular diffusion for each molecular species at a given altitude during integration.
RATE	Stores values of photodissociation rates computed by Subroutine SUNPHI.
RCX	Stores intermediate values of number densities for a single altitude computed during local time step in integration.

RDTEMP	Stores values of temperature coefficients used in computing molecular diffusion coefficients.
SCX	Stores values of number densities computed at end of global time step for next lower altitude; values eventually replace corresponding numbers in CONSP.
TCX	Stores intermediate values of number densities used in computing α, β and g values during integration of local time step for a single altitude.
TEMPK	Stores values of temperatures at each altitude, values are computed by function TKEL.
TET0T	Stores values of total electron densities at each altitude, values are set by DATA statement.
TFMT	Stores format statement for computer running time printout, set by DATA statement.
TNO	Stores values of NO^+ concentrations at each altitude, set by DATA statement.
TO	Stores values of O^+ concentrations at each altitude, set by DATA statement.
TO2	Stores values of O_2^+ concentrations at each altitude, set by DATA statement.
TST	Used as workspace by Subroutine ROMTST.

The names of integer arrays and their uses are:

IDENT	Stores 60 character comment read from

	first data card, which is used in page headings.
IDNT	Stores image of last 20 characters of first data card.
LSP	Stores Hollerith constants used in printing column headings.
MLPRNT	Stores data on number of iterations required for convergence of integration at a given altitude, and (largest) index of species which did not converge in MLMAX iterations.

Real Variables:

AFACNA	Temporary storage of value of AFAC for altitude at which integration is being performed.
BFACNA	Temporary storage of value of BFAC for altitude at which integration is being performed.
BTEDNA	Temporary storage of value of BTED for altitude at which integration is being performed.
BTSQ	Square of β term used in integration.
BTT	Value (partial or complete) of β term used in integration for a single molecular species.
CFACNA	Temporary storage of value of CFAC for altitude at which integration is being performed.
CHI	Solar zenith angle, radiance, computed by Subroutine SOLLY.

CHISAV	Temporary storage for value of CHI.
CMDNSP	Temporary storage for value of molecular diffusion coefficient for single molecular species.
CON	Temporary storage for factor used in scaling molecular concentrations at altitude NTOP to fit scale height defined by molecular scale temperature.
CONFAC	Temporary storage for factor used to scale molecular concentrations to fit local scale heights defined by molecular scale temperature.
CONSPN	Temporary storage for concentration of a single molecular species at a given altitude.
C1	Temporary storage for molecular nitrogen concentration.
C2	Temporary storage for molecular oxygen concentration.
C3	Temporary storage for atomic oxygen concentration.
DCT	Temporary storage for single value from array DCX.
DDDZ	Temporary storage for value of $\frac{\partial D}{\partial z}$ for single molecular species.
DELZ	Altitude increment, cm.
DFACNA	Temporary storage for value of DFAC at altitude for which integration is being performed.
DHDZ	Temporary storage for value of $\frac{\partial H}{\partial z}$.
DKDZ	Temporary storage for value of $\frac{\partial K}{\partial z}$.

DLT	Value of δ used in integration for a single molecular species.
DLTIME	Time increment for integration step, read in as data.
DPHI	East longitude of point of integration, read in as data; not needed in present program which assumes times are true solar times.
DTDZ	Temporary storage for value of $\frac{\partial T}{\partial z}$.
DTHETA	Colatitude of point at which integration is performed, read in as data.
DTIME	Storage for value of local time step in integration.
DZOHA DZOHB	Values of integrated production rates of OH.
ED	Temporary storage for value of eddy diffusion coefficient at a given altitude.
EPS	Value of relative error allowed in integration.
ETA	Fraction of solar disc visible at given altitude.
FAC	Temporary storage for intermediate results, use varies in different parts of program.
FACB	Temporary storage for either βt or δt in Equations (2-10).
FACDMB	Temporary storage for $\delta - \beta$ in Equatio (3-5).
FACM	Temporary storage for value of $1 - \exp(-\delta t)$.
FACRT	Temporary storage for $4 \alpha g/\beta^2$.
FAC2AX	Temporary storage for $2 \alpha x$.
FAC4AQ	Temporary storage for $4 \alpha g$.
G	Value of graviational acceleration at altitude NALT.

GL	Value of graviational acceleration at altitude NALT -1.
GREF	Surface value of graviational acceleration, read in as data.
GU	Value of graviational acceleration at altitude NALT +1.
QCOHA, QCOHB	Values of OH production rates at altitude of integration.
QJEDLN	Temporary storage for value of QJEDL at altitude of integration.
QJEDUN	Temporary storage for value of QJEDU at altitude of integration.
QJT	Temporary storage for partial or complete value of g term used in integration of a single molecular species.
RDELZ	Reciprocal of twice the altitude increment DELZ.
RE	Effective radius of earth, km, used in geometrical computations.
REFF	Effective radius of earth, km, used in gravitational calculations, read in as data.
RFAC	Temporary storage of intermediate results.
RI1, ...	Storage for values of ionic reaction rates.
RMWT	Mean molecular mass at altitude NALT.
RMWTL	Mean molecular mass at altitude NALT -1.
RMWTU	Mean molecular mass at altitude NALT +1.
ROBS	Altitude of given point, km.
RTIME	Variable used in printing time values.
R1, ...	Values of forward neutral reaction rates.
R2DELZ	Square of reciprocal of altitude increment, DELZ.

SECA, SECB, SECC	Storage for values of computation of computer running time.
SFAC	Temporary storage of intermediate results.
SH	Value of mixing scale height at altitude NALT.
SHL	Value of mixing scale height at altitude NALT -1.
SHU	Value of mixing scale height at altitude NALT +1.
SI1, ...	Values of reverse ionic reaction rates.
SOLDEC	Solar declination angle, read in as data.
SUM, SUML, SUMU	Temporary storage used in forming mean molecular masses.
S1, ...	Values of reverse neutral reaction rate coefficients.
TCH4, ...	Values of individual molecular concentrations in array TCX, named individually for convenience in forming α, β and g coefficients. TCH4 corresponds to CH_4, TCO corresponds to CO, etc.
TEMP	Value of temperature at altitude NALT.
TEMPI	Reciprocal of temperature.
TEMPL	Value of temperature at altitude NALT - 1.
TEMPU	Value of temperature at altitude NALT +1.
TEMP12	Value of minus one-half power of temperature.
TEMP32	Value of minus three-halves power of temperature.

TEMP52	Value of minus five-halves power of temperature.
TFAC	Temporary storage of intermediate results.
TIME	Time, expressed in seconds from noon, read in as data; altered in program to be the time at midpoint of a global integration step so that approximate mean values of photodissociation rates will be computed by Subroutine SUNPHI.
XM4, . . .	Third body concentrations for appropriate three-body reactions.
XXX12	Storage for factor used in computing g term for CO_2 and β term for CO.

Integer Variables:

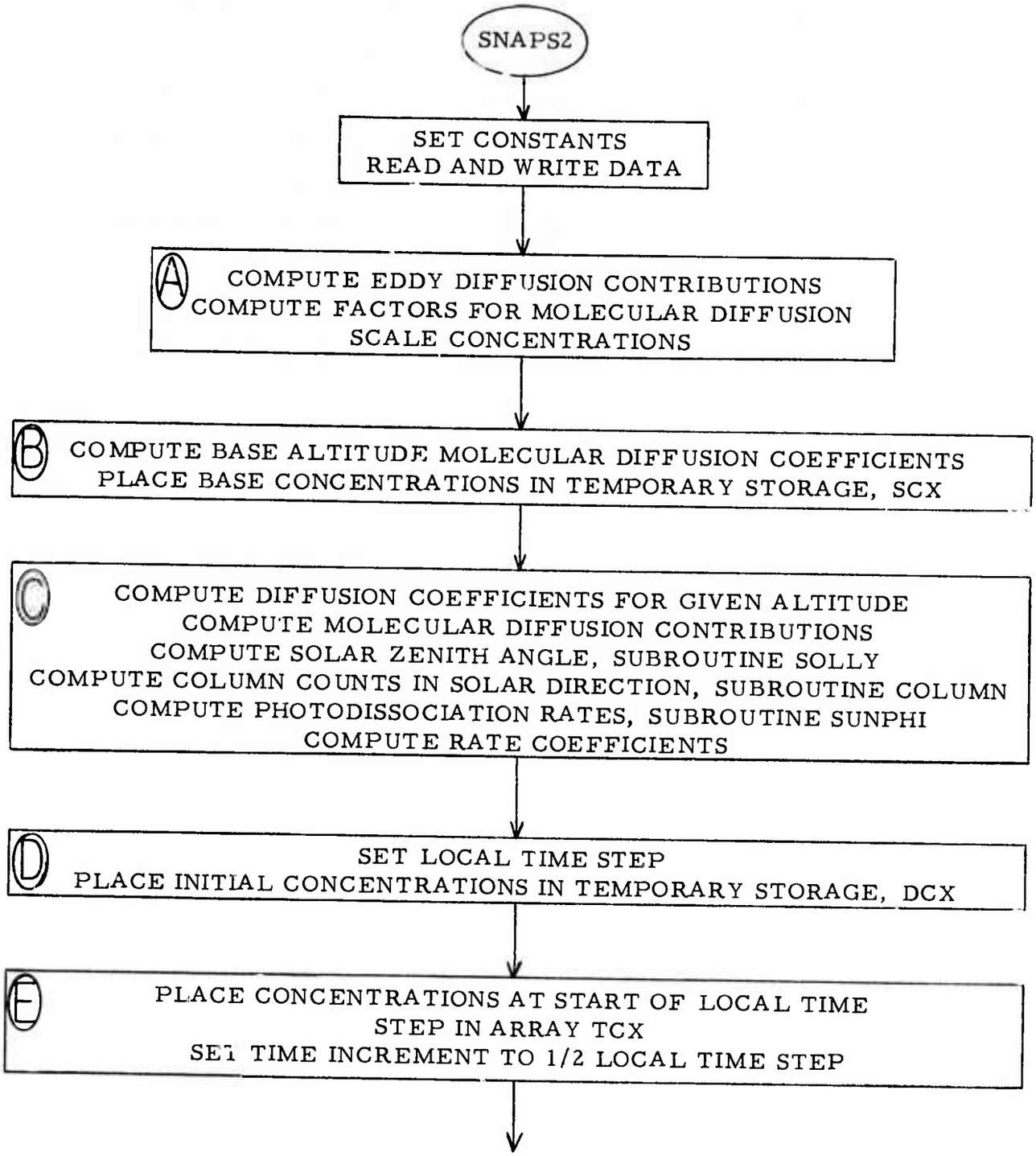
I	Index in DO loop 95, auxiliary index in DO loop 110, and index in aimplied input/output DO loops.
IF	Final parameter in innermost implied DO loop used in printing output concentrations within DO loop 440.
IHR	Index in DO loop 450.
IHRMAX	Final parameter in DO loop 450, input as data.
IP	Index in print control DO loop 440.
IS	Initial parameter in implied DO loop, see IF.
ITER	Control index in integration, used between statements 200 and 260 to select proper values for array TCX.

IX	Index in DO loop 435, incremented for each page of output.
J	Index in implied DO loops used in reading data into array CONSP within DO loop 100.
KCH4, ...	Index variables used to select appropriate values of quantities related to given species; most uses replaced by use of TCH4, etc.
KTEST	Parameter returned by Subroutine ROMTST, zero if integration has converged, non-zero otherwise.
LALT, LCHI, LCOLN2, LCOLO2, LCOLO3, LDZOHA, LDZOHB, LJO2, LJO3, LML, LQCOHA, LQCOHB, LTIME	Store Hollerith constants used in printing column headings and are set by DATA statements.
M	Index of DO loop 400, incremented for each global time step.
ML	Index of DO loop 300, incremented for each iteration of integration over global time step at a given altitude.
MLMAX	Final parameter of DO loop 300, read in as data.
MLSAVE	Temporary storage for value of ML, used after exit from DO loop 300 to print number of iterations required for convergence.
MMAX	Final parameter of DO loop 400, read in as data.
MMF	Final parameter of DO loop 290, computed from ML and is number of local subdivisions of global time step DLTIME for the ML iteration of integration.

