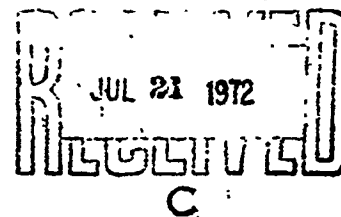


AD 745319

A HIGH ALTITUDE RADIANCE MODEL

Thomas C. Degges

Visidyne, Inc.
169 Merrimac Street
Woburn, Massachusetts 01801



Contract No. F19628-71-C-0156
Project No. 8692

FINAL REPORT

17 March 1971 - 14 May 1972

14 May 1972

Contract Monitor: Bertram D. Schurin
Optical Physics Laboratory

Approved for public release; distribution unlimited

Sponsored by

Advanced Research Projects Agency
ARPA Order No. 1366

Monitored by

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
BEDFORD, MASSACHUSETTS 01730

NATIONAL TECHNICAL
INFORMATION SERVICE

Unclassified
Security Classification

DOCUMENT CONTROL DATA - F1D		
(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)		
1. ORIGINATING ACTIVITY (Corporate author) Visidyne, Inc. 169 Merrimac Street Woburn, Massachusetts 01801		2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP n/a
3. REPORT TITLE A HIGH ALTITUDE RADIANCE MODEL		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Scientific. Final. 17 March 1971-14 May 1972		Approved 6 June 1972
5. AUTHOR(S) (First name, middle initial, last name) Thomas C. Degges		
6. REPORT DATE 14 May 1972	7a. TOTAL NO. OF PAGES 226	7b. NO. OF REFS 153
8a. CONTRACT OR GRANT NO. F19628-71-C-0156	9a. ORIGINATOR'S REPORT NUMBER(S) VI-91	
b. PROJECT, TASK, WORK UNIT NOS. 8692 n/a n/a	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) AFCRL-72-0273	
c. DOD ELEMENT 62301D		
d. DOD SUBELEMENT n/a		
10. DISTRIBUTION STATEMENT A - Approved for public release; distribution unlimited.		
11. SUPPLEMENTARY NOTES This research was supported by Advanced Research Projects Agency.		12. SPONSORING MILITARY ACTIVITY Air Force Cambridge Research Laboratories (OP) L. G. Hanscom Field Bedford, Massachusetts 01730
13. ABSTRACT 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 computer program that computes infrared radiances for an earth's limb viewing geometry. The nominal spectral region of this study lies between five and twenty-five micrometers and emphasis is placed on radiation originating at altitudes between 70 and 500 km. An earlier model for the transport of infrared radiation in molecular bands with Doppler line shape, accurate only for linear molecules, has been extended to bands of polyatomic molecules. A chemistry program which includes the effects of vertical transport by eddy mixing and molecular diffusion has been developed to make possible an estimate of diurnal variation in the abundances of infrared emitting species. Computational results are given for the latitude of		

DD FORM 1473 1 NOV 66 White Sands, New Mexico.

Unclassified
Security Classification

Unclassified
Security Classification

18. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Atmospheric Radiance Model Infrared						

Unclassified
Security Classification

iii

A HIGH ALTITUDE RADIANCE MODEL

Thomas C. Degges

Visidyne, Inc.
159 Merrimac Street
Woburn, Massachusetts 01801

Contract No. F19628-71-C-0156
Project No. 8692

FINAL REPORT

17 March 1971 - 14 May 1972

14 May 1972

Contract Monitor: Bertram D. Schurin
Optical Physics Laboratory

Approved for public release; distribution unlimited

Sponsored by

Advanced Research Projects Agency
ARPA Order No. 1366

Monitored by

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
BEDFORD, MASSACHUSETTS 01730

TABLE OF CONTENTS

<u>Section</u>		<u>Page No.</u>
I	Introduction.....	1
II	Chemistry	5
III	Excitation Mechanisms.....	33
IV	Description of Programs.....	63
V	Results and Discussion.....	95
	References	105
	Appendix	115

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 five and twenty-five 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) who 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. For these reasons, one of the tasks of the current contract has been to develop 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 quickly incorporate 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 vibrational temperatures. Corbin, et al. (1969) included collisional excitation by molecular nitrogen and oxygen in their model and the work reported here adds the effects of atomic oxygen as a collision partner.

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. Corbin et al. (1969) 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. This approach has been extended to water vapor and ozone bands lying in the spectral region of interest.

The following sections of this report summarize the research performed during this contract. The computational algorithm used in the chemistry program, relevant chemical reaction rate coefficients and currently known chemical abundances in the upper atmosphere are discussed in Section II. Relevant collisional and radiative excitation mechanisms are reviewed in Section III, where the computational model for predicting high altitude infrared radiance is developed. The chemistry and infrared radiance programs are described and listed in Section IV. Sample results for White Sands, New Mexico are presented in Section V, where limitations of the model and suggestions for future work are discussed.

SECTION II

CHEMISTRY

There have been relatively few measurements of the variation with altitude of infrared emitting species at altitudes above 30 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 are reasonably well known between about 80 and 150 km, 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 (1971a) 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 (1971b) gives a measured altitude profile for hydroxyl with error limits ± 120 per cent.

At present, then, estimates of chemical abundances of minor species required to predict infrared radiation require the use of photochemical computations. Most computations of atmospheric chemistry before the past four or five years have neglected vertical transport by molecular diffusion and eddy mixing. For molecules which have photochemical lifetimes greater than 10^4 to 10^5 seconds, this neglect may yield results greatly in error, as a comparison of the calculations of Hunt (1966), Hesstvedt (1968) and Shimazaki

and Laird (1970) quickly shows. Even calculations including transport processes cannot be accepted at face value because the values of eddy mixing coefficients used at present are for the most part only informed estimates, and published calculations often use outdated rate coefficients. Because of the above reasons and the future need to estimate abundances for conditions for which no published experimental data or calculations may be available, development of a computational model for atmospheric photochemistry including vertical transport was considered desirable. The present status of this model is described below.

Following Shimazaki and Laird (1970) and Bowman, et al. (1970), the continuity equation for an atmospheric chemical constituent or species may be written

$$\frac{\partial n(z)}{\partial t} = Q(z) - L(z) - \frac{\partial \Phi(z)}{\partial z} \quad (2-1)$$

where n is the concentration of the constituent, Q and L are photochemical production and loss rates and Φ is the vertical flux due to transport processes. When only vertical transport by molecular diffusion and eddy or turbulent mixing is considered, the vertical flux is

$$\Phi(z) = n(z)u(z) + n(z)v(z) \quad (2-2)$$

where u and v are vertical velocities of the species produced by molecular diffusion and eddy mixing, respectively. If D is an "average" molecular diffusion coefficient, H is the scale height of the species in the absence of

mixing and T is the temperature,

$$u(z) = -D \left(\frac{\partial n}{\partial z} + \frac{n}{H} + \frac{n}{T} \frac{\partial T}{\partial z} \right) \quad (2-3)$$

If K is the eddy mixing coefficient and H_{av} is the average scale height for all species present,

$$v(z) = -K \left(\frac{\partial n}{\partial z} + \frac{n}{H_{av}} + \frac{n}{T} \frac{\partial T}{\partial z} \right) \quad (2-4)$$

Then

$$\frac{\partial n(z)}{\partial t} = Q(z) - L(z) + \frac{\partial}{\partial z} \left[(D+K) \left(\frac{\partial n}{\partial z} + \frac{n}{T} \frac{\partial T}{\partial z} \right) + \left(\frac{D}{H} + \frac{K}{H_{av}} \right) n \right] \quad (2-5)$$

As was done by Bowman, et al. (1970), it is assumed that only atomic oxygen and molecular nitrogen and oxygen are important as collision partners. The effective diffusion coefficients for atomic oxygen and molecular nitrogen and oxygen are taken to be

$$D_i(z) = N_o \left(\frac{T(z)}{T_o} \right)^{0.75} / \sum_{j=1}^3 \frac{n_j(z)}{D_{ij}} \quad (2-6)$$

where N_o is number density at standard temperature and pressure (STP), T_o is 273.15 K, and the D_{ij} are mutual diffusion coefficients at STP. The

values adopted for $D_{O_2-N_2}$, D_{O-O_2} and D_{O-N_2} are those used by Colegrove, et al. (1966), 0.181, 0.260 and 0.260, respectively. That for $D_{O_2-O_2}$, 0.189, is taken from Chapman and Cowling (1960), while $D_{N_2-N_2}$ and D_{O-O} are estimated to be 0.180 and 0.370. The effective diffusion coefficients for other species are taken to be

$$D_i = 0.707 D_{O_2} \left(1 + \frac{m(O_2)}{m_i} \right)^{1/2} \quad (2-7)$$

where m_i represents molecular weight.

In converting Equation (2-5) into a difference scheme for numerical integration, it is assumed that all quantities may be represented by a quadratic function of altitude at any given altitude. Then if the superscript (+) denotes a value at the next higher integration point and the superscript (-) denotes a value at the next lower point, the difference equation is

$$\begin{aligned} \frac{\partial n}{\partial t} = & Q - L + \frac{D^+ - D^-}{2\Delta z} \left[\frac{n^+ - n^-}{2\Delta z} + \left(\frac{1}{T} \frac{T^+ - T^-}{2\Delta z} + \frac{1}{H} \right) n \right] \\ & + \frac{K^+ - K^-}{2\Delta z} \left[\frac{n^+ - n^-}{2\Delta z} + \left(\frac{1}{T} \frac{T^+ - T^-}{2\Delta z} + \frac{1}{H_{av}} \right) n \right] \\ & + (D + K) \left[\frac{n^+ - 2n + n^-}{(\Delta z)^2} - \left(\frac{1}{T} \frac{T^+ - T^-}{2\Delta z} \right)^2 n + \frac{1}{T} \left(\frac{T^+ - 2T + T^-}{(\Delta z)^2} \right) n \right. \\ & \left. + \frac{1}{T} \left(\frac{T^+ - T^-}{2\Delta z} \right) \left(\frac{n^+ - n^-}{2\Delta z} \right) + \frac{1}{H} \left(\frac{n^+ - n^-}{2\Delta z} \right) - \frac{n}{H^2} \left(\frac{H^+ - H^-}{2\Delta z} \right) \right] \\ & + D \left[\frac{1}{H} \left(\frac{n^+ - n^-}{2\Delta z} \right) - \frac{n}{H^2} \left(\frac{H^+ - H^-}{2\Delta z} \right) \right] + K \left[\frac{1}{H_{av}} \left(\frac{n^+ - n^-}{2\Delta z} \right) - \frac{n}{H_{av}^2} \left(\frac{H_{av}^+ - H_{av}^-}{2\Delta z} \right) \right] \end{aligned}$$

(2-8)

The computational problem then is simply to integrate Equation (2-8) at a set of altitudes with spacing Δz , taking successive time steps $h = \Delta t$ over the desired time of interest. The method used takes advantage of the observation that except where a species is important as a third body in an association reaction which removes it, the chemical rate equations for a coupled system may be cast in the form

$$\dot{x}_i = q_i(x) - \beta_i(x)x_i - \alpha_i(x)x_i^2 \quad (2-9)$$

with formal solutions (assuming q , α and β constant and omitting obvious subscripts)

$$x_i(t) = \frac{[(\beta + \delta)\exp(-\delta t) + (\delta - \beta)]x_i(0) + 2q[1 - \exp(-\delta t)]}{(2\alpha x_i(0) + \beta)[1 - \exp(-\delta t)] + [1 + \exp(-\delta t)]\delta}$$

$$q > 0, \beta > 0, \alpha > 0 \quad (2-10a)$$

where

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

$$x_i(t) = x_i(0)\exp(-\beta t) + q[1 - \exp(-\beta t)]/\beta$$

$$q > 0, \beta > 0, \alpha = 0 \quad (2-10b)$$

$$x_i(t) = \frac{[1 + \exp(-\delta t)]x_i(0)\delta + 2q[1 - \exp(-\delta t)]}{2\alpha x_i(0)[1 - \exp(-\delta t)] + [1 + \exp(-\delta t)]\delta}$$

$$q > 0, \beta = 0, \alpha > 0 \quad (2-10c)$$

where

$$\delta = (4\alpha q)^{1/2}$$

$$x_i(t) = x_i(0) + qt \quad q > 0, \beta = 0, \alpha = 0 \quad (2-10d)$$

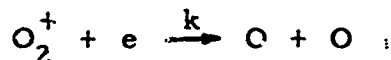
$$x_i(t) = \frac{\beta x_i(0) \exp(-\beta t)}{\alpha x_i(0)[1 - \exp(-\beta t)] + \beta}, \quad q = 0, \beta > 0, \alpha > 0 \quad (2-10e)$$

$$x_i(t) = x_i(0)\exp(-\beta t), \quad q = 0, \beta > 0, \alpha = 0 \quad (2-10f)$$

$$x_i(t) = x_i(0)/[1 + \alpha t x_i(0)] \quad q = 0, \beta = 0, \alpha > 0 \quad (2-10g)$$

$$x_i(t) = x_i(0), \quad q = 0, \beta = 0, \alpha = 0 \quad (2-10h)$$

Care must be exercised in the definition of the α and β terms. For example, the reaction



contributes to both the α and the β terms in the differential equation for O_2^+ concentration if other positive ions are present;

$$\alpha = k$$

$$\beta = k([e] - [\text{O}_2^+])$$

The solutions, Equation (2-10), are exact only if q , β , and α do not change during an integration interval. Since q , β , and α are functions of the time dependent concentrations of the other species in the system, the actual integration scheme must allow for this variation.