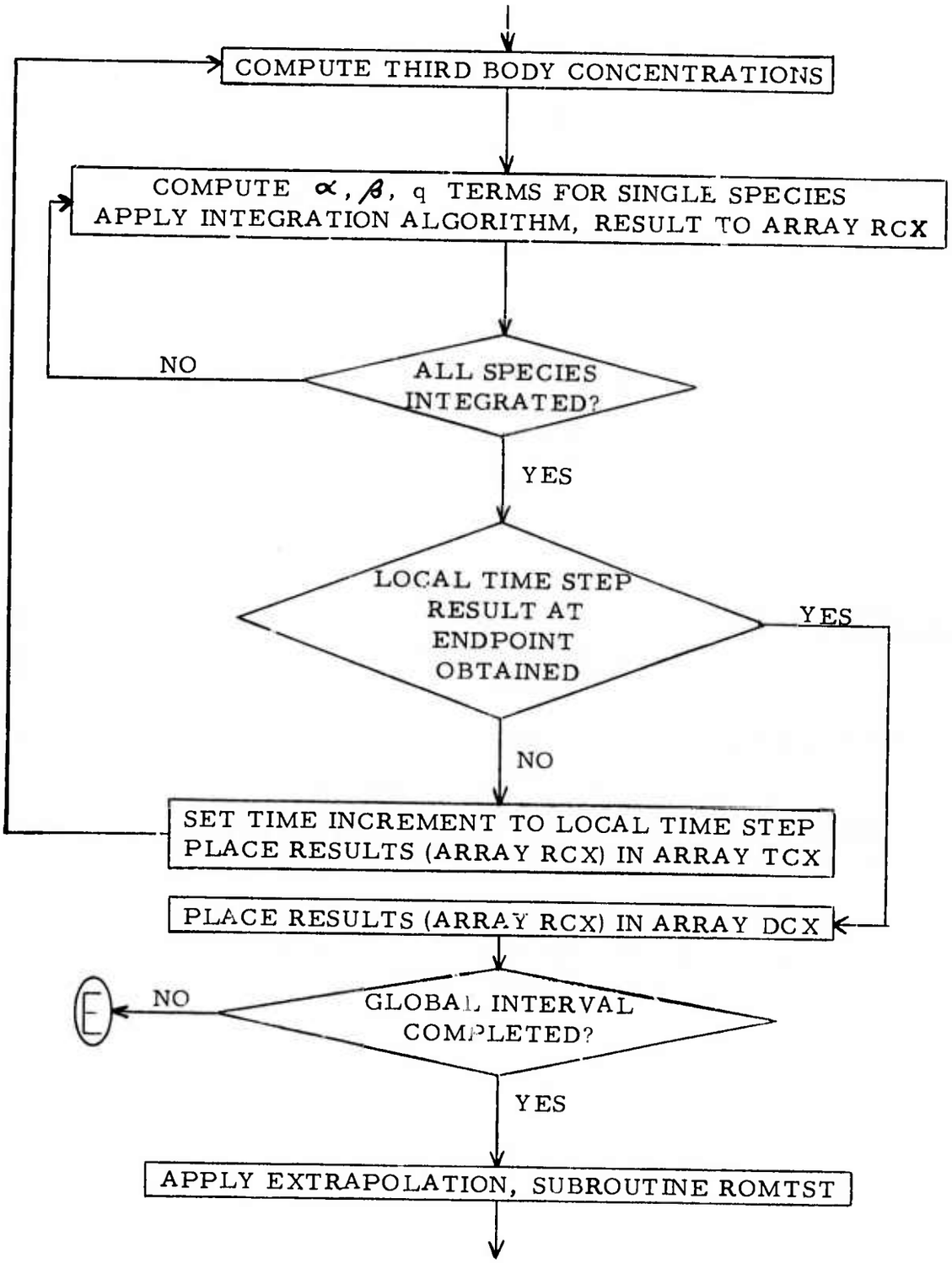
MMS	Index for DO loop 290, see MMF.
NA	Auxiliary index, usually set to NALT -1.
NALT	Index for DO loops 90, 100, 110, 350, 420 and 430; denotes a given altitude.
NAMIN	Value of NALT -1 in DO loop 350.
NB	Value of NALT -2 in DO loop 110.
NBASE	Index of lowest altitude for integration, read in as data.
NF	Final parameter of DO loop 430.
NMIN	Stores input value of NBASE, which is changed by program.
NPAGE	Stores output page number.
NPLU	Value of NALT +1 in DO loop 350.
NPRNT	Variable available to control output of concentrations at integration steps intermediate to fully documented output at end of DO loop 450, set to 1 at present, so values are output for each interval DLTIME.
NS	Initial parameter of DO loop 430.
NSP	Index in DO loops 102, 104, 130, 140, 180, 190, 255, 258, 280, 315, and 320; a value of NSP selects a value of a variable corresponding to a single molecular species.
NSPM	Number of molecular species included in program, 20 at present, used as final parameter of DO loops with NSP as index.
NT	Final parameter of DO loop 420, in which output cards are punched.

NTIME	Used in printing time values.
NTOP	Index of highest altitude used in integration, input as data.
NW	Channel number for printing computer running time information, set to 6 here.
NXX	Temporary storage for altitude index read from cards containing concentration data.
N1, N2, N3	Storage space used for sequence numbers of cards read in or punched out at a given altitude.

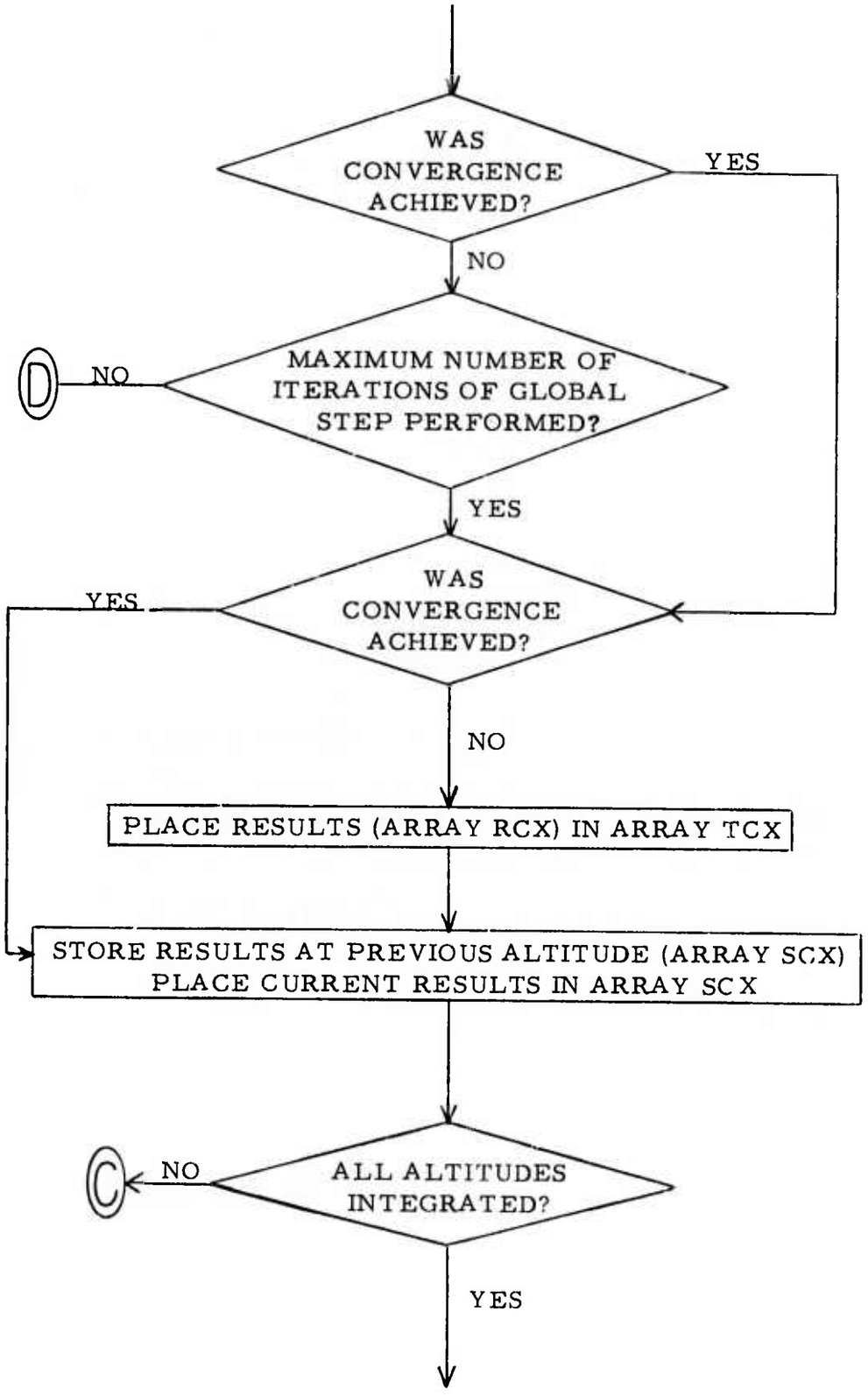
Flow Chart for Chemistry Program



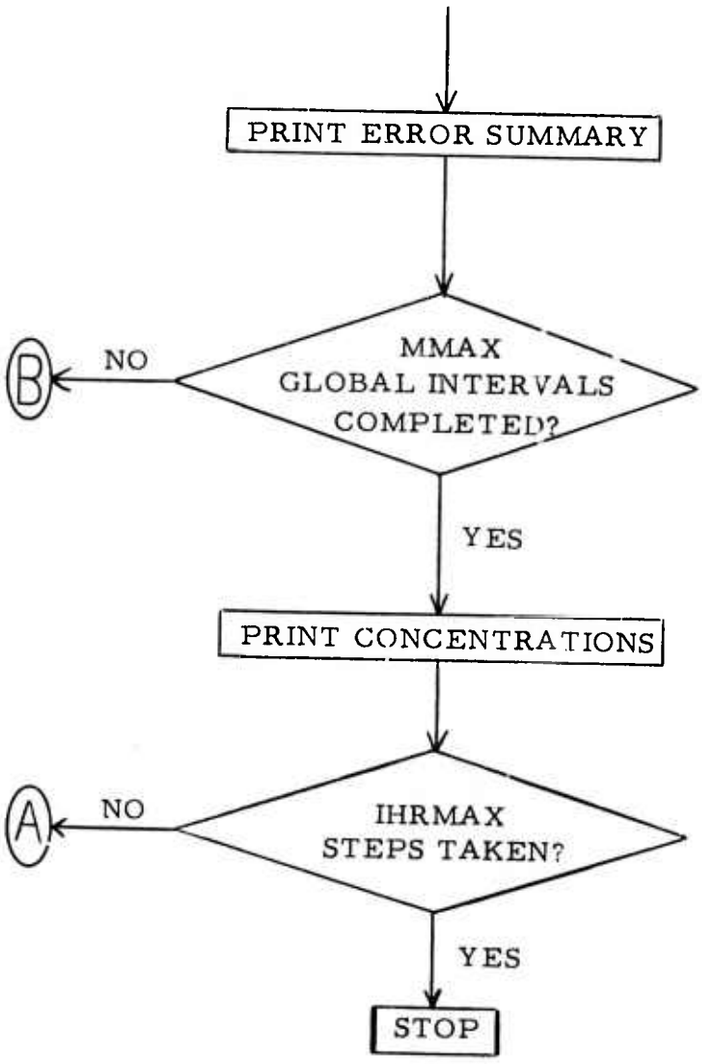
Flow Chart for Chemistry Program (continued)



Flow Chart for Chemistry Program (continued)