If the time step $t \ll 2/|2\alpha x + \beta - \delta|$ and $\delta t \ll 2$, the denominator of Equation (2-10a) may be expanded as $1 - (2\alpha x + \beta - \delta)t/2\delta$ and

$$x_i(t) = x_i(0) + (q_i - \beta_i x_i(0) - \alpha_i x_i^2(0))t + O(t^2) \quad (2-11)$$

For a small enough time step, the algorithm is equivalent to Euler's method. With the same restrictions on the time step, the algorithm may be embedded in

a Runge-Kutta formula of any order and will give equivalent results.

The program uses an analog of the improved tangent second-order Runge-Kutta method in the basic step h . Let $x(0)$ be the value of a dependent variable at the beginning of the step. Then $\alpha(0)$, $\beta(0)$ and $q(0)$ are computed and a new value of x computed for $t = \frac{1}{2}h$ from the general expression, Equation (2-10a), or from an equivalent expression, if any α , β or q is equal to zero. $\alpha(h/2)$, $\beta(h/2)$ and $q(h/2)$ are computed and Equation (2-10) evaluated with these values and $t = h$.

Following Ceschino and Kuntzmann (1966), assume that the stepsize is chosen so that a linearized theory of errors may be applied. That is, for a method of order n , the error for one step of length h is Ah^{n+1} . Divide an interval Ht into an even number k of steps of length h . Then

$$x_i^{(h)}(t) = x_i^{(o)}(t) + kAh^{n+1} + O(h^{n+2}) \quad (2-12)$$

where $x_i^{(o)}(t)$ is the "exact" result of the integration and addition or subtraction of the error by propagation is neglected. The result using a step of length $2h$ is

$$x_i^{(2h)}(t) = x_i^{(o)}(t) + k/2 A(2h)^{n+1} + O(h^{n+2}) \quad (2-13)$$

The difference between the two results is

$$x_i^{(2h)}(t) - x_i^{(h)}(t) \approx kAh^{n+1}(2^n - 1) \quad (2-14)$$

and the principal part of the error using the step h is

$$\epsilon_h \approx \frac{x_i^{(2h)}(t) - x_i^{(h)}(t)}{2^n - 1} \quad (2-15)$$

The simplest way to use this result is to divide an interval into two equal steps h and integrate twice, once with two steps of length h and then with one step of length $2h$. The principal part of the error may be used both to correct the result $x_i^{(2h)}(t)$ and to estimate the optimum length of the next interval of integration. The Runge-Kutta integration program of Gear (1971) is easy to adapt for both uses of the error estimate, and was used in early tests.

It is more efficient to use Romberg extrapolation, which subdivides an interval H in more than two ways, and uses higher order error terms. Either polynomial Romberg extrapolation [Davis and Rabinowitz, (1967)] or rational Romberg extrapolation [Bulirsch and Stoer (1966)] may be used.

Denote the result of integrating from the $t = 0$ to $t = H$ with a constant step size h by $X(h)$. Assume that $X(h)$ has an asymptotic expansion

$$X(h) = \xi_0 + \xi_1 h^{\gamma_1} + \xi_2 h^{\gamma_2} + \dots \quad (2-16)$$

with $\xi_0, \xi_1, \xi_2, \dots$ constant, so that in the limit $h \rightarrow 0$, $X(0) = \xi_0$.

the exact solution. An approximation to $X(0)$ may be obtained by extrapolation from the results of integration with finite values of h .

Let the global interval of integration H be successively divided into 2, 3, 4, 6, 8, 12, ... intervals with $h = \frac{1}{2}H, \frac{1}{3}H, \frac{1}{4}H, \frac{1}{6}H, \dots$

Let $X_0^{(0)}(H)$ be the value obtained with $h = \frac{1}{2}H$, $X_0^{(1)}(H)$ be that obtained from $h = \frac{1}{3}H$, etc. Then a table of extrapolated values

$$\begin{array}{cccccc}
 X_0^{(0)}(H) & & & & & \\
 X_0^{(1)}(H) & X_1^{(0)}(H) & & & & \\
 X_0^{(2)}(H) & X_1^{(1)}(H) & X_2^{(0)}(H) & & & \\
 X_0^{(3)}(H) & X_1^{(2)}(H) & X_2^{(1)}(H) & X_3^{(0)}(H) & & \\
 \dots & \dots & \dots & \dots & \dots & \\
 X_0^{(m)}(H) & X_1^{(m-1)}(H) & X_2^{(m-2)}(H) & X_3^{(m-3)}(H) & \dots & X_m^{(0)}(H)
 \end{array} \tag{2-17}$$

is computed using

$$X_1^{(0)}(H) = \frac{A_1^{(0)} X_0^{(1)}(H) - X_0^{(0)}(H)}{A_1^{(0)} - 1}$$

$$X_1^{(1)}(H) = \frac{A_1^{(1)} X_0^{(2)}(H) - X_0^{(1)}(H)}{A_1^{(1)} - 1} \quad X_2^{(0)}(H) = \frac{A_2^{(0)} X_1^{(1)}(H) - X_1^{(0)}(H)}{A_2^{(0)} - 1}$$

.....

(2-18)

$$X_k^{(j)}(H) = \frac{A_k^{(j)} X_{k-1}^{(j+1)}(H) - X_{k-1}^{(j)}(H)}{A_k^{(j)} - 1}$$

If the method used in integrating over a step h is of order n , the exponents in Equation (2-16) have the values

$$\gamma_i = n - 1 + i \quad (2-19)$$

Equation (2-16) may be written out for as many values of h as desired and these equations solved simultaneously. The coefficients $A_k^{(j)}$ of Equation (2-18) are obtained from the multipliers used in solving such a system of algebraic equations. Table II-1 gives the coefficients $A_k^{(j)}$ for $k = 1, 2, \dots, 9$ for first, second, third and fourth order integration methods. Only the second order coefficients are used here.