Flow Chart for Chemistry Program (continued)



```
      PROGRAM SNAPS2 (INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,      SNAP  10
     1  TAPE4=PUNCH,TAPE3)                                              SNAP  20
C                                                                       SNAP  30
C                                                                       SNAP  40
C     **********    VISIDYNE, INC.    WOBURN, MASS.    **********       SNAP  50
C                                                                       SNAP  60
C                                                                       SNAP  70
C                                                                       SNAP  80
      COMMON CONSP(501,20)                                              SNAP  90
      COMMON AFAC(501),BFAC(501),CFAC(501),DFAC(501),RCTEMP(501)        SNAP 100
C                                                                       SNAP 110
      DIMENSION CONC(30)                                                SNAP 120
      DIMENSION RATE(30)                                                SNAP 130
      DIMENSION CCL(10)                                                 SNAP 140
      DIMENSION DCX(50),RCX(50),SCX(50),TCX(50)                         SNAP 150
      DIMENSION QJEC(50)                                                SNAP 160
      DIMENSION BTED(501),QJEDL(501),QJEDU(501)                         SNAP 170
      DIMENSION CMDU(50),CMC(50),CMDL(50),BTEMD(50),QJEMC(50)           SNAP 180
       DIMENSION AMNT(50)                                               SNAP 190
      DIMENSION ECCCEF(501),TEMPK(501)                                  SNAP 200
      DIMENSION IDENT(10)                                               SNAP 210
      DIMENSION ICNT(4)                                                 SNAP 220
      DIMENSION CARD(12)                                                SNAP 230
      DIMENSION TST(50,10)                                              SNAP 240
      DIMENSION LSP(25)                                                 SNAP 250
      DIMENSION MLPRNT(501)                                             SNAP 260
      DIMENSION ECCF(41)                                                SNAP 270
      DIMENSION TINO(501)                                               SNAP 280
      DIMENSION TETOT(501)                                              SNAP 290
      DIMENSION TIO(501),TIC2(501)                                      SNAP 300
      DIMENSION TFMT(6)                                                 SNAP 310
C                                                                       SNAP 320
C                                                                       SNAP 330
C                                                                       SNAP 340
C                                                                       SNAP 350
      EQUIVALENCE      (TCX(1),TN2),   (TCX(2),TC2),   (TCX(3),TC),     SNAP 360
```

```
     1  (TCX(4),TO3),   (TCX(5),TOH),   (TCX(6),TH),    (TCX(7),TH2C),   SNAP 370
     2  (TCX(8),THC2),  (TCX(9),TH2O2), (TCX(10),TH2),  (TCX(11),TCH4), SNAP 380
     3  (TCX(12),TCO2), (TCX(13),TCO),  (TCX(14),TO1D), (TCX(15),TN),   SNAP 390
     4  (TCX(16),TNO),  (TCX(17),TNO2), (TCX(18),TN2C), (TCX(19),TN2O)  SNAP 400
C                                                                       SNAP 410
C                                                                       SNAP 420
      DATA TST /500*0.0/                                                SNAP 430
      DATA TINO /50*10.0,20.0,40.0,80.0,1.6E2,3.2E2,6.5E2,30*1.0E3,     SNAP 440
     1 1.5E3,2.0E3,2.5E3,3.0E3,3.5E3,4.0E3,4.5E3,5.0E3,5.5E3,6.0E3,     SNAP 450
     2 1.1E4,1.6E4,2.1E4,2.6E4,3.2E4,3.5E4,3.8E4,4.1E4,4.3E4,4.7E4,     SNAP 460
     3 395*5.0E4/                                                       SNAP 470
      DATA TETOT /50*10.0,20.0,40.0,80.0,1.6E2,3.2E2,6.5E2,26*1.0E3,    SNAP 480
     1 1.1E3,1.2E3,1.4E3,1.5E3,2.4E3,3.3E3,4.2E3,5.1E3,6.0E3,6.8E3,     SNAP 490
     2 7.6E3,8.4E3,9.2E3,1.0E4,1.9E4,2.8E4,3.7E4,4.6E4,5.5E4,6.4E4,     SNAP 500
     3 7.3E4,8.2E4,9.1E4,396*1.0E5/                                     SNAP 510
      DATA TIO /91*0.0,0.1,0.2,0.3,0.4,0.5,0.7,1.0,1.5,2.5,5.0,         SNAP 520
     1  10.0,15.0,20.0,25.0,30.0,35.0,40.0,50.0,60.0,80.0,100.0,        SNAP 530
     2  120.0,140.0,160.0,180.0,220.0,300.0,400.0,500.0,600.0,          SNAP 540
     3  700.0,750.0,800.0,850.0,900.0,1.0E3,1.2E3,1.4E3,1.6E3,          SNAP 550
     4  1.8E3,2.0E3,2.2E3,2.4E3,2.7E3,3.0E3,3.5E3,4.0E3,4.5E3,          SNAP 560
     5  5.0E3,5.5E3,6.0E3,7.0E3,8.0E3,9.0E3,1.0E4,1.1E4,1.2E4,          SNAP 570
     6  1.3E4,1.4E4,1.5E4,1.8E4,2.1E4,2.4E4,2.7E4,3.0E4,3.3E4,          SNAP 580
     7  3.6E4,3.9E4,4.2E4,341*4.5E4/                                    SNAP 590
      DATA TIO2 /81*0.0,100.0,200.0,300.0,400.0,600.0,900.0,            SNAP 600
     1  1.2E3,1.6E3,2.0E3,2.3E3,2.8E3,3.1E3,3.4E3,3.7E3,4.0E3,          SNAP 610
     2  5.0E3,7.0E3,1.0E4,1.5E4,2.0E4,2.3E4,2.6E4,2.9E4,3.2E4,          SNAP 620
     3  3.5E4,3.8E4,4.1E4,4.4E4,4.7E4,391*5.0E4/                        SNAP 630
C                                                                       SNAP 640
      DATA LALT,LTIME,LCHI,LCOLO2,LCOLN2,LCOLO3,LJO2,LJO3,LQCOHA,       SNAP 650
     1  LQCOHB,LDZCHA,LDZCHB,LML  /6H   ALT,6H TIME ,6H  CHI ,6H COLO2,  SNAP 660
     2  6H COLN2,6H COLO3,6H  JO2 ,6H  JO3 ,6HQA(OH),6HQB(CH),6HIA(OH),  SNAP 670
     3  6HIB(OH),6H    ML/                                              SNAP 680
C                                                                       SNAP 690
      DATA  AMWT       / 28.0134, 31.9988, 15.9994, 47.9982, 17.0074,   SNAP 700
     1  1.00797, 18.0153, 33.0067, 34.0147, 2.01594, 16.0430, 44.0099,  SNAP 710
     2  28.0105, 15.9994, 14.0067, 30.0061, 46.0055, 44.0128, 14.0067,  SNAP 720
```

```
     3  39.9480, 30*0.0 /                                                    SNAP 730
C                                                                           SNAP 740
      DATA EDCOEF / 51*1.0E5,1.096E5,1.202E5,1.318E5,1.445E5,1.585E5,       SNAP 750
     1  1.738E5,1.905E5,2.089E5,2.291E5,2.512E5,2.754E5,3.02E5,3.31E5,      SNAP 760
     2  3.63E5,3.98E5,4.36E5,4.79E5,5.25E5,5.75E5,6.31E5,6.92E5,7.59E5,     SNAP 770
     3  8.32E5,9.12E5,1.0E6,1.096E6,1.202E6,1.318E6,1.445E6,1.585E6,        SNAP 780
     4  1.738E6,1.905E6,2.089E6,2.291E6,2.512E6,2.754E6,3.02E6,3.31E6,      SNAP 790
     5  3.63E6,3.98E6,4.36E6,4.78E6,5.25E6,5.75E6,6.31E6,6.92E6,7.59E6,     SNAP 800
     6  8.32E6,9.12E6,1.00E7,1.09E7,1.20E7,1.24E7,1.27E7,1.24E7,            SNAP 810
     7  1.18E7,1.05E7,8.20E6,3.162E6,1.00E6,3.162E5,1.00E5,3.162E4,         SNAP 820
     8  1.00E4,3.162E3,1.00E3,3.162E2,1.00E2,3.162E1,1.00E1,380*1.0/        SNAP 830
      DATA EDCF /12*1.0E5,7.943E4,6.310E4,5.012E4,3.981E4,3.162E4,         SNAP 840
     1  2.512E4,1.995E4,1.585E4,1.259E4,11*1.0E4,1.259E4,1.585E4,           SNAP 850
     2  1.995E4,2.512E4,3.162E4,3.981E4,5.012E4,6.310E4,7.943E4/            SNAP 860
C                                                                           SNAP 870
      DATA LSP /6H  N2  ,6H  O2  ,6H O(3P),6H  O3  ,6H  CH  ,6H   H  ,      SNAP 880
     1  6H  H2O ,6H  HO2 ,6H H2O2 ,6H  H2  ,6H  CH4 ,6H  CO2 ,6H  CO  ,      SNAP 890
     2  6H O(1D),6H N(4S),6H  NO  ,6H  NO2 ,6H  N2O ,6H N(2D),               SNAP 900
     3  6*6H     /                                                          SNAP 910
C                                                                           SNAP 920
      DATA TFMT /10H(/5X,5HSTA,10HRT,6X,3HEN,10HD,7X,5HTCT,                 SNAP 930
     1  10HAL,6X,5HTI,10HMES/1X,3F1,5H0.3/)/                                SNAP 940
C                                                                           SNAP 950
C                                                                           SNAP 960
C       STATEMENT FUNCTION TO CONVERT DEGREES TO RADIANS                    SNAP 970
      RAD(DEG) = 1.7453292519943E-2*DEG                                     SNAP 980
C                                                                           SNAP 990
C       STATEMENT FUNCTION TO COMPUTE 1.0 - EXP(-X)                         SNAP1000
      EXPM(X) = ((((0.008333*X - 0.04166667)*X + 0.16666667)*X              SNAP1010
     1  - 0.5)*X + 1.0)*X                                                   SNAP1020
C                                                                           SNAP1030
C                                                                           SNAP1040
C       EDDY DIFFUSION COEFFICIENTS ARE STORED IN ARRAY EDCOEF              SNAP1050
C       KINETIC TEMPERATURES ARE STORED IN ARRAY TEMPK                      SNAP1060
C       NUMBER DENSITIES FOR ALL ALTITUDES AND SPECIES ARE                  SNAP1070
C           STORED IN ARRAY CONSP                                           SNAP1080
```

```
C       NUMBER DENSITIES FOR ALTITUDE NALT ARE STORED IN ARRAY DCX       SNAP1090
C       NUMBER DENSITIES FOR NALT - 1 AT TIME T ARE STORED IN ARRAY SCX SNAP1100
C       ARRAYS RCX AND TCX STORE APPROXIMATIONS DURING INTEGRATION       SNAP1110
C       DIFFUSION CONTRIBUTIONS TO THE G COEFFICIENTS ARE                SNAP1120
C           STORED IN ARRAY GJED                                         SNAP1130
C       DIFFUSION CONTRIBUTIONS TO THE BETA COEFFICIENTS ARE             SNAP1140
C           STORED IN ARRAY ETED                                         SNAP1150
C       TERMS USED TO COMPUTE QJED ARE STORED IN ARRAYS GJEDL AND QJEDU SNAP1160
C                                                                        SNAP1170
C                                                                        SNAP1180
C     INIALIZATION OF CONSTANTS                                          SNAP1190
      NW = 6                                                             SNAP1200
      NPAGE = -1                                                         SNAP1210
      PI = 3.1415926535898                                               SNAP1220
      RE = 6371.0                                                        SNAP1230
      DELZ = 1.0E5                                                       SNAP1240
      RDELZ = 0.5/DELZ                                                   SNAP1250
      R2DELZ = DELZ**(-2)                                                SNAP1260
      NPRNT = 1                                                          SNAP1270
     DO 90 NALT = 1, 150                                                 SNAP1280
      EDCOEF(NALT) = EDCOEF(NALT + 9)                                    SNAP1290
   90 CONTINUE                                                           SNAP1300
     DO 95 I = 1, 41                                                     SNAP1310
      EDCOEF(I) = EDCF(I)                                                SNAP1320
   95 CONTINUE                                                           SNAP1330
C                                                                        SNAP1340
C                                                                        SNAP1350
C       FOR CONVENIENCE IN CHECKING FORMATION OF COEFFICIENTS IN THE     SNAP1360
C           DIFFERENTIAL EQUATIONS, INDEX VALUES OF THE FORM KN2, KC2,   SNAP1370
C           ETC  ARE USED IN PLACE OF SPECIFIC VALUES OF NSP.            SNAP1380
      KN2 = 1                                                            SNAP1390
      KC2 = 2                                                            SNAP1400
      KC  = 3                                                            SNAP1410
      KO3 = 4                                                            SNAP1420
      KOH = 5                                                            SNAP1430
      KH  = 6                                                            SNAP1440
```

```
      KH2O = 7                                                          SNAP1450
      KHC2 = 8                                                          SNAP1460
      KH2C2 = 9                                                         SNAP1470
      KH2 = 10                                                          SNAP1480
      KCH4 = 11                                                         SNAP1490
      KC02 = 12                                                         SNAP1500
      KCO = 13                                                          SNAP1510
      KC1D = 14                                                         SNAP1520
      KN = 15                                                           SNAP1530
      KNO = 16                                                          SNAP1540
      KNO2 = 17                                                         SNAP1550
      KN2O = 18                                                         SNAP1560
      KN2O = 19                                                         SNAP1570
C                                                                       SNAP1580
       READ (5,21) IDENT,IDNT                                           SNAP1590
      WRITE (4,21) IDENT,IDNT                                           SNAP1600
      WRITE (6,20) IDENT,NPAGE                                          SNAP1610
      NPAGE = NPAGE + 1                                                 SNAP1620
C                                                                       SNAP1630
C     PRINT INPUT PARAMETERS                                            SNAP1640
C                                                                       SNAP1650
C                                                                       SNAP1660
      WRITE (6,21) IDENT,IDNT                                           SNAP1670
       READ (5,15) DTHETA,CARD                                          SNAP1680
      WRITE (6,15) DTHETA,CARD                                          SNAP1690
      WRITE (4,15) DTHETA,CARD                                          SNAP1700
       READ (5,15) DPHI,CARD                                            SNAP1710
      WRITE (6,15) DPHI,CARD                                            SNAP1720
      WRITE (4,15) DPHI,CARD                                            SNAP1730
       READ (5,15) SCLDEC,CARD                                          SNAP1740
      WRITE (6,15) SCLDEC,CARD                                          SNAP1750
      WRITE (4,15) SCLDEC,CARD                                          SNAP1760
       READ (5,15) GREF,CARD                                            SNAP1770
      WRITE (6,15) GREF,CARD                                            SNAP1780
      WRITE (4,15) GREF,CARD                                            SNAP1790
       READ (5,15) REFF,CARD                                            SNAP1800
```

```
      WRITE (6,15) REFF,CARD                                            SNAP1810
      WRITE (4,15) REFF,CARD                                            SNAP1820
       READ (5,15) EPS,CARD                                             SNAP1830
      WRITE (6,15) EPS,CARD                                             SNAP1840
      WRITE (4,15) EPS,CARD                                             SNAP1850
       READ (5,16) MLMAX,CARD                                           SNAP1860
      WRITE (6,16) MLMAX,CARD                                           SNAP1870
      WRITE (4,16) MLMAX,CARD                                           SNAP1880
       READ (5,16) NBASE,CARD                                           SNAP1890
      WRITE (6,16) NBASE,CARD                                           SNAP1900
      WRITE (4,16) NBASE,CARD                                           SNAP1910
       READ (5,16) NTOP,CARD                                            SNAP1920
      WRITE (6,16) NTOP,CARD                                            SNAP1930
      WRITE (4,16) NTOP,CARD                                            SNAP1940
       READ (5,16) IHRMAX,CARD                                          SNAP1950
      WRITE (6,16) IHRMAX,CARD                                          SNAP1960
      WRITE (4,16) IHRMAX,CARD                                          SNAP1970
       READ (5,16) MMAX,CARD                                            SNAP1980
      WRITE (6,16) MMAX,CARD                                            SNAP1990
      WRITE (4,16) MMAX,CARD                                            SNAP2000
       READ (5,15) DLTIME,CARD                                          SNAP2010
      WRITE (6,15) DLTIME,CARD                                          SNAP2020
      WRITE (4,15) DLTIME,CARD                                          SNAP2030
       READ (5,15) TIME,CARD                                            SNAP2040
      WRITE (6,15) TIME,CARD                                            SNAP2050
C                                                                       SNAP2060