Subdivision of the basic interval is continued until the value of

Table II-1

Coefficients $A_k^{(j)}$ for Polynomial Extrapolation

using Subintervals $H/2, H/3, H/4, H/6, H/8, \dots$

<u>k</u>	<u>First Order</u>		<u>Second Order</u>		<u>Third Order</u>		<u>Fourth Order</u>	
	<u>j even</u>	<u>j odd</u>	<u>j even</u>	<u>j odd</u>	<u>j even</u>	<u>j odd</u>	<u>i even</u>	<u>j odd</u>
1	$\frac{3}{2}$	$\frac{4}{3}$	$\frac{9}{4}$	$\frac{16}{9}$	$\frac{27}{8}$	$\frac{64}{27}$	$\frac{81}{16}$	$\frac{256}{81}$
2	2	2	$\frac{20}{7}$	$\frac{14}{5}$	$\frac{152}{37}$	$\frac{74}{19}$	$\frac{208}{35}$	$\frac{70}{13}$
3	3	$\frac{8}{3}$	$\frac{13}{3}$	$\frac{48}{13}$	$\frac{69}{11}$	$\frac{352}{69}$	$\frac{173}{19}$	$\frac{1216}{173}$
4	4	4	$\frac{40}{7}$	$\frac{28}{5}$	$\frac{2320}{283}$	$\frac{1132}{145}$	$\frac{1760}{149}$	$\frac{596}{55}$
5	6	$\frac{16}{3}$	$\frac{198}{23}$	$\frac{736}{99}$	$\frac{582}{47}$	$\frac{3008}{291}$	$\frac{9666}{541}$	$\frac{69248}{4833}$
6	8	8	$\frac{80}{7}$	$\frac{56}{5}$	$\frac{3424}{209}$	$\frac{1672}{107}$	$\frac{23360}{991}$	$\frac{7928}{365}$
7	12	$\frac{32}{3}$	$\frac{292}{17}$	$\frac{1088}{73}$	$\frac{7716}{313}$	$\frac{40064}{1929}$	$\frac{89476}{2521}$	$\frac{645376}{22369}$
8	16	16	$\frac{160}{7}$	$\frac{112}{5}$	$\frac{43072}{1315}$	$\frac{21040}{673}$	$\frac{10880}{231}$	$\frac{3696}{85}$
9	24	$\frac{64}{3}$	$\frac{3672}{107}$	$\frac{13696}{459}$	$\frac{3592}{73}$	$\frac{18688}{449}$	$\frac{11112}{157}$	$\frac{80384}{1389}$

$$\left(X_k^{(0)} - X_{k-1}^{(0)} \right) / X_k^{(0)}$$

is less than a preassigned value ϵ . In practice, adequate results are obtained if for $k > 6$ the test is on values of

$$\left(X_6^{(j)} - X_6^{(j-1)} \right) / X_6^{(j)}.$$

The chief advantage of this integration method is that large time intervals may be used and the algorithm contains an error estimate.

As applied in the program, the L term in Equation (2-8) is divided into two parts, $\propto n^2$ and $\propto n$. To the Q term is added the terms containing n^+ and n^- . The multipliers of the terms in n are added to the β term from the photochemistry and Equation (2-10) is used with interval subdivision and polynomial Romberg extrapolation as described above.

Table II-2 lists the main neutral reactions which may play a role in the chemistry of the infrared emitting molecules NO , CO_2 , H_2O , O_3 , N_2O and CH_4 . Table II-3 lists the photodissociation processes considered. Table II-4 lists the ionic reactions needed for nitric oxide chemistry. Where possible, an estimated error is given for the value of the rate coefficient. Critical reviews are the principal source of rates listed here and in most cases newer rates from the literature fall within estimated error limits.

Not all the reactions listed in Table II-2 have been included in the program, as some occur in branching chains which have not been verified in

Table II-2

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		$1.65 \times 10^{-9} \exp(-1120/T)$	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%

	<u>Reaction</u>	<u>Rate Coefficient, c.g.s. units</u>	<u>Source</u>	<u>Error Estimate</u>
R9	$\text{OH} + \text{O}_3 \rightarrow \text{HO}_2 + \text{O}_2$	45.0×10^{-13}	Kaufman (1969)	
R10	$\text{HO}_2 + \text{O}_3 \rightarrow \text{OH} + 2\text{O}_2$	10^{-14}	Hunt (1964)	An estimate
R11a	$\text{OH} + \text{OH} \rightarrow \text{H}_2\text{O} + \text{O}$	$9.3 \times 10^{-12} \exp(-390/T)$	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	$\text{OH} + \text{HO}_2 \rightarrow \text{H}_2\text{O} + \text{O}_2$	$> 10^{-11}$	Kaufman (1969)	See Text
R13	$\text{H} + \text{HO}_2 \rightarrow \text{H}_2 + \text{O}_2$	2.0×10^{-13}	Glyne and Thrush (1963)	30%
R14	$\text{O} + \text{H}_2 \rightarrow \text{OH} + \text{H}$	$7.0 \times 10^{-11} \exp(-5100/T)$	Wong and Potter (1965)	30%
R15	$\text{HO}_2 + \text{HO}_2 \rightarrow \text{H}_2\text{O}_2 + \text{O}_2$	1.5×10^{-12}	Kaufman (1969)	30%
R16	$\text{OH} + \text{H}_2\text{O}_2 \rightarrow \text{H}_2\text{O} + \text{HO}_2$	4.0×10^{-13}	Foner and Hudson (1962)	50%
R17	$\text{O} + \text{H}_2\text{O}_2 \rightarrow \text{OH} + \text{HO}_2$	10^{-15}	Foner and Hudson (1962)	50%
R18	$\text{H} + \text{H}_2\text{O}_2 \rightarrow \text{H}_2 + \text{HO}_2$	10^{-13}	Foner and Hudson (1962)	50%
R19a	$\text{OH} + \text{H}_2 \rightarrow \text{H}_2\text{O} + \text{H}$	$* 3.6 \times 10^{-11} \exp(-2600/T)$	Baulch, et al. (1968b)	35% at 300K
R19b		$* 1.4 \times 10^{-10} \exp(-10'00/T)$	Baulch, et al. (1968b)	35% at 300 K
R20	$\text{O} + \text{H} + \text{M} \rightarrow \text{OH} + \text{M}$	$* 8.0 \times 10^{-33}$	Bates and Nicolet (1950)	Estimate
R21	$\text{O} + \text{OH} + \text{M} \rightarrow \text{HO}_2 + \text{M}$	$* 1.4 \times 10^{-31}$	Kaufman (1964)	30%