C     ************                                                      SNAP2070
C                                                                       SNAP2080
      WRITE (6,20) IDENT,NPAGE                                          SNAP2090
      NPAGE = NPAGE + 1                                                 SNAP2100
      DO 100 NALT = 1, 201                                              SNAP2110
       READ (5,34) NXX,N1,(CONSP(NALT,J), J = 1, 7)                     SNAP2120
      WRITE (6,35) NXX,N1,(CONSP(NALT,J), J = 1, 7)                     SNAP2130
      IF (NXX .NE. NALT)  GO TO 500                                     SNAP2140
      IF (N1 .NE. 1)  GO TO 500                                         SNAP2150
       READ (5,34) NXX,N2,(CONSP(NALT,J), J = 8, 14)                    SNAP2160
```

```
      WRITE (6,35) NXX,N2,(CONSP(NALT,J), J = 8, 14)                    SNAP2170
      IF (NXX .NE. NALT)  GO TO 500                                     SNAP2180
      IF (N2 .NE. 2)  GO TO 500                                         SNAP2190
       READ (5,34) NXX,N3,(CONSP(NALT,J), J = 15, 20)                   SNAP2200
      WRITE (6,35) NXX,N3,(CONSP(NALT,J), J = 15, 20)                   SNAP2210
      IF (NXX .NE. NALT)  GO TO 500                                     SNAP2220
      IF (N3 .NE. 3)  GO TO 500                                         SNAP2230
      CONSP(NALT,KN) = 0.1*CONSP(NALT,KN)                               SNAP2240
  100 CONTINUE                                                          SNAP2250
C                                                                       SNAP2260
C     INITIALIZE GEOMETRY ROUTINE                                       SNAP2270
C     CONVERT TO RADIAN MEASURE                                         SNAP2280
      DTHETA=RAD(DTHETA)                                                SNAP2290
      DPHI=RAD(DPHI)                                                    SNAP2300
      SOLDEC=RAD(SOLDEC)                                                SNAP2310
       NSPS = 1                                                         SNAP2320
       NSPF = 20                                                        SNAP2330
       NSPM = 20                                                        SNAP2340
       KTEST = 1                                                        SNAP2350
       NBSAV = NBASE                                                    SNAP2360
C                                                                       SNAP2370
C     SNAPS2 ASSUMES DIFFUSIVE EQUILIBRIUM ABOVE ALTITUDE NTOP          SNAP2380
      DO 450 IHR = 1, IHRMAX                                            SNAP2390
       CALL SECOND (SECA)                                               SNAP2400
       TIME = TIME + 0.5*DLTIME                                         SNAP2410
       WRITE (6,20) IDENT,NPAGE                                         SNAP2420
       NPAGE = NPAGE + 1                                                SNAP2430
C                                                                       SNAP2440
       IF (IHR .EQ. 1)  NBASE = 1                                       SNAP2450
       TEMPU = 0.0                                                      SNAP2460
       TEMP = 0.0                                                       SNAP2470
       SHU = 0.0                                                        SNAP2480
       SH = 0.0                                                         SNAP2490
       GU = 0.0                                                         SNAP2500
       G = 0.0                                                          SNAP2510
       RMWTU = 0.0                                                      SNAP2520
```

```
      RMWT = 0.0                                                        SNAP2530
      SUMU = 0.0                                                        SNAP2540
      SUM = 0.0                                                         SNAP2550
      I = 0                                                             SNAP2560
      DO 110 NALT = NBASE, NTOP                                         SNAP2570
      I = I + 1                                                         SNAP2580
      NA = NALT - 1                                                     SNAP2590
      NB = NALT - 2                                                     SNAP2600
      TEMPL = TEMP                                                      SNAP2610
      TEMP = TEMPU                                                      SNAP2620
      SHL = SH                                                          SNAP2630
      SH = SHU                                                          SNAP2640
      GL = G                                                            SNAP2650
      G = GU                                                            SNAP2660
      RMWTL = RMWT                                                      SNAP2670
      RMWT = RMWTU                                                      SNAP2680
      SUML = SUM                                                        SNAP2690
      SUM = SUMU                                                        SNAP2700
      SUMU = 0.0                                                        SNAP2710
      RMWTU = 0.0                                                       SNAP2720
      DO 102 NSP = 1, 20                                                SNAP2730
      SUMU = SUMU + CONSP(NALT,NSP)                                     SNAP2740
      RMWTU = RMWTU + AMWT(NSP)*CONSP(NALT,NSP)                         SNAP2750
  102 CONTINUE                                                          SNAP2760
      RMWTU = RMWTU/SUMU                                                SNAP2770
      ROBS = NA                                                         SNAP2780
      TEMPU = TKEL(ROBS)                                                SNAP2790
      RDTEMP(NALT) = 2.68684E19*(TEMPU/273.15)**0.75                    SNAP2800
      TEMPK(NALT) = TEMPU                                               SNAP2810
      GU = GREF*(REFF/(REFF + ROBS))**2                                 SNAP2820
      SHU = 8.31432E7*TEMPU/(RMWTL*GU)                                  SNAP2830
      IF (I .EQ. 2)  SHFACL = EXP(1.0E5/SHU)                            SNAP2840
      IF (I .LE. 2)  GO TO 110                                          SNAP2850
      IF (NALT .EQ. NTOP - 1)  SHFAC = EXP(-1.0E5/SHU)                  SNAP2860
      DTDZ = (TEMPU - TEMPL)*RDELZ                                      SNAP2870
      DKDZ = (EDDYEF(NALT) - EDDYEF(NB))*RDELZ                          SNAP2880
```

```
      DHDZ = (SHU - SHL)*RDELZ                                                  SNAP2890
      TFAC = DTDZ/TEMP                                                          SNAP2900
      AFAC(NA) = TFAC                                                           SNAP2910
      SFAC = (4.0 - (TEMPU + TEMPL)/TEMP)*R2DELZ + TFAC*TFAC                    SNAP2920
      BFAC(NA) = SFAC                                                           SNAP2930
      FAC = 1.0/SH + TFAC                                                       SNAP2940
      EC = EDCOEF(NA)                                                           SNAP2950
      BTED(NA) = (SFAC + DHDZ/(SH*SH))*ED - FAC*DKDZ                            SNAP2960
      CFAC(NA) = (TEMPU/GU - TEMPL/GL)*G*G*RDELZ/(8.31432E7*TEMP*TEMP)          SNAP2970
      DFAC(NA) = G/(8.31432E7*TEMP)                                             SNAP2980
      FAC = (FAC*ED + DKDZ)*RDELZ                                               SNAP2990
      QJEDU(NA) = R2DELZ*ED + FAC                                               SNAP3000
      QJEDL(NA) = R2DELZ*ED - FAC                                               SNAP3010
      IF (IHR .GT. 1)  GO TO 110                                                SNAP3020
      CONFAC = SUML*(QJEDL(NA)/QJEDU(NA) + 1.0)/((BTED(NA)/QJEDU(NA)            SNAP3030
     1  + (TEMPU - TEMPL)/TEMP + 2.0*DELZ/SH)*SUM)                              SNAP3040
      SUM = CONFAC*SUM                                                          SNAP3050
      IF (NALT .EQ. NTOP)  CON = (BTED(NA)*SUM - QJEDL(NA)*SUML)                SNAP3060
     1  /(QJEDU(NA)*SUMU)                                                       SNAP3070
      DO 104 NSP = 1, NSPM                                                      SNAP3080
      CONSP(NA,NSP) = CONSP(NA,NSP)*CONFAC                                      SNAP3090
      IF (NALT .EQ. NTOP)  CONSP(NTOP,NSP) = CON*CONSP(NTOP,NSP)                SNAP3100
  104 CONTINUE                                                                  SNAP3110
      WRITE (6,6) NB,CONSP(NA,1),CONSP(NA,2),GL,SH,DTDZ,DKDZ,DHDZ,              SNAP3120
     1  BTED(NA),QJEDL(NA),QJEDU(NA)                                            SNAP3130
      WRITE (6,6) N1,RMWT,TFAC,SFAC,CFAC(NA),DFAC(NA),RDTEMP(NA)                SNAP3140
C     WRITE (6,6) NALT,CONC(1),CONC(2),CONC(4),CONC(7),CONC(9),CONC(10)         SNAP3150
  110 CONTINUE                                                                  SNAP3160
      NBASE = NBSAV                                                             SNAP3170
      NBASE = NBASE + 1                                                         SNAP3180
      NTOP = NTOP - 1                                                           SNAP3190
      NMIN = NBASE - 1                                                          SNAP3200
      DO 400 M = 1, MMAX                                                        SNAP3210
      NTIME = (0.5*DLTIME + TIME)/60.0 + 720.1                                  SNAP3220
      RTIME = FLOAT(NTIME)/60.0                                                 SNAP3230
      NTIME = 100*(NTIME/60) + MOD(NTIME,60)                                    SNAP3240
```

```
      IF (MOD(M,NPRNT) .NE. 0)  GO TO 120                                SNAP3250
      WRITE (6,20) IDENT,NPAGE                                           SNAP3260
      NPAGE = NPAGE + 1                                                  SNAP3270
      WRITE (6,6) NTIME,RTIME                                            SNAP3280
      WRITE (6,19)                                                       SNAP3290
C     WRITE (6,22) LALT,LTIME,LCHI,LCOLO2,LCOLN2,LCCLC3,LJC2,LJO3        SNAP3300
      WRITE (6,22) LALT,LQCCHA,LQCCHB,LDZCHA,LDZCHB,(LSP(I),I=1,6)       SNAP3310
C     WRITE (6,22) LML,(LSP(I),I=7,14)                                   SNAP3320
      WRITE (6,19)                                                       SNAP3330
  120 CONTINUE                                                           SNAP3340
      RFAC = RDTEMP(NMIN)                                                SNAP3350
      C1 = CONSP(NMIN,1)                                                 SNAP3360
      C2 = CONSP(NMIN,2)                                                 SNAP3370
      C3 = CONSP(NMIN,3)                                                 SNAP3380
      CMDL(1) = RFAC/(C1/0.180 + C2/0.181 + C3/0.260)                    SNAP3390
      CMDL(2) = RFAC/(C1/0.181 + C2/0.189 + C3/0.260)                    SNAP3400
      CMDL(3) = RFAC/(C1/0.260 + C2/0.260 + C3/0.370)                    SNAP3410
      RFAC = RDTEMP(NBASE)                                               SNAP3420
      C1 = CONSP(NBASE,1)                                                SNAP3430
      C2 = CONSP(NBASE,2)                                                SNAP3440
      C3 = CONSP(NBASE,3)                                                SNAP3450
      CMD(1) = RFAC/(C1/0.180 + C2/0.181 + C3/0.260)                     SNAP3460
      CMD(2) = RFAC/(C1/0.181 + C2/0.189 + C3/0.260)                     SNAP3470
      CMD(3) = RFAC/(C1/0.260 + C2/0.260 + C3/0.370)                     SNAP3480
      DO 130 NSP = 1,NSPM                                                SNAP3490
      SCX(NSP) = CONSP(NBASE-1,NSP)                                      SNAP3500
      IF (NSP .LT. 4) GO TO 130                                          SNAP3510
      CMDL(NSP) = 0.707*CMDL(2)*SQRT(31.9988/AMWT(NSP) + 1.0)            SNAP3520
      CMD(NSP) = 0.707*CMD(2)*SQRT(31.9988/AMWT(NSP) + 1.0)              SNAP3530
  130 CONTINUE                                                           SNAP3540
      DZOHA = 0.0                                                        SNAP3550
      DZOHB = 0.0                                                        SNAP3560
      DO 350 NALT = NBASE, NTOP                                          SNAP3570
      NPLU = NALT + 1                                                    SNAP3580
      NAMIN = NALT - 1                                                   SNAP3590
      RFAC = RDTEMP(NPLU)                                                SNAP3600
```

```
      C1 = CONSP(NPLU,1)                                                SNAP3610
      C2 = CONSP(NPLU,2)                                                SNAP3620
      C3 = CONSP(NPLU,3)                                                SNAP3630
      CMDU(1) = RFAC/(C1/0.180 + C2/0.181 + C3/0.260)                   SNAP3640
      CMDU(2) = RFAC/(C1/0.181 + C2/0.189 + C3/0.260)                   SNAP3650
      CMDU(3) = RFAC/(C1/0.260 + C2/0.260 + C3/0.370)                   SNAP3660
      AFACNA = AFAC(NALT)                                               SNAP3670
      BFACNA = BFAC(NALT)                                               SNAP3680
      CFACNA = CFAC(NALT)                                               SNAP3690
      DFACNA = DFAC(NALT)                                               SNAP3700
      QJEDUN = QJEDU(NALT)                                              SNAP3710
      QJEDLN = QJEDL(NALT)                                              SNAP3720
      BTEDNA = BTED(NALT)                                               SNAP3730
     DO 140 NSP = 1,NSPM                                                SNAP3740
      CONSPN = CONSP(NALT,NSP)                                          SNAP3750
      DCX(NSP) = CONSPN                                                 SNAP3760
      TCX(NSP) = CONSPN                                                 SNAP3770
      CMDNSP = CMD(NSP)                                                 SNAP3780
      IF (NSP .LT. 4) GO TO 135                                         SNAP3790
      CMDU(NSP) = 0.707*CMDU(2)*SQRT(31.9988/AMWT(NSP) + 1.0)           SNAP3800
135   CONTINUE                                                          SNAP3810
      FAC = AMWT(NSP)*CFACNA + AFACNA                                   SNAP3820
      DDDZ = (CMDU(NSP) - CMDL(NSP))*RDELZ                              SNAP3830
      BTEMD(NSP) = (BFACNA + AMWT(NSP)*CFACNA)*CMDNSP - FAC*DDDZ        SNAP3840
     1  + BTEDNA                                                        SNAP3850
      FAC = (FAC*CMDNSP + DDDZ)*RDELZ                                   SNAP3860
      QJEMD(NSP) = (R2DELZ*CMDNSP + FAC + QJEDUN)*CONSP(NPLU,NSP)       SNAP3870
     1  + (R2DELZ*CMDNSP - FAC + QJEDLN)*CONSP(NAMIN,NSP)               SNAP3880
      CMDL(NSP) = CMDNSP                                                SNAP3890
      CMD(NSP) = CMDL(NSP)                                              SNAP3900
140   CONTINUE                                                          SNAP3910
      TEMP = TEMPK(NALT)                                                SNAP3920
      RCBS = FLOAT(NALT - 1)                                            SNAP3930
      CALL SOLLY (DTHETA,ROES,TIME,SOLDEC,CHI,ETA)                      SNAP3940
      IF (ETA .LE. 0.0)  GO TO 160                                      SNAP3950
      CHISAV = 1.0                                                      SNAP3960
```

```
      IF (CHI .GT. 1.0) GO TO 145                                       SNAP3970
      CHISAV = 1.0/COS(CHI)                                             SNAP3980
      CHI = 0.0                                                         SNAP3990
  145 CONTINUE                                                          SNAP4000
      CALL COLUMN (RCES,CHI,COL,ETA)                                    SNAP4010
      COL(1) = CHISAV*COL(1)                                            SNAP4020
      COL(2) = CHISAV*COL(2)                                            SNAP4030
      COL(3) = CHISAV*COL(3)                                            SNAP4040
      COL(4) = CHISAV*COL(4)                                            SNAP4050
  150 CONTINUE                                                          SNAP4060
  160 CONTINUE                                                          SNAP4070
      CALL SUNPHI (COL,ETA,RATE)                                        SNAP4080
      TEMPI = 1.0/TEMP                                                  SNAP4090
      TEMP12 = SQRT(TEMPI)                                              SNAP4100
      TEMP32 = TEMPI*TEMP12                                             SNAP4110