	<u>Reaction</u>	<u>Rate Coefficient, c.g.s. units</u>	<u>Source</u>	<u>Error Estimate</u>
R22	$H + H + M \rightarrow H_2 + M$	$* 1.2 \times 10^{-52} (273/T)^{0.7}$	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)	50%
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	$O(^1D) + CH_4 \rightarrow \text{Products}$	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%

	<u>Reaction</u>	<u>Rate Coefficient, c.g.s. units</u>	<u>Source</u>	<u>Error Estimate</u>
R38	$\text{CH}_3 + \text{O}_2 \rightarrow \text{CH}_2\text{O} + \text{OH}$	$*5.0 \times 10^{-11} \exp(-5000/T)$	Dean and Kistiakowski (1971)	Uncertain
R39	$\text{CH}_2\text{O} + \text{OH} \rightarrow \text{H}_2\text{O} + \text{HCO}$	$*5.3 \times 10^{-10} \exp(-2120/T)$	Schofield (1967)	50%
R40	$\text{CH}_2\text{O} + \text{HO}_2 \rightarrow \text{H}_2\text{O}_2 + \text{HCO}$	$*7.5 \times 10^{-15}$	Schofield (1967)	Uncertain
R41	$\text{HCO} + \text{O}_2 \rightarrow \text{CO} + \text{HO}_2$	$*1.0 \times 10^{-13}$	Estimate	Uncertain
R42a	$\text{N} + \text{O}_2 \rightarrow \text{NO} + \text{O}$	$1.1 \times 10^{-14} T \exp(-3140/T)$	Baulch, et al. (1969)	30% at 300 K
R42b		$2.6 \times 10^{-15} T \exp(-19500/T)$	Baulch, et al. (1969)	30%
R43a	$\text{N} + \text{NO} \rightarrow \text{N}_2 + \text{O}$	$5.1 \times 10^{-11} \exp(-168/T)$	Baulch, et al. (1969)	Factor of 2
R43b		$2.3 \times 10^{-10} \exp(-37500/T)$	Baulch, et al. (1969)	Factor of 2
R44a	$\text{NO} + \text{NO} \rightarrow \text{N}_2 + \text{O}_2$	$1.3 T^{-5/2} \exp(-42800/T)$	Freedman and Daiber (1961)	Highly uncertain
R44b		$15.2 T^{-5/2} \exp(-64300/T)$	Wray and Teare (1962)	Highly uncertain
R45	$\text{N} + \text{N} + \text{M} \rightarrow \text{N}_2 + \text{M}$	$1.4 \times 10^{-33} \exp(-500/T)$	Schiff (1969)	Factor of 2
R46a	$\text{N} + \text{O} + \text{M} \rightarrow \text{NO} + \text{M}$	$1.1 \times 10^{-32} (273/T)^{1/2}$	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	$\text{O} + \text{NO} \rightarrow \text{NO}_2 + h\nu$	$6.5 \times 10^{-17} (T/296)$	Baulch, et al. (1970)	Factor of 2
R48	$\text{O} + \text{NO}_2 \rightarrow \text{NO} + \text{O}_2$	$1.6 \times 10^{-11} \exp(-300/T)$	Baulch, et al. (1970)	20%

	<u>Reaction</u>	<u>Rate Coefficient, c.g.s. units</u>	<u>Source</u>	<u>Error Estimate</u>
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,etal.(1970)	20%
R53b		$1.8 \times 10^{-8} \exp(-32800/T)$	Baulch,etal.(1970)	25% at 1500 K
R54a	$NO + NO + O_2 \rightarrow NO_2 + NO_2$	$6.6 \times 10^{-39} \exp(530/T)$	Baulch,etal.(1970)	50%
R54b		$6.6 \times 10^{-12} \exp(-13500/T)$	Baulch,etal.(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,etal.(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 II-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	$H_2O_2 + h\nu \rightarrow OH + OH$	Schurnb, et al. (1955)
J7	$HC_2 + h\nu \rightarrow OH + O(^3P)$	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).

Table II-4

Ionic Reactions Required for Nitric Oxide Chemistry

	Reaction	Rate Coefficient, cgs units	Source
I1a	$N^+ + O_2 \rightarrow NO^+ + O$	6.0×10^{-10}	Fehsenfeld, et al. (1969)
I1b	$N^+ + O_2 \rightarrow O_2^+ + N$		
I2	$O^+ + N_2 \rightarrow NO^+ + N$	$6.0 \times 10^{-13} (600/T)$	Dunkin, et al. (1968)
I3	$O^+ + O_2 \rightarrow O_2^+ + O$	$1.1 \times 10^{-9} T^{-0.7}$	Dunkin, et al. (1968)
I4	$O^+ + NO \rightarrow NO^+ + O$	2.0×10^{-11}	Goldan, et al. (1966)
I5	$O^+(^2D) + N_2 \rightarrow N_2^+ + O$	1.0×10^{-9}	Dalgarno and McElroy (1966)
I6	$O^+(^2D) + e \rightarrow O^+(^4S) + e$	1.0×10^{-7}	Huntten and McElroy (1966)
I7a	$N_2^+ + O \rightarrow NO^+ + N(^4S)$	1.4×10^{-10}	Fehsenfeld, et al. (1970)
I7b	$N_2^+ + O \rightarrow NO^+ + N(^2D)$		
I8a	$N_2^+ + e \rightarrow 2N(^4S)$	2.8×10^{-7}	Kasner and Biondi (1965)
I8b	$N_2^+ + e \rightarrow N(^2D) + N(^4S)$		
I9	$N_2^+ + O_2 \rightarrow O_2^+ + N_2$	$6.0 \times 10^{-11} (300/T)^{0.6}$	Johnson, et al. (1970)
I10	$N_2^+ + NO \rightarrow NO^+ + N_2$	3.3×10^{-10}	Fehsenfeld, et al. (1970)
I11	$O_2^+ + N \rightarrow NO^+ + O$	1.8×10^{-10}	Goldan, et al. (1966)
I12	$O_2^+ + NO \rightarrow NO^+ + O_2$	6.3×10^{-10}	Fehsenfeld, et al. (1970)
I13	$O_2^+ + e \rightarrow 2O$	$6.6 \times 10^{-5}/T$	Biondi (1969)
I14a	$NO^+ + e \rightarrow N(^4S) + O$	$1.0 \times 10^{-7} (T/1000)^{-1.5}$	Weller and Biondi (1968)
I14b	$NO^+ + e \rightarrow N(^2D) + O$		
I15	$O_2^+ + N_2 \rightarrow NO^+ + NO$	$< 2 \times 10^{-16}$	Shahin (1967)

the laboratory. The choice of reactions has been guided by those used in published calculations and reviews. The most uncertain rates are those involving the carbon compounds. These have been studied principally at elevated temperatures and extrapolation to the low temperatures of the upper stratosphere and mesosphere is not reliable.