      TEMP52 = TEMPI*TEMP32                                             SNAP4120
C     FORWARD REACTIONS                                                 SNAP4130
C     REVERSE REACTIONS                                                 SNAP4140
C     REACTION  1    N + C2 = NC + C                                    SNAP4150
      R1 = 1.1E-14*TEMP*EXP(-3140.0*TEMPI)                              SNAP4160
      S1 = 2.6E-15*TEMP*EXP(-19500.0*TEMPI)                             SNAP4170
C     REACTION  2    N + NO = N2 + C                                    SNAP4180
      R2 = 5.1E-11*EXP(-168.0*TEMPI)                                    SNAP4190
      S2 = 2.3E-10*EXP(-37500.0*TEMPI)                                  SNAP4200
C     REACTION  3    NO + NO = N2 + C2                                  SNAP4210
      R3 = 1.3*TEMP52*EXP(-42800.0*TEMPI)                               SNAP4220
      S3 = 15.2*TEMP52*EXP(-64300.0*TEMPI)                              SNAP4230
C     REACTION  4    O + C + M = C2 + M                                 SNAP4240
      R4 = 3.80E-30*TEMPI*EXP(-171.0*TEMPI)                             SNAP4250
      S4 = 4.57E-5*TEMPI*EXP(-59700.0*TEMPI)                            SNAP4260
C     REACTION  5    N + N + M = N2 + M                                 SNAP4270
      R5 = 1.4E-33*EXP(-500.0*TEMPI)                                    SNAP4280
      S5 = 3.2E-7*TEMP12*EXP(-112000.0*TEMPI)                           SNAP4290
C     REACTION  6    N + N + N2 = N2 + N2                               SNAP4300
      R6 = 7.3E-32*TEMP12                                               SNAP4310
      S6 = 7.8E-7*TEMP12*EXP(-112000.0*TEMPI)                           SNAP4320
```

```
C      REACTION  7     N + N + N = N2 + N                                SNAP4330
         R7 = 6.25E-27*TEMP32                                           SNAP4340
         S7 = 6.75E-2*TEMP32*EXP(-112000.0*TEMPI)                       SNAP4350
C      REACTION  8     N + O + M = NO + M                                SNAP4360
        R8 = 1.1E-32*SQRT(273.0*TEMPI)                                  SNAP4370
        S8 = 3.8E-7*TEMP12*EXP(-74500.0*TEMPI)                          SNAP4380
C      REACTION  9     NO + O = NO2 + H-NU                               SNAP4390
        R9 = 6.5E-17*TEMP/296.0                                         SNAP4400
        S9 = RATE(4)                                                    SNAP4410
C      REACTION 10     O + NO2 = NO + O2                                 SNAP4420
        R10 = 1.6E-11*EXP(-300.0*TEMPI)                                 SNAP4430
        S10 = 1.0E-37                                                   SNAP4440
C      REACTION 11     N + NO2 = N2 + O2                                 SNAP4450
        R11 = 2.3E-12                                                   SNAP4460
        S11 = 1.0E-37                                                   SNAP4470
C      REACTION 12     N + NO2 = NO + NO                                 SNAP4480
        R12 = 5.9E-12                                                   SNAP4490
        S12 = 1.0E-35                                                   SNAP4500
C      REACTION 13     NO + O + M = NO2 + M                              SNAP4510
        R13 = 2.9E-33*EXP(940.0*TEMPI)                                  SNAP4520
        S13 = 1.8E-8*EXP(-32800.0*TEMPI)                                SNAP4530
C      REACTION 14     NO + NO + O2 = NO2 + NO2                          SNAP4540
        R14 = 6.6E-39*EXP(530.0*TEMPI)                                  SNAP4550
        S14 = 6.6E-12*EXP(-13500.0*TEMPI)                               SNAP4560
C      REACTION 15     O + O = O2 + H-NU                                 SNAP4570
        S15 = RATE(1)                                                   SNAP4580
C      REACTION 16     N + O = NO + H-NU                                 SNAP4590
        S16 = RATE(3)                                                   SNAP4600
C      REACTION 21   NO + O3 = O2 + NO2                                  SNAP4610
        R21 = 9.5E-13*EXP(-1240.0*TEMPI)                                SNAP4620
        S21 = 1.0E-13*EXP(-23900.0*TEMPI)                               SNAP4630
      REACTION 31    O + O2 + M = O3 + M                                SNAP4640
      R31   = 4.63E-35* EXP( 1060.00 *TEMPI)                            SNAP4650
      S31   = 1.65E-9 * EXP(-11920.0 *TEMPI)                            SNAP4660
      REACTION 32    O + O3 = O2 + O2                                   SNAP4670
      R32   = 2.00E-11* EXP(-2410.0 *TEMPI)                             SNAP4680
```

```
      S32 = 2.12E-11*EXP(-50700.0 * TEMPI)                              SNAP4690
      REACTION 33   H + O3 = OH + O2                                    SNAP4700
      R33  = 2.6E-11                                                    SNAP4710
      S33  = 0.                                                         SNAP4720
      REACTION 34   O + CH = H + O2                                     SNAP4730
      R34  = 2.06E-11                                                   SNAP4740
      S34  = 4.19E-10 * EXP(-8450.0*TEMPI)                              SNAP4750
C       R35  HO2 + O = OH + O2                                          SNAP4760
      R35 = 1.0E-11                                                     SNAP4770
C       R36  HO2 + H = OH + OH                                          SNAP4780
      R36 = 2.0E-10*EXP(-2000.0*TEMPI)                                  SNAP4790
C       R37  H + O2 + M = HO2 + M                                       SNAP4800
      R37 = 3.0E-32*(273.0*TEMPI)**1.3                                  SNAP4810
C       R38  OH + O3 = HO2 + O2                                         SNAP4820
      R38 = 5.0E-13                                                     SNAP4830
C       R39  HO2 + O3 = OH + O2 + O2                                    SNAP4840
      R39 = 1.0E-14                                                     SNAP4850
C       R40  OH + OH = H2O + O                                          SNAP4860
      R40 = 9.3E-12*EXP(-390.0*TEMPI)                                   SNAP4870
      S40 = 9.3E-11*EXP(-9000.0*TEMPI)                                  SNAP4880
C       R41  OH + HO2 = H2O + O2                                        SNAP4890
      R41 = 1.0E-11                                                     SNAP4900
C       R42  H + HO2 = H2 + O2                                          SNAP4910
      R42 = 2.0E-13                                                     SNAP4920
C       R43  O + H2 = OH + H                                            SNAP4930
      R43 = 7.0E-11*EXP(-5100.0*TEMPI)                                  SNAP4940
C       R44  HO2 + HO2 = H2O2 + O2                                      SNAP4950
      R44 = 1.5 E-12                                                    SNAP4960
C       R45  OH + H2O2 = H2O + HO2                                      SNAP4970
      R45 = 4.0E-13                                                     SNAP4980
C       R46  O + H2O2 = OH + HO2                                        SNAP4990
      R46 = 1.0E-15                                                     SNAP5000
C       R47  H + H2O2 = H2 + HO2                                        SNAP5010
      R47 = 1.0E-13                                                     SNAP5020
C       R50  O(1D) + O3 = O2 + O2                                       SNAP5030
      R50 = 5.0E-11                                                     SNAP5040
```

```
C        R51  O(1D) + N2 = O + N2                                    SNAP5050
         R51 = 5.0E-11                                               SNAP5060
C        R52  O(1D) + H2 = OH + H                                    SNAP5070
         R52 = 1.0E-11                                               SNAP5080
C        R53  O(1D) + H2O = OH + OH                                  SNAP5090
         R53 = 3.0E-11                                               SNAP5100
C        R54  O(1D) + CH4 = PRODUCTS                                 SNAP5110
         R54 = 1.0E-11                                               SNAP5120
C        R55  CH4 + OH = H2O + CH3                                   SNAP5130
         R55 = 1.2E-10*EXP(-2960.0*TEMPI)                            SNAP5140
C        R56  O + CH4 = CH + CH3                                     SNAP5150
         R56 = 3.5E-11*EXP(-4550.0*TEMPI)                            SNAP5160
C        R60  CO + OH = CO2 + H                                      SNAP5170
         R60 = 1.0E-12*EXP(-540.0*TEMPI)                             SNAP5180
C        R61  CO + O + M = CO2 + M                                   SNAP5190
         R61 = 1.4E-33*EXP(-1250.0*TEMPI)                            SNAP5200
C       IONIC REACTION 1   O2+ + E = O + O                           SNAP5210
-        RI1 = 6.0E-5*TEMPI                                          SNAP5220
         SI1 = 0.0                                                   SNAP5230
C       IONIC REACTION 2   N2+ + E = N + N                           SNAP5240
         RI2 = 9.0E-5*TEMPI                                          SNAP5250
         SI2 = 0.0                                                   SNAP5260
C       IONIC REACTION 3   N2+ + O2 = N2 + O2+                       SNAP5270
         RI3 = 1.0E-10                                               SNAP5280
         SI3 = 0.0                                                   SNAP5290
C       IONIC REACTION 4   N2+ + E = N2 + H-NU                       SNAP5300
         RI4 = 1.0E-16                                               SNAP5310
         SI4 = 1.0E-16                                               SNAP5320
C        WRITE (6,19)                                                SNAP5330
         NA = NALT - 1                                               SNAP5340
C                                                                    SNAP5350
C                                                                    SNAP5360
  170  CONTINUE                                                      SNAP5370
C                                                                    SNAP5380
C                                                                    SNAP5390
      DO 300 ML = 1, MLMAX                                           SNAP5400
```

```
      MLSAVE = ML                                                        SNAP5410
      MMF = (MOD(ML + 1,2) + 2)*2**((ML - 1)/2)                          SNAP5420
      IF (MLMAX .EQ. 1)  MMF = 1                                         SNAP5430
     DO 180 NSP = 1,NSPM                                                 SNAP5440
      DCX(NSP) = CONSP(NALT,NSP)                                         SNAP5450
  180 CONTINUE                                                           SNAP5460
     DO 290 MMS = 1,MMF                                                  SNAP5470
     DO 190 NSP = 1,NSPM                                                 SNAP5480
      TCX(NSP) = DCX(NSP)                                                SNAP5490
  190 CONTINUE                                                           SNAP5500
      DTIME = 0.5*DLTIME/FLOAT(MMF)                                      SNAP5510
      ITER = 1                                                           SNAP5520
  200 CONTINUE                                                           SNAP5530
      XM4 = TN2/3.0 + TO2 + 3.0*TO                                       SNAP5540
      XM5 = TO2 + TNO + TO                                               SNAP5550
      XM8 = XM4                                                          SNAP5560
      XM13 = XM4 - TNO + TNO2                                            SNAP5570
      XM31 = 0.39*TN2 + 0.44*TO2                                         SNAP5580
      XM37 = TN2 + TO2                                                   SNAP5590
      XM61 = XM37 + TO                                                   SNAP5600
     DO 255 NSP = 1, NSPM                                                SNAP5610
      DCT = DCX(NSP)                                                     SNAP5620
      GO TO (201,202,203,204,205,206,207,208,209,210,211,212,213,214,    SNAP5630
     1  215,216,217,218,219,220,221,222,223,224,225), NSP                SNAP5640
C                                                                        SNAP5650
C       MOLECULAR NITROGEN                                               SNAP5660
  201 CONTINUE                                                           SNAP5670
      RCX(KN2) = DCT                                                     SNAP5680
      APT = 0.0                                                          SNAP5690
      BTT = 0.0                                                          SNAP5700
      GJT = 0.0                                                          SNAP5710
      GO TO 251                                                          SNAP5720
C                                                                        SNAP5730
C       MOLECULAR OXYGEN                                                 SNAP5740
  202 CONTINUE                                                           SNAP5750
      APT = 2.0*S32                                                      SNAP5760
```

```
      BTT = S4*XM4 + R31*XM31*TO + (S34 + R37*XM37)*TH + RATE(1)        SNAP5770
      QJT = ((R32*TO + R39*THO2 + R50*TO1D)*2.0 + RATE(5)               SNAP5780
     1  + S31*XM31 + R33*TH + R38*TCH)*TO3                             SNAP5790
     2  + (R4*XM4*TO + R34*TCH + R35*THO2)*TO                          SNAP5800
     3  + (R41*TOH + R42*TH + R44*THO2)*THO2                           SNAP5810
      GO TO 251                                                         SNAP5820
C                                                                       SNAP5830
C       ATOMIC OXYGEN, O(3P)                                            SNAP5840
  203 CONTINUE                                                          SNAP5850
      APT = 2.0*R4*XM4                                                  SNAP5860
      BTT = R31*XM31*TO2 + R32*TO3 + R34*TCH + R35*THO2                 SNAP5870
     1  + R43*TH2 + R46*TH2O2 + R56*TCH4 + R61*XM61*TCO                 SNAP5880
      QJT = (2.0*S4*XM4 + S32*TO2 + S34*TH + RATE(8))*TO2               SNAP5890
     1  + (S31*XM31 + RATE(10))*TO3 + R40*TOH*TOH + R51*TN2*TO1D        SNAP5900
      GO TO 251                                                         SNAP5910
C                                                                       SNAP5920
C       OZONE                                                           SNAP5930
  204 CONTINUE                                                          SNAP5940
      APT = 0.0                                                         SNAP5950
      BTT = S31*XM31 + R32*TO + R33*TH + R38*TOH + R39*THO2             SNAP5960
     1  + R50*TO1D + RATE(5)                                            SNAP5970
      QJT = (R31*XM31*TO + S32*TO2)*TO2                                 SNAP5980
      GO TO 251                                                         SNAP5990
C                                                                       SNAP6000
C       HYDROXYL                                                        SNAP6010
  205 CONTINUE                                                          SNAP6020
      APT = 2.0*R40                                                     SNAP6030
      BTT = R34*TO + R38*TO3 + R41*THO2 + R45*TH2O2 + R60*TCO           SNAP6040
     1  + R55*TCH4                                                      SNAP6050
      QJT = (R35*THO2 + R43*TH2 + R46*TH2O2 + R56*TCH4)*TO              SNAP6060
     1  + (R52*TH2 + 2.0*R53*TH2O + R54*TCH4)*TO1D + R39*THO2*TO3       SNAP6070
     2  + (R33*TO3 + S34*TO2 + 2.0*R36*THO2)*TH + 2.0E-5*RATE(5)*TH2O2  SNAP6080
     3  + RATE(6)*TH2O                                                 SNAP6090
      GO TO 251                                                         SNAP6100
C                                                                       SNAP6110
C       ATOMIC HYDROGEN                                                 SNAP6120
```

```
  206  CONTINUE                                                          SNAPE130
       APT = 0.0                                                         SNAPE140
       BTT = (S34 + R37*XM37)*TO2 + (R42 + R36)*THO2 + R33*TO3           SNAPE150
      1  + R47*TH2O2                                                    SNAPE160
       QJT = (R34*TOH + R43*TH2)*TC + R52*TC1D*TH2 + R60*TCH*TCC         SNAPE170
      1  + RATE(6)*TH2O + RATE(7)*TCH4                                  SNAPE180
       GO TO 251                                                        SNAPE190
C                                                                       SNAPE200
C       WATER VAPOR                                                     SNAPE210