The results of Nicolet (1970) and Bowman, et al. (1971) were used as a guide to the selection of reactions to include in the chemistry of oxygen and hydrogen compounds. Rates for Reactions R1, R2 and R3 were taken from Johnston (1968) in preference to other sources because they are thermodynamically consistent for the $O-O_2-O_3$ system. The rates for nitric oxide reactions follow closely those used by Stroebel (1971a, 1971b). The choice of reactions for methane decomposition is based on the partial kinetic systems of Herron (1969) and Hoare and Patel (1969). Reactions leading to formation of CH_2 have been omitted.

Of the chemical species considered here, nitric oxide abundances are best known above an altitude of 70 km. Observations of solar ultraviolet induced fluorescence in the gamma bands have been made by Barth (1964, 1966a, 1966b), Pearce (1969a) and Meira (1971). Abundances measured in this way are much higher than predicted by earlier photochemical theories such as those of Nicolet (1965a; 1965b). Norton and Barth (1970) and Stroebel, et al. (1970) showed that $N(^2D)$ produced in dissociative recombination of NO^+ (Reaction I14b) and in charge exchange of N_2^+ with atomic oxygen (Reaction I7b) could form sufficient nitric oxide in reacting with molecular oxygen to explain measured abundances. The amount of $N(^2D)$ formed in the relevant reactions has not been determined in the laboratory and remains an adjustable parameter in all computations. The nitric oxide concentration above 80 km may be considered reasonably well known, probably to within a factor of two. Below that altitude, the chemistry is uncertain and the effects of reactions such as

R54, R55 and R60 have not been studied thoroughly. Stroebel (1972) shows that values lower than those obtained by Meira (1971) are possible and suggests using limits obtained from electron densities in the D region. Figure II-1 shows nitric oxide concentrations used here as a basis for predicting infrared radiance at high altitudes, and is based on Stroebel's calculations.

Water vapor abundances are much less well known. There is general agreement that water vapor is present in the upper stratosphere in amounts of between 2 and 10 parts per million, Anderson (1971a), but the exact amount is unknown. Anderson (1971a, 1971b) has determined OH abundances from observation of fluorescent scattering of solar ultraviolet radiation. The concentrations are about 5×10^6 molecules/cm³ between 45 and 70 km, with error limits of about 120 per cent. Anderson's first paper includes a steady state photochemical model with diffusion and he uses this to infer a water vapor abundance of about one to two parts per million at 85 km. The calculations with the chemistry model described above assumed an initial concentration of three parts per million and were not run long enough to reach diurnal equilibrium. In addition, the eddy diffusion coefficients above 50 km, taken from Keneshea and Zimmerman (1970), are larger than the upper limit deduced by Anderson (1971a). It was decided to use them, however, because M. B. McElroy (private communication, 1972) has informed us that there has been an unpublished measurement of Reaction R12 which gives a value 2×10^{-10} cm³/sec which may lead to lower OH concentrations for a given water vapor abundance. The adopted water vapor abundances are given in Figure II-2, and are based on the (incomplete) chemistry program results.

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

Nitric Oxide Concentration Profiles

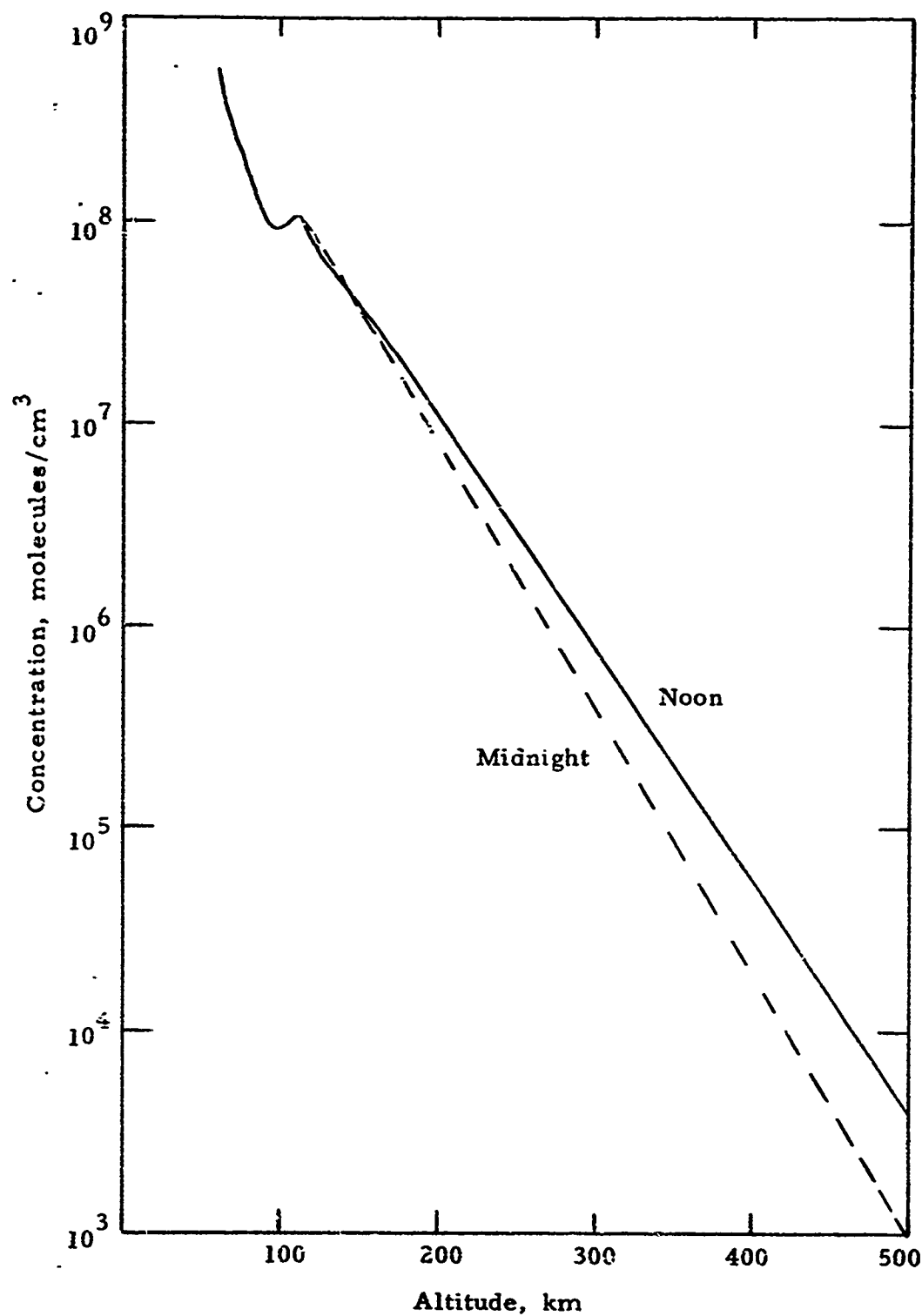
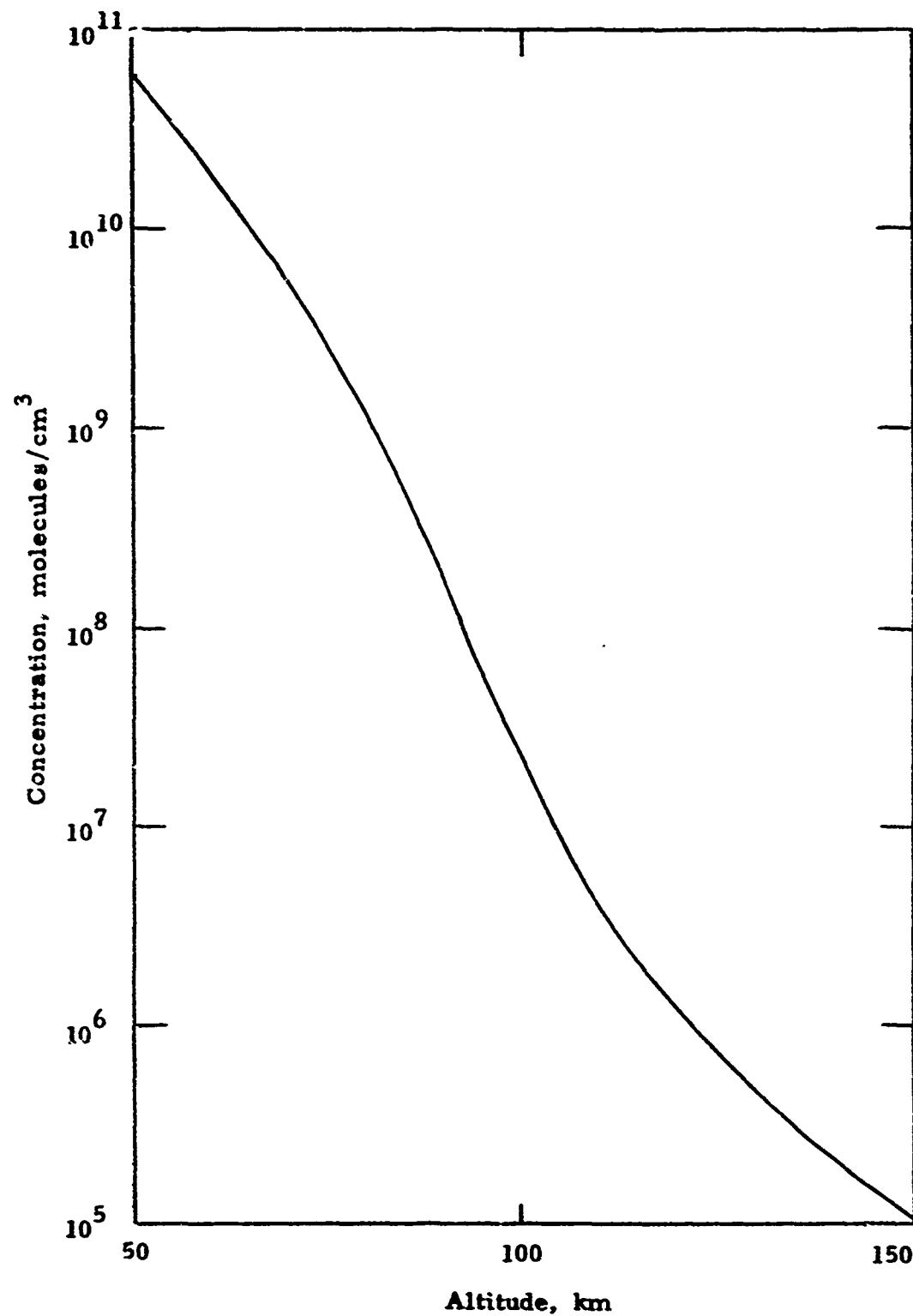


FIGURE II-1

Water Vapor Concentration Profile



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 II-3, and were obtained from the chemistry program.

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 above 100 km is 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. While Hays and Olivero (1970) have obtained a rapid falloff in mixing ratio above 80 km, their results are incorrect because they neglect recombination by Reactions R33 and R34 above 70 km. While the chemistry program was not run long enough to reach realistic values of carbon dioxide concentrations, the profile shown in Figure II-4, based in part on Offerman and von Zahn (1971) should be adequate as a basis for computation of infrared radiances.

Although included in the chemistry program, methane is not considered as an important source of infrared radiation above 50 km, as it is destroyed by photodissociation and reaction with atomic oxygen and OH faster than it can be transported upward.