  207  CONTINUE                                                          SNAPE220
       APT = 0.0                                                        SNAPE230
       BTT = R53*TC1C + RATE(6)                                         SNAPE240
       QJT = (R40*TCH + R41*THO2 + R45*TH2C2 + R55*TCH4)*TCH            SNAPE250
       GO TO 251                                                        SNAPE260
C                                                                       SNAPE270
C       PERHYDROXYL                                                     SNAPE280
  208  CONTINUE                                                          SNAPE290
       APT = 2.0*R44                                                    SNAPE300
       BTT = R35*TC + R39*TO3 + R41*TOH + (R36 + R42)*TH                SNAPE310
       QJT = (R45*TCH + R46*TO + R47*TH)*TH2O2 + R38*TCH*TC3            SNAPE320
      1  + R37*XM37*TH*TC2                                              SNAPE330
       GO TO 251                                                        SNAPE340
C                                                                       SNAPE350
C       HYDROGEN PEROXIDE                                               SNAPE360
  209  CONTINUE                                                          SNAPE370
       APT = 0.0                                                        SNAPE380
       BTT = R45*TCH + R46*TC + R47*TH + 1.0E-2*RATE(5)                 SNAPE390
       QJT = R44*THC2*THC2                                              SNAPE400
       GO TO 251                                                        SNAPE410
C                                                                       SNAPE420
C       MOLECULAR HYDROGEN                                              SNAPE430
  210  CONTINUE                                                          SNAPE440
       APT = 0.0                                                        SNAPE450
       BTT = R43*TC + R52*TO1D                                          SNAPE460
       QJT = (R42*THC2 + R47*TH2O2)*TH                                  SNAPE470
       GO TO 251                                                        SNAPE480
```

```
C
C        METHANE                                                        SNAP6490
C                                                                       SNAP6500
  211   CONTINUE                                                        SNAP6510
        APT = 0.0                                                       SNAP6520
        BTT = RATE(7) + R54*TO1D + R55*TOH + R56*TO                     SNAP6530
        QJT = 0.0                                                       SNAP6540
        GO TO 251                                                       SNAP6550
C                                                                       SNAP6560
C        CARBON DIOXIDE                                                 SNAP6570
C                                                                       SNAP6580
  212   CONTINUE                                                        SNAP6590
        APT = 0.0                                                       SNAP6600
        BTT = RATE(2) + 1.0E-9*ETA*TIO(NALT)                            SNAP6610
        XXX12 = R60*TOH + R61*XM61*TO                                   SNAP6620
        QJT = XXX12*TCO                                                 SNAP6630
        GO TO 251                                                       SNAP6640
C                                                                       SNAP6650
C        CARBON MONOXIDE                                                SNAP6660
C                                                                       SNAP6670
  213   CONTINUE                                                        SNAP6680
        APT = 0.0                                                       SNAP6690
        BTT = XXX12                                                     SNAP6700
        QJT = (RATE(2) + 1.0E-9*ETA*TIO(NALT))*TCO2                     SNAP6710
        GO TO 251                                                       SNAP6720
C                                                                       SNAP6730
C        ATOMIC OXYGEN, O(1D)                                           SNAP6740
C                                                                       SNAP6750
  214   CONTINUE                                                        SNAP6760
        APT = 0.0                                                       SNAP6770
        BTT = R50*TC3 + R51*TN2 + R52*TH2 + R53*TH2O + R54*TCH4         SNAP6780
        QJT = RATE(2)*TCO2 + RATE(9)*TO2 + RATE(11)*TC3                 SNAP6790
        GO TO 251                                                       SNAP6800
C                                                                       SNAP6810
C        ATOMIC NITROGEN, N(4S)                                         SNAP6820
C                                                                       SNAP6830
  215   CONTINUE                                                        SNAP6840
        QJT = ((S5*XM5 + S6*TN2 + S7*TN)*2.0 + S2*TO                    SNAP6850
     1   + S11*TO2)*TN2 + (S1*TO + S8*XM8 + S12*TNO + S10)*TNO          SNAP6860
     3 + 1.0E-14*TN2*TN2O                                               SNAP6870
        BTT = (R11 + R12)*TNO2 + R8*XM8*TO + R2*TNO + R1*TO2            SNAP6880
```

```
      APT = (R5*XM5 + R6*TN2 + R7*TN)*2.0                                SNAP6850
      GO TO 251                                                          SNAP6860
C                                                                        SNAP6870
C       NITRIC OXIDE                                                     SNAP6880
  216 CONTINUE                                                           SNAP6890
      QJT = ((R12*TN + S14*TNO2)*2.0 + S9 + R10*TO                       SNAP6900
     1  + S13*XM13)*TNO2 + (R1*TO2 + R8*XM8*TO)*TN                        SNAP6910
     2  + (S2*TO + 2.0*S3*TO2)*TN2                                        SNAP6920
     3  + 5.0E-12*TN2O*TO2                                              SNAP6930
      BTT = (R13*XM13 + R9 + S1)*TO + R2*TN + R21*TO3                    SNAP6940
     1  + S8*XM8 + S10*TO2 + S16                                          SNAP6950
     2  + 0.1*RATE(7)                                                   SNAP6960
      APT = (R14*TO2 + S12 + R3)*2.0                                     SNAP6970
      GO TO 251                                                          SNAP6980
C                                                                        SNAP6990
C       NITROGEN DIOXIDE                                                 SNAP7000
  217 CONTINUE                                                           SNAP7010
      QJT = ((2.0*R14*TNO + R13*XM13 + R9)*TO + R21*TO3                  SNAP7020
     1  + S12*TNO + S10*TO2)*TNO + S11*TN2*TO2                           SNAP7030
      BTT = (R11 + R12)*TN + S9 + R10*TO + S13*XM13                      SNAP7040
      APT = 0.0                                                          SNAP7050
      GO TO 251                                                          SNAP7060
C                                                                        SNAP7070
C       NITROUS OXIDE                                                    SNAP7080
  218 CONTINUE                                                           SNAP7090
      QJT = 0.0                                                          SNAP7100
      BTT = 0.0                                                          SNAP7110
      APT = 0.0                                                          SNAP7120
      GO TO 251                                                          SNAP7130
C                                                                        SNAP7140
C       ATOMIC NITROGEN, N(2D)                                           SNAP7150
  219 CONTINUE                                                           SNAP7160
      QJT = 1.5E-4*TEMPI*ETA*TINC(NALT)*TETOT(NALT)                      SNAP7170
      BTT = 1.0E-14*TN2 + 5.0E-12*TO2                                    SNAP7180
      APT = 0.0                                                          SNAP7190
      GO TO 251                                                          SNAP7200
```

```
C                                                                               SNAP7210
  220  CONTINUE                                                                 SNAP7220
       QJT = 0.0                                                                SNAP7230
       BTT = 0.0                                                                SNAP7240
       APT = 0.0                                                                SNAP7250
       GO TO 251                                                                SNAP7260
C                                                                               SNAP7270
  221  CONTINUE                                                                 SNAP7280
       GO TO 253                                                                SNAP7290
  222  CONTINUE                                                                 SNAP7300
       GO TO 253                                                                SNAP7310
  223  CONTINUE                                                                 SNAP7320
       GO TO 253                                                                SNAP7330
  224  CONTINUE                                                                 SNAP7340
       GO TO 253                                                                SNAP7350
  225  CONTINUE                                                                 SNAP7360
       GO TO 253                                                                SNAP7370
  251  CONTINUE                                                                 SNAP7380
       BTT = BTT + BTEMD(NSP)                                                   SNAP7390
       QJT = QJT + QJEMD(NSP)                                                   SNAP7400
       IF (APT .LE. 0.0)  GO TO 252                                             SNAP7410
       FAC4AQ = 4.0*APT*QJT                                                     SNAP7420
       BTSQ = BTT*BTT                                                           SNAP7430
       FACRT = FAC4AQ/BTSQ                                                      SNAP7440
       DLT = SQRT(FAC4AQ + BTSQ)                                                SNAP7450
       FACDMB = DLT - BTT                                                       SNAP7460
       IF (FACRT .LT. 0.01)  FACDMB = (((((0.027344 - 0.0205*FACRT)             SNAP7470
     1  *FACRT - 0.0390625)*FACRT + 0.0625)*FACRT - 0.125)*FACRT                 SNAP7480
     2  + 0.5)*FACRT*BTT                                                         SNAP7490
       FAC2AX = 2.0*APT*QCT                                                     SNAP7500
       FACB = DLT*DTIME                                                         SNAP7510
       FAC = EXP(AMAX1(-675.0,AMIN1(-FACB,675.0)))                              SNAP7520
       FACM = 1.0 - FAC                                                         SNAP7530
       IF (FACB .LT. 0.01)  FACM = EXPM(FACB)                                   SNAP7540
       RCX(NSP) = (((BTT + DLT)*FAC + FACDMB)*QCT + 2.0*QJT*FACM)               SNAP7550
     1  /((FAC2AX + BTT)*FACM + (FAC + 1.0)*DLT)                                SNAP7560
```

```
      GO TO 253                                                          SNAP7570
252   CONTINUE                                                           SNAP7580
      FACB = BTT*DTIME                                                   SNAP7590
      FAC = EXP(AMAX1(-675.0,AMIN1(-FACB,675.0)))                        SNAP7600
      FACM = 1.0 - FAC                                                   SNAP7610
      IF (FACB .LT. 0.01)  FACM = EXPM(FACB)                             SNAP7620
      RCX(NSP) = DCT*FAC + FACM*QJT/BTT                                  SNAP7630
253   CONTINUE                                                           SNAP7640
255   CONTINUE                                                           SNAP7650
      ITER = ITER + 1                                                    SNAP7660
      IF (ITER .GT. 2)  GO TO 260                                        SNAP7670
     DO 258 NSP = 1, NSPM                                                SNAP7680
      TCX(NSP) = RCX(NSP)                                                SNAP7690
258   CONTINUE                                                           SNAP7700
      DTIME = 2.0*DTIME                                                  SNAP7710
      IF (ITER .EQ. 2) GO TO 200                                         SNAP7720
260   CONTINUE                                                           SNAP7730
270   CONTINUE                                                           SNAP7740
     DO 280 NSP = 1,NSPM                                                 SNAP7750
      DCX(NSP) = ABS(RCX(NSP))                                           SNAP7760
280   CONTINUE                                                           SNAP7770
290   CONTINUE                                                           SNAP7780
      IF (MLMAX .EQ. 1)  GO TO 310                                       SNAP7790
      CALL ROMTST (ML,NSPM,DCX,TST,TCX,KTEST,EPS)                        SNAP7800
      IF(KTEST .EQ. 0) GO TO 310                                         SNAP7810
300   CONTINUE                                                           SNAP7820
310   CONTINUE                                                           SNAP7830
      ML = 100*KTEST + MLSAVE                                            SNAP7840
      MLPRNT(NALT) = ML                                                  SNAP7850
      IF (KTEST .EQ. 0)  GO TO 318                                       SNAP7860
     DO 315 NSP = 1, NSPM                                                SNAP7870
      TCX(NSP) = RCX(NSP)                                                SNAP7880
315   CONTINUE                                                           SNAP7890
318   CONTINUE                                                           SNAP7900
     DO 320 NSP = 1,NSPM                                                 SNAP7910
      CONSP(NAMIN,NSP) = SCX(NSP)                                        SNAP7920
```

```
      SCX(NSP) = ABS(TCX(NSP))                                          SNAP7930
      IF(NALT .EQ. NTOP) CONSP(NALT,NSP) = SCX(NSP)                     SNAP7940
      IF (NALT .EQ. NTOP)                                               SNAP7950
     1  CONSP(NPLU,NSP) = SCX(NSP)*EXP(-1.0E5*AMWT(NSP)*CFACNA)          SNAP7960
  320 CONTINUE                                                          SNAP7970
      QCOHA = R33*TCX(KH)*TCX(KO3)                                      SNAP7980
      QCOHB = R35*TCX(KHO2)*TCX(KC)                                     SNAP7990
      DZOHA = DZOHA + 1.0E5*QCOHA                                       SNAP8000
      DZOHB = DZOHB + 1.0E5*QCOHB                                       SNAP8010
      IF (MOD(M,NPRNT) .NE. 0)  GO TO 350                               SNAP8020
      WRITE (6,6) NA,QCOHA,QCOHB,DZOHA,DZOHB,(TCX(I),I=1,6)             SNAP8030
      WRITE (6,6) ML,(TCX(I),I=7,16)                                    SNAP8040
  350 CONTINUE                                                          SNAP8050
      IF (MOD(M,NPRNT) .EQ. 0)                                          SNAP8060
     1  WRITE (6,30) (MLPRNT(I), I = NBASE, NTOP)                       SNAP8070
      TIME = TIME + DLTIME                                              SNAP8080
      IF (TIME .GT. 4.32E4)  TIME = TIME - 8.64E4                       SNAP8090
  400 CONTINUE                                                          SNAP8100
      TIME = TIME - 0.5*DLTIME                                          SNAP8110
      WRITE (4,15) TIME,CARD                                            SNAP8120
      NB = NBASE - 1                                                    SNAP8130
      NT = NTOP + 1                                                     SNAP8140
      DO 420 NALT =  1, NT                                              SNAP8150
      WRITE (4,34) NALT,N1,(CONSP(NALT,I), I = 1, 7)                    SNAP8160
      WRITE (4,34) NALT,N2,(CONSP(NALT,I), I = 8, 14)                   SNAP8170
      WRITE (4,34) NALT,N3,(CONSP(NALT,I), I = 15, 20)                  SNAP8180
  420 CONTINUE                                                          SNAP8190
      IS = 1                                                            SNAP8200
      DO 440 IP = 1, 2                                                  SNAP8210
      IF = IS + 9                                                       SNAP8220
      NS = 1                                                            SNAP8230
      DO 435 IX = 1, 5                                                  SNAP8240
      NF = NS + 50                                                      SNAP8250
      WRITE (6,20) IDENT,NPAGE                                          SNAP8260
      NPAGE = NPAGE + 1                                                 SNAP8270
      WRITE (6,6) NTIME,RTIME                                           SNAP8280
```

```
      WRITE (6,19)                                                      SNAP8290
      WRITE (6,22) LALT,(LSF(I), I = IS, IF)                            SNAP8300
      WRITE (6,19)                                                      SNAP8310
     DO 430 NALT = NS, NF                                               SNAP8320
      NA = NALT - 1                                                     SNAP8330
      WRITE (6,6) NA,(CONSP(NALT,I), I = IS, IF)                        SNAP8340
  430 CONTINUE                                                          SNAP8350
      NS = NF                                                           SNAP8360
  435 CONTINUE                                                          SNAP8370
      IS = 11                                                           SNAP8380
  440 CONTINUE                                                          SNAP8390
      NBASE = NBASE - 1                                                 SNAP8400
      NTOP = NTOP + 1                                                   SNAP8410
      CALL SECOND (SECB)                                                SNAP8420
      SECC = SECB - SECA                                                SNAP8430
      WRITE (NW,TFMT) SECA,SECB,SECC                                    SNAP8440
  450 CONTINUE                                                          SNAP8450
      WRITE (6,20) IDENT,NPAGE                                          SNAP8460
      NPAGE = NPAGE + 1                                                 SNAP8470
  500 CONTINUE                                                          SNAP8480
      WRITE (6,20) IDENT,NPAGE                                          SNAP8490
      NPAGE = NPAGE + 1                                                 SNAP8500
     STOP                                                               SNAP8510
    6 FORMAT (1X,I6,1P10E12.3)                                          SNAP8520
   14 FORMAT (I6,1P8E9.3)                                               SNAP8530
   15 FORMAT (1X,F11.4,11A6,A2)                                         SNAP8540
   16 FORMAT (1X,I11,11A6,A2)                                           SNAP8550
   19 FORMAT (1X)                                                       SNAP8560
   20 FORMAT (50H1****** SNAP S2 CHEMISTRY ****** HARF  DEVELOPMENT,    SNAP8570
     1  8H ****** ,10A6,13H ****** PAGE ,I3/)                          SNAP8580
   21 FORMAT (10A6,4A5)                                                 SNAP8590
   22 FORMAT (1X,A6,10(3X,A6,3X))                                       SNAP8600
   24 FORMAT (I6,3X,6(1PE9.3,1H,),3X,A3,I3,I2)                          SNAP8610
   26 FORMAT (7X,1P6E18.9)                                              SNAP8620
   30 FORMAT (/(1X,30I4))                                               SNAP8630
   34 FORMAT (I4,I2,1X,1P7E10.4)                                        SNAP8640
   35 FORMAT (5X,I4,I2,1X,1P7E12.4)                                     SNAP8650
      END                                                               SNAP8660
```

```
      SUBROUTINE COLUMN (HA,CHI,COL,ETA)                                CLMN  10
C     COMPUTES THE INTEGRATED COLUMNAR NUMBER DENSITIES FOR             CLMN  20
C     COL(1)              THE COLUMNAR NUMBER DENSITY FOR O2            CLMN  30
C     COL(2)              THE COLUMNAR NUMBER DENSITY FOR N2            CLMN  40
C     COL(3)              THE COLUMNAR NUMBER DENSITY FOR O3            CLMN  50
C     ALONG A PATH  BEGINNING AT ALTITUDE HA AND EXTENDING TO INFINITY  CLMN  60
C     AND MAKING AN ANGLE CHI(RADIANS) WITH RESPECT TO THE LOCAL        CLMN  70
C     VERTICAL AT HA                                                    CLMN  80
C         CHI=SOLAR ZENITH ANGLE                                        CLMN  90
      COMMON CONSP(501,25)                                              CLMN 100
      DIMENSION CONC(20),COL(5),TERM(5)                                 CLMN 110
      DATA RE /6371.0/                                                  CLMN 120
      DATA A23,A43 / 0.6666666666667, 1.3333333333333 /                 CLMN 130
      RHA = RE + HA                                                     CLMN 140
      RMIN = RHA*SIN(CHI)                                               CLMN 150
      IF (RMIN .LE. RE)  RMIN = RE + 0.0001                             CLMN 160
      RMINSQ = RMIN*RMIN                                                CLMN 170
      COSCHI = COS(CHI)                                                 CLMN 180
      HMIN = ABS(RHA*COSCHI)                                            CLMN 190
      NPT = HA + 1.001                                                  CLMN 200
      DO 110 I = 1,4                                                    CLMN 210
      TERM(I) = 0.0                                                     CLMN 220
  110 CONTINUE                                                          CLMN 230
      ISW = 1                                                           CLMN 240
      R1 = RHA + 1.0                                                    CLMN 250
      HSUB = HMIN                                                       CLMN 260
      IF (COSCHI .GE. 0.0) GO TO 215                                    CLMN 270
      ISW = 0                                                           CLMN 280
      LPT = RMIN + 1.0 - RE                                             CLMN 290
      R1 = FLOAT(LPT) + RE                                              CLMN 300
      H1 = SQRT(R1*R1 - RMINSQ)                                         CLMN 310
      HSUB = H1                                                         CLMN 320
      R1 = R1 + 1.0                                                     CLMN 330
      NA = LPT + 1                                                      CLMN 340
      CALL CONCEN(RMIN - RE,CONC,ETA)                                   CLMN 350
      IF (MOD(NPT-LPT,2) .EQ. 0) GO TO 130                              CLMN 360
```

```
      H13 = H1/3.0                                                          CLMN 370
     DO 120 I = 1,4                                                         CLMN 380
      IF (I .EQ. 3) GO TO 120                                               CLMN 390
      TERM(I) = (2.0*CONC(I) + CONSP(NA,I))*H13                             CLMN 400
120   CONTINUE                                                              CLMN 410
      NS = LPT + 1                                                          CLMN 420
      GO TO 150                                                             CLMN 430
130   CONTINUE                                                              CLMN 440
      H2 = SQRT(R1*R1 - RMINSQ) - HSUB                                      CLMN 450
      HSUB = HSUB + H2                                                      CLMN 460
      R1 = R1 + 1.0                                                         CLMN 470
      H12 = 0.5*H1/H2                                                       CLMN 480
      H21 = 0.5*H2/H1                                                       CLMN 490
      HS3 = (H1 + H2)/3.0                                                   CLMN 500
      NB = NA + 1                                                           CLMN 510
     DO 140 I = 1,4                                                         CLMN 520
      IF (I .EQ. 3) GO TO 140                                               CLMN 530
      YA = CONC(I)                                                          CLMN 540
      YB = CONSP(NA,I)                                                      CLMN 550
      YC = CONSP(NB,I)                                                      CLMN 560
      TERM(I) = ((YB - YC)*H12 + (YB - YA)*H21 + YA + YB + YC)*HS3          CLMN 570
140   CONTINUE                                                              CLMN 580
      NS = LPT + 2                                                          CLMN 590
150   CONTINUE                                                              CLMN 600
      IF (NS .EQ. NPT) GO TO 205                                            CLMN 610
      NF = NPT - 2                                                          CLMN 620
170   CONTINUE                                                              CLMN 630
     DO 200 NA = NS, NF, 2                                                  CLMN 640
      NB = NA + 1                                                           CLMN 650
      NC = NB + 1                                                           CLMN 660
      H1 = SQRT(R1*R1 - RMINSQ) - HSUB                                      CLMN 670
      R1 = R1 + 1.0                                                         CLMN 680
      HSUB = HSUB + H1                                                      CLMN 690
      H2 = SQRT(R1*R1 - RMINSQ) - HSUB                                      CLMN 700
      R1 = R1 + 1.0                                                         CLMN 710
      HSUB = HSUB + H2                                                      CLMN 720
```

```
      H12 = 0.5*H1/H2                                                    CLMN 730
      H21 = 0.5*H2/H1                                                    CLMN 740
      HS3 = (H1 + H2)/3.0                                                CLMN 750
     DO 180 I = 1,4                                                      CLMN 760
      IF (I .EQ. 3) GO TO 180                                            CLMN 770
      YA = CONSP(NA,I)                                                   CLMN 780
      YB = CONSP(NB,I)                                                   CLMN 790
      YC = CONSP(NC,I)                                                   CLMN 800
      TERM(I) = ((YB - YC)*H12 + (YB - YA)*H21 + YA + YB + YC)*HS3       CLMN 810
     1  + TERM(I)                                                       CLMN 820
  180 CONTINUE                                                           CLMN 830
  200 CONTINUE                                                           CLMN 840
      IF (ISW .EQ. 1) GO TO 220                                          CLMN 850
  205 CONTINUE                                                           CLMN 860
      ISW = 1                                                            CLMN 870
     DO 210 I = 1, 4                                                     CLMN 880
      TERM(I) = 2.0*TERM(I)                                              CLMN 890
  210 CONTINUE                                                           CLMN 900
  215 CONTINUE                                                           CLMN 910
      NS = NPT                                                           CLMN 920
      NF = MIN0(NPT + 38,198 + MOD(NPT,2))                               CLMN 930
      IF (NPT .GT. 193) GO TO 220                                        CLMN 940
      IF (CHI .EQ. 0.0) GO TO 250                                        CLMN 950
      GO TO 170                                                          CLMN 960
  220 CONTINUE                                                           CLMN 970
      COL(1) = 1.0E5*TERM(2)                                             CLMN 980
      COL(2) = 1.0E5*TERM(1)                                             CLMN 990
      COL(3) = 1.0E5*TERM(4)                                             CLMN1000
      RETURN                                                             CLMN1010
  250 CONTINUE                                                           CLMN1020
     DO 260 I = 1, 4                                                     CLMN1030
      TERM(I) = 0.333333333333*CONSP(NPT,I)                              CLMN1040
  260 CONTINUE                                                           CLMN1050
     DO 280 NA = NS, NF, 2                                               CLMN1060
      NB = NA + 1                                                        CLMN1070
      NC = NB + 1                                                        CLMN1080
```

```
      DO 270 I = 1, 4                                                    CLMN1090
       IF (I .EQ. 3) GO TO 270                                           CLMN1100
       TERM(I) = A43*CONSP(NB,I) + A23*CONSP(NC,I) + TERM(I)             CLMN1110
  270  CONTINUE                                                          CLMN1120
  280  CONTINUE                                                          CLMN1130
       GO TO 220                                                         CLMN1140
       END                                                               CLMN1150
```

```
      SUBROUTINE CONCEN(H,CONC,ETA)                                      CONC  10
C                                                                        CONC  20
C                                                                        CONC  30
C     PROVIDES AMBIENT CONCENTRATIONS AS A FUNCTION OF ALTITUDE          CONC  40
C     CONC(1)= NO.DENSITY OF N2  IN PARTICLES/CC                         CONC  50
C     CONC(2)= NO.DENSITY OF O2  IN PARTICLES/CC                         CONC  60
C     CONC(4)= NO.DENSITY OF O   IN PARTICLES/CC                         CONC  70
C     CONC(7)= NO.DENSITY OF O3  IN PARTICLES/CC                         CONC  80
C                                                                        CONC  90
C     THIS VERSION (1 MARCH 1971) IS FOR SNAPS2 ONLY.                    CONC 100
C                                                                        CONC 110
      COMMON CONSP(501,25)                                               CONC 120
C                                                                        CONC 130
      DIMENSION CONC(25)                                                 CONC 140
C                                                                        CONC 150
      DATA JMAX /10/                                                     CONC 160
C                                                                        CONC 170
C                                                                        CONC 180
      IF (H .LE. 200.0)  GO TO 115                                       CONC 190
      DO 110 J = 1, JMAX                                                 CONC 200
      CONC(J) = 0.0                                                      CONC 210
  110 CONTINUE                                                           CONC 220
      GO TO 200                                                          CONC 230
  115 CONTINUE                                                           CONC 240
      K = MAX0(2,IFIX(H + 1.5))                                          CONC 250
      K = MIN0(200,K)                                                    CONC 260
      HX = H - FLOAT(K - 1)                                              CONC 270
      DO 130 I = 1, 4                                                    CONC 280
      CX = CONSP(K,I)                                                    CONC 290
      BX = 0.5*(CONSP(K+1,I) - CONSP(K-1,I))                             CONC 300
      AX = BX + CONSP(K-1,I) - CX                                        CONC 310
      CONC(I) = (AX*HX + BX)*HX + CX                                     CONC 320
  130 CONTINUE                                                           CONC 330
  200 CONTINUE                                                           CONC 340
      RETURN                                                             CONC 350
      END                                                                CONC 360
                                                                         CONC 370
```

```
      SUBROUTINE ROMTST (M,N,DINT,T,RINT,KTEST,EPS)                    RTST  10
C                                                                      RTST  20
CC                                                                     RTST  30
CC      M IS INDEX OF M-TH ENTRY TO ROMTST DURING SINGLE INTERVAL      RTST  40
CC      N IS NUMBER OF DIFFERENTIAL EQUATIONS                          RTST  50
CC      DINT IS VECTOR OF ENDPOINT INTEGRATION VALUES                  RTST  60
CC      T IS ARRAY IN WHICH ROMBERG EXTRAPOLATION IS COMPUTED          RTST  70
CC      RINT IS VECTOR IN WHICH FINAL INTEGRALS ARE RETURNED           RTST  80
CC      KTEST IS SET TO 1 IF CONVERGENCE CRITERION IS NOT SATISFIED    RTST  90
CC      EPS IS RELATIVE ERROR ALLOWED                                  RTST 100
CC                                                                     RTST 110
C                                                                      RTST 120
      DIMENSION DINT(50)                                               RTST 130
      DIMENSION RINT(50)                                               RTST 140
      DIMENSION T(50,10)                                               RTST 150
C                                                                      RTST 160
C     IF (M .GT. 1)  WRITE (6,6) (DINT(I), I = 1, N)                   RTST 170
      KTEST = 0                                                        RTST 180
      TIK = 1.0E14                                                     RTST 190
      DO 150 I = 1, N                                                  RTST 200
      T(I,1) = 0.0                                                     RTST 210
      K = 1                                                            RTST 220
  110 CONTINUE                                                         RTST 230
      K = K + 1                                                        RTST 240
      IF (K .GT. M)  GO TO 140                                         RTST 250
      IF (K .GT. 1)  GO TO 130                                         RTST 260
      TA = T(I,1)                                                      RTST 270
      CA = DINT(I)                                                     RTST 280
      T(I,1) = CA                                                      RTST 290
      WA = CA - TA                                                     RTST 300
      SUM = CA                                                         RTST 310
      FAC = 1.0                                                        RTST 320
      GO TO 110                                                        RTST 330
  130 CONTINUE                                                         RTST 340
      TB = TA                                                          RTST 350
      TA = T(I,K)                                                      RTST 360
```

```
      FAC = 4.0*FAC                                                    RTST 370
      FACB = FAC*TB                                                    RTST 380
      DEN = WA/(FACB - CA)                                             RTST 390
      JOV = LEGVAR(DEN)                                                RTST 400
      IF (JOV .NE. 0)  DEN = 0.0                                       RTST 410
      TIK = CA*DEN                                                     RTST 420
      T(I,K) = TIK                                                     RTST 430
      CA = FACB*DEN                                                    RTST 440
      WA = CA - TA                                                     RTST 450
      SUM = SUM + TIK                                                  RTST 460
      IF (TIK .EQ. 0.0)  GO TO 145                                     RTST 470
      GO TO 110                                                        RTST 480
  140 CONTINUE                                                         RTST 490
      IF (ABS(TIK) .GT. EPS*(1.0E5 + SUM))  KTEST = 1                  RTST 500
  145 CONTINUE                                                         RTST 510
      RINT(I) = SUM                                                    RTST 520
  150 CONTINUE                                                         RTST 530
C     WRITE (6,7) M,KTEST,(RINT(I), I = 1, N)                          RTST 540
      RETURN                                                           RTST 550
C   6 FORMAT (7X,1P6E18.9)                                             RTST 560
C   7 FORMAT (1X,2I3,1P6E18.9)                                         RTST 570
      END                                                              RTST 580
```

```
      SUBROUTINE SOLLY (OTHETA,OALT,TIME,SOLDEC,CHI,ETA)                SOLL  10
      DATA PI /3.14159265358979/                                        SOLL  20
      DATA RE /6371.0/                                                  SOLL  30
      PID2 = PI*0.5                                                     SOLL  40
      T = TIME                                                          SOLL  50
      IF (T .GT. 4.32E4)  T = T - 8.64E4                                SOLL  60
      XCRIT = ACOS(RE/(RE + OALT)) + PID2                               SOLL  70
      XMIN = XCRIT-PI/720.0                                             SOLL  80
      XMAX = XCRIT+PI/720.0                                             SOLL  90
      OLAT = PID2-OTHETA                                                SOLL 100
      SINLAT = SIN(OLAT)                                                SOLL 110
      COSLAT = COS(OLAT)                                                SOLL 120
      SINDEC = SIN(SOLDEC)                                              SOLL 130
      COSDEC = COS(SOLDEC)                                              SOLL 140
      CHI = ACOS(SINLAT*SINDEC+COSLAT*COSDEC*COS(2.*PI*T/86400.))       SOLL 150
      IF(CHI.LE.XMIN)  GO TO 1                                          SOLL 160
      IF (CHI .GE. XMAX)  GO TO 2                                       SOLL 170
      HDR = (CHI-XMIN)*720.0/PI                                         SOLL 180
      ETA = 1.-(ACOS(1.-HDR)-(1.-HDR)*SQRT(2.*HDR-HDR*HDR))/PI          SOLL 190
      GO TO 3                                                           SOLL 200
    1 ETA = 1.0                                                         SOLL 210
      GO TO 3                                                           SOLL 220
    2 ETA = 0.0                                                         SOLL 230
    3 RETURN                                                            SOLL 240
      END                                                               SOLL 250
```

```
      SUBROUTINE SUNPHI(DENSCO,ETA,RATE)                                SPHI   10
C          THIS SUBROUTINE COMPUTES THE PHOTODISSOCIATION RATES AT A     SPHI   20
C          POINT IN THE ATMOSPHERE DUE TO THE SOLAR FLUX                 SPHI   30
       DIMENSION PHIZ(30)                                               SPHI   40
       DIMENSION DENSCO(10)                                             SPHI   50
       DIMENSION RATE(30)                                               SPHI   60
       DIMENSION PHITOP(30)                                             SPHI   70
       DIMENSION XS(30,10)                                              SPHI   80
       DATA PHITOP /  4.59E09,2.62E09,2.86E09,2.00E09,2.59E09,1.30E10,  SPHI   90
     1  1.35E09,3.10E11,7.97E09,1.44E10,2.28E10,2.07E10,4.05E10,6.96E10,SPHI  100
     2  1.15E11,2.06E11,5.52E11,1.28E12,2.57E12,5.90E12,1.56E13,6.39E13,SPHI  110
     3  1.05E14,1.26E14,2.97E14,6.68E14,1.93E15,3.67E15,5.95E15,8.68E15/SPHI  120
       DATA (XS(I),I=1,30)       /1.51E-18,1.03E-18,7.91E-19,5.51E-19,  SPHI  130
     1  6.03E-19,1.13E-18,2.59E-19,1.03E-20,2.91E-19,5.84E-19,3.33E-18,SPHI  140
     2  1.37E-17,1.40E-17,1.10E-17,6.54E-18,2.96E-18,7.10E-19,2.25E-20,SPHI  150
     3  1.25E-22,2.50E-23,1.00E-23,7.30E-24,4.12E-24,1.00E-24,6*0.0/   SPHI  160
       DATA (XS(I),I=31,60)      /6*0.0,1.59E-17,2.29E-17,6.20E-18,     SPHI  170
     1  1.23E-17,1.47E-17,6.78E-18,5.16E-18,3.44E-18,1.86E-18,9.36E-19,SPHI  180
     2  8.25E-19,7.95E-19,6.14E-19,3.87E-19,4.07E-19,1.13E-18,3.88E-18,SPHI  190
     3  8.82E-18,1.03E-17,5.22E-18,7.35E-19,3.65E-20,1.03E-21,0.0/     SPHI  200
       DATA (XS(I),I=61,90)      /7*0.0,1.44E-17,6*0.0,2.65E-18,        SPHI  210
     1  4.23E-18,3.80E-18,1.11E-18,6.60E-20,11*0.0/                    SPHI  220
       DATA (XS(I),I=91,120)       /7*0.0,7.32E-20,6*0.0,               SPHI  230
     1  2.6E-19,8.6E-20,2.0E-20,6.7E-21,12*0.0/                        SPHI  240
       DATA XS8CH4 /1.50E-17/                                           SPHI  250
C       XS(I,1) ARE MOLECULAR OXYGEN TOTAL CROSS SECTIONS.             SPHI  260
CC      VALUES FOR I = 7, 30 ARE TAKEN FROM ACKERMAN (1970).           SPHI  270
CC      VALUES FOR I = 1,6 WILL BE REVISED LATER                       SPHI  280
CC      XS(I,2) ARE OZONE CROSS SECTIONS FROM ACKERMAN (1970).         SPHI  290
CC      XS(I,3) ARE WATER VAPOR CROSS SECTIONS.                        SPHI  300
CC      THE ONLY CROSS SECTION  USED FOR CH4 IS XS8CH4 AT 1215.7 A.    SPHI  310
CC      RATE( 1) IS TOTAL  O2 PHOTODISSOCIATION RATE                   SPHI  320
CC      RATE( 2) IS TOTAL CO2 PHOTODISSOCIATION RATE                   SPHI  330
CC      RATE( 3) IS TOTAL  NO PHOTODISSOCIATION RATE                   SPHI  340
CC      RATE( 4) IS TOTAL NO2 PHOTODISSOCIATION RATE                   SPHI  350
C       RATE( 5) IS TOTAL  O3 PHOTODISSOCIATION RATE                   SPHI  360
```

```
C       RATE( 6) IS TOTAL H2O PHOTODISSOCIATION RATE                    SPHI 370
C       RATE( 7) IS TOTAL CH4 PHOTODISSOCIATION RATE                    SPHI 380
C       RATE( 8) IS RATE OF O(3P) PRODUCTION FROM O2 PHOTODISSOCIATION  SPHI 390
C       RATE( 9) IS RATE OF O(1D) PRODUCTION FROM O2 PHOTODISSOCIATION  SPHI 400
C       RATE(10) IS RATE OF O(3P) PRODUCTION FROM O3 PHOTODISSOCIATION  SPHI 410
C       RATE(11) IS RATE OF O(1D) PRODUCTION FROM O3 PHOTODISSOCIATION  SPHI 420
C       FLUX VALUES FOR INTERVALS 1-6 WILL BE REVISED LATER             SPHI 430
C     INPUT- DENSCO(I)=COLUMN NUMBER DENSITY FROM SOURCE TO OBSERVER    SPHI 440
C     IN PARTICLES PER SQ CM.                                           SPHI 450
C     I=1 IS O2, I=2 IS N2 AND I=3 IS O3.                               SPHI 460
C     ETA IS THE FRACTION OF THE SOLAR DISK SHOWING.                    SPHI 470
C     IT COMES FROM SUBROUTINE SOLLY AND RANGES FROM 0 TO 1.            SPHI 480
C                                                                       SPHI 490
C     OUTPUT- RATE(I) = REACTION RATES IN REACTIONS PER SEC.  I=1 IS O2,SPHI 500
C     I=2 IS CO2, I=3 IS NO AND I=4 IS NO2 AT THE OBSERVER.             SPHI 510
C     RATE(5) IS O3 PHOTODISSOCIATION RATE                              SPHI 520
C                                                                       SPHI 530
C     SOLAR FLUX AT Z= INFINITY IN PHOTONS PER SQ CM PER SEC.  PHITOP(I)SPHI 540
C     FOR I=1,30 IS THIS FLUX BROKEN INTO 30 EQUAL WINDOWS OF 0.25      SPHI 550
C     RECIPROCAL MICRONS CENTERED AT 9.875 (I=1) TO 2.625 (I=30)        SPHI 560
C     RECIPROCAL MICRONS.                                               SPHI 570
C                                                                       SPHI 580
C     SET RATES=0 IF ETA=0.                                             SPHI 590
      DO 110 I = 1, 11                                                  SPHI 600
       RATE(I) = 0.0                                                    SPHI 610
  110  CONTINUE                                                         SPHI 620
       IF (ETA .LE. 0.0)  GO TO 500                                     SPHI 630
C                                                                       SPHI 640
C     COMPUTATION OF SOLAR FLUX REACHING HEIGHT Z IN PHOTONS PER SQ CM  SPHI 650
C     PER SEC. THIS IS THE PORTION OF THE SOLAR FLUX THAT HAS NOT BEEN  SPHI 660
C     ABSORBED ENROUTE. IT IS MODULATED BY ETA.                         SPHI 670
C     PHIZ(I) FOR I=1,30 ARE THE THIRTY WINDOWS OF THE SOLAR FLUX       SPHI 680
C     REACHING THE OBSERVER                                             SPHI 690
       COLO2 = ABS(DENSCO(1))                                           SPHI 700
       COLO3 = ABS(DENSCO(3))                                           SPHI 710
      DO 120 I = 7, 29                                                  SPHI 720
```

```
      ATTEN = COLO2*XS(I,1) + COLO3*XS(I,2)                                SPHI 730
      PHIZ(I) = ETA*PHITOP(I)*EXP(-ATTEN)                                  SPHI 740
  120 CONTINUE                                                             SPHI 750
      DJ1 = 0.0                                                            SPHI 760
     DO 130 I = 7, 17                                                      SPHI 770
      DJ1 = DJ1 + PHIZ(I)*XS(I,1)                                          SPHI 780
  130 CONTINUE                                                             SPHI 790
      DJ2 = 1.46E-7*ETA/((COLO2*5.71821E-19)**0.793455 + 1.0)              SPHI 800
     DO 140 I = 21, 24                                                     SPHI 810
      DJ2 = DJ2 + PHIZ(I)*XS(I,1)                                          SPHI 820
  140 CONTINUE                                                             SPHI 830
      DJ3 = PHIZ(8)*XS(8,2)                                                SPHI 840
     DO 150 I = 18, 27                                                     SPHI 850
      DJ3 = DJ3 + PHIZ(I)*XS(I,2)                                          SPHI 860
  150 CONTINUE                                                             SPHI 870
      DJ4 = PHIZ(28)*XS(28,2) + PHIZ(29)*XS(29,2)                          SPHI 880
      DJ5 = PHIZ(8)*XS(8,3)                                                SPHI 890
     DO 160 I = 15, 19                                                     SPHI 900
      DJ5 = DJ5 + PHIZ(I)*XS(I,3)                                          SPHI 910
  160 CONTINUE                                                             SPHI 920
      DJ6 = PHIZ(8)*XS(8,4)                                                SPHI 930
     DO 170 I = 15, 18                                                     SPHI 940
      DJ6 = DJ6 + PHIZ(I)*XS(I,4)                                          SPHI 950
  170 CONTINUE                                                             SPHI 960
      DJ7 = (PHIZ(7) + PHIZ(8))*1.0E-19                                    SPHI 970
     DO 180 I = 9, 19                                                      SPHI 980
      DJ7 = DJ7 + 1.0E-18*PHIZ(I)                                          SPHI 990
  180 CONTINUE                                                             SPHI1000
       RATE(1) = DJ1 + DJ2                                                 SPHI1010
      RATE(2) = DJ6                                                        SPHI1020
      RATE(3) = DJ7                                                        SPHI1030
      RATE(4) = DJ7                                                        SPHI1040
      RATE(5) = DJ3 + DJ4                                                  SPHI1050
      RATE(6) = DJ5                                                        SPHI1060
      RATE(7) = XS8CH4*PHIZ(8)                                             SPHI1070
      RATE(8) = RATE(1) + DJ2                                              SPHI1080
```

```
      RATE(9)  = DJ1                                                SPHI1090
      RATE(10) = DJ4                                                SPHI1100
      RATE(11) = DJ3                                                SPHI1110
500   CONTINUE                                                      SPHI1120
      RETURN                                                        SPHI1130
      END                                                           SPHI1140
```

```
      FUNCTION TKEL (Z)
C     RETURNS TEMPERATURES FOR WINTER 30 DEGREE LATITUDE MODEL
C     ATMOSPHERE IF Z IS LESS THEN 120 KM, AND APPROXIMATE
C     TEMPERATURES FOR HIGH ALTITUDE WINTER MODEL ATMOSPHERE WITH
C     EXOSPHERE TEMPERATURE 1000 K IF Z IS GREATER THAN 120 KM.
C     Z IS INPUT ALTITUDE, KM.
C     TKEL RETURN VALUE IS ABSOLUTE TEMPERATURE, DEGREES KELVIN.
C     LOWER ATMOSPHERE VALUES COMPUTED BY LINEAR INTERPOLATION.
C     HIGH ALTITUDE VALUES COMPUTED BY ANALYTICAL APPROXIMATION.
      DIMENSION ZX(15),TX(15)
      DATA ZX /  0.000,  2.003, 12.039, 17.069, 18.076, 22.107,
     1   32.206, 47.416, 51.484, 59.636, 80.107, 89.361,101.742,
     2  112.096,120.000/
      DATA TX /288.52, 281.86, 216.15, 203.15, 203.15, 213.15,
     1  233.15, 269.15, 269.15, 253.15, 191.15, 191.03, 227.07,
     2  276.22, 334.10/
      IF (Z .GT. 120.0) GO TO 50
      DO 10 I = 1, 15
      IF (Z - ZX(I)) 30, 20, 10
   10 CONTINUE
   20 CONTINUE
      TKEL = TX(I)
      RETURN
   30 CONTINUE
      IF (I .GT. 1)  GO TO 40
      TKEL = 288.52
      RETURN
   40 CONTINUE
      TKEL = TX(I-1) + (TX(I) - TX(I-1))*(Z - ZX(I-1))/
     1  (ZX(I) - ZX(I-1))
      RETURN
   50 CONTINUE
     TKEL = 1000.0 - 665.9*EXP(0.022525*(120.0 - Z))
     RETURN
      END
```

```
TKEL  10
TKEL  20
TKEL  30
TKEL  40
TKEL  50
TKEL  60
TKEL  70
TKEL  80
TKEL  90
TKEL 100
TKEL 110
TKEL 120
TKEL 130
TKEL 140
TKEL 150
TKEL 160
TKEL 170
TKEL 180
TKEL 190
TKEL 200
TKEL 210
TKEL 220
TKEL 230
TKEL 240
TKEL 250
TKEL 260
TKEL 270
TKEL 280
TKEL 290
TKEL 300
TKEL 310
TKEL 320
TKEL 330
TKEL 340
TKEL 350
```


☆U. S. GOVERNMENT PRINTING OFFICE: 1975--600-019--49