Ozone Concentration Profiles

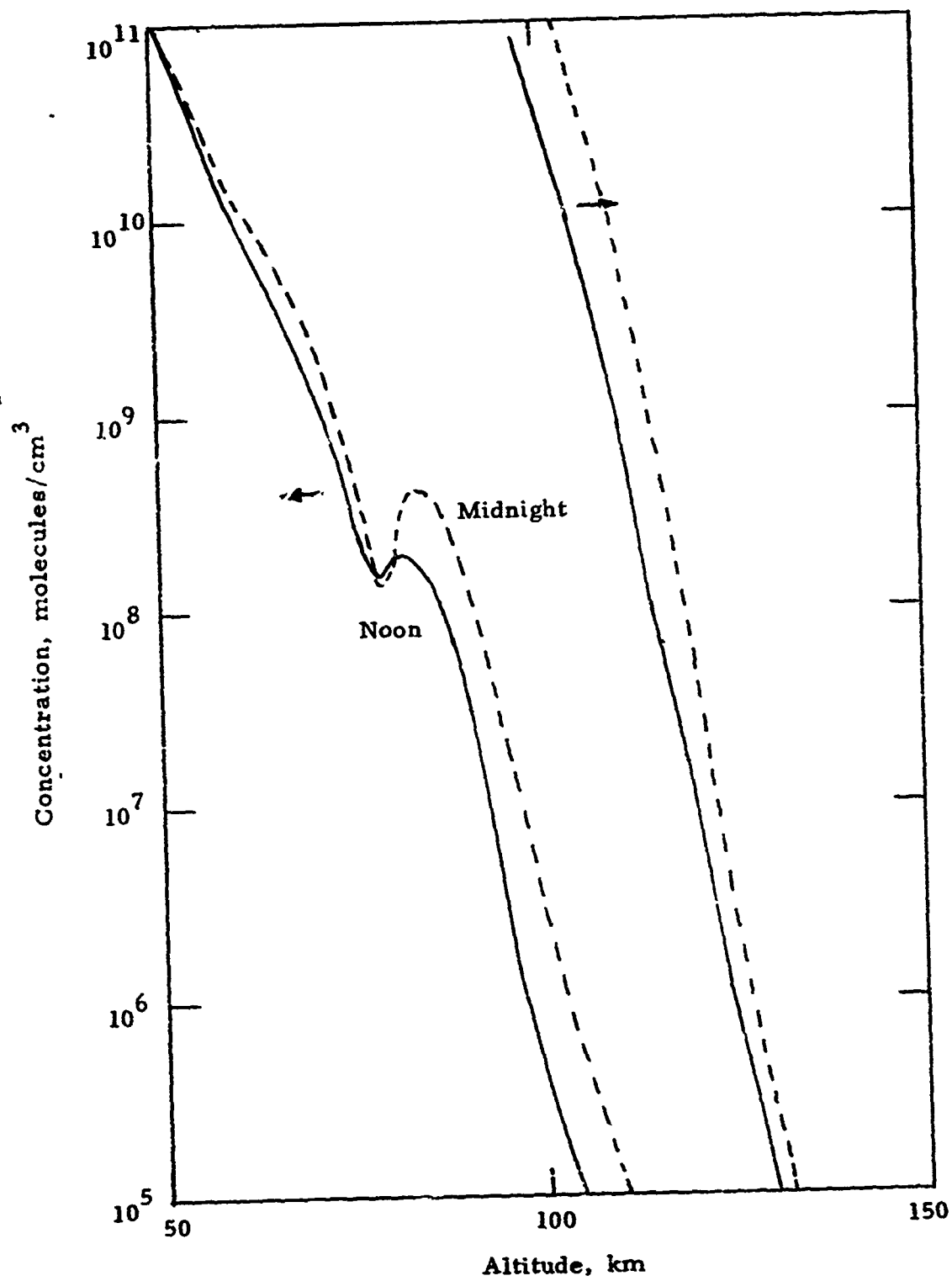
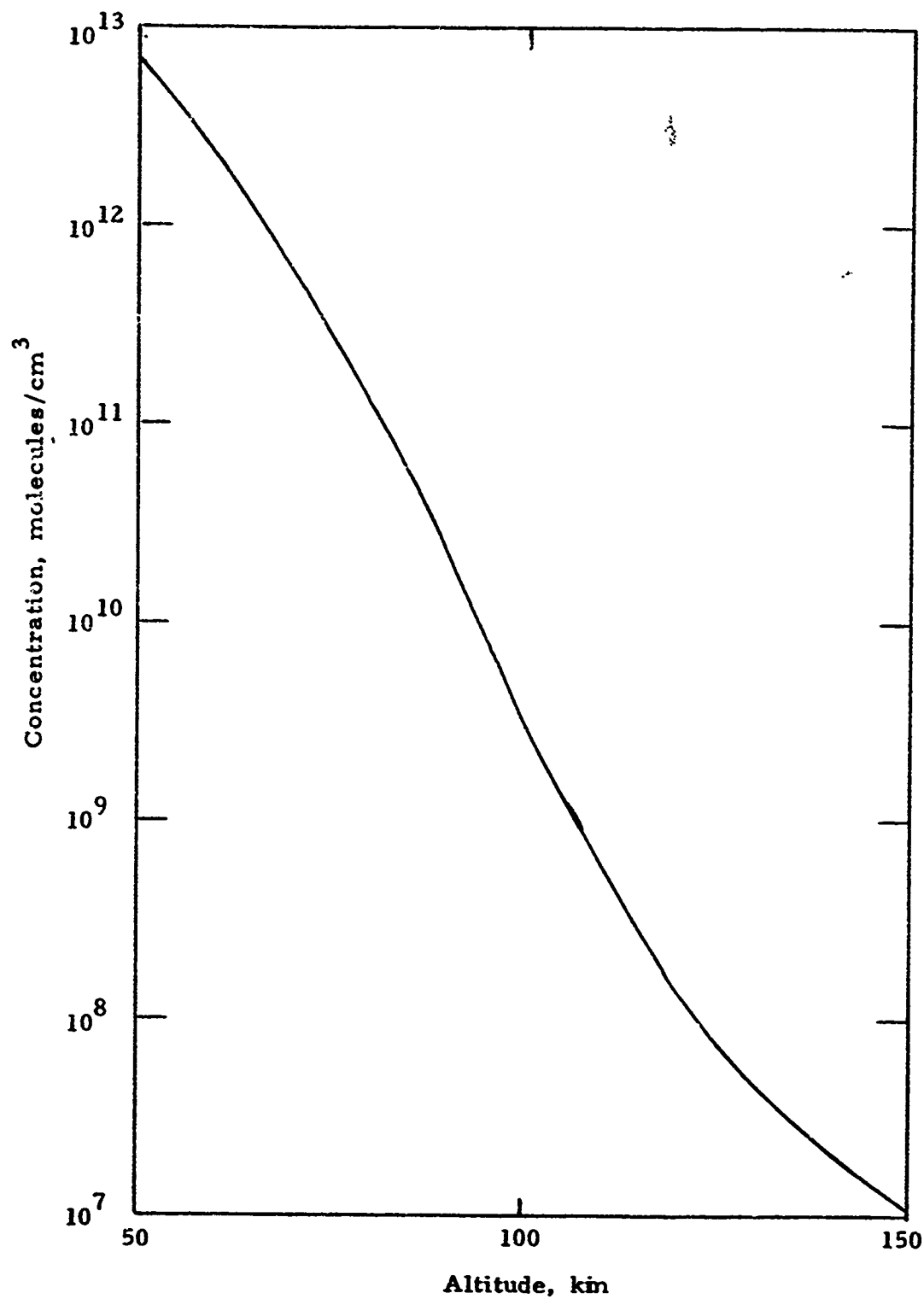


FIGURE II-3

Carbon Dioxide Concentration Profile



SECTION III

EXCITATION MECHANISMS

A. General Considerations

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 physical processes affecting the population of vibrational states includes collisional excitation, radiative excitation by resonant absorption or fluorescent decay, chemical formation in a vibrationally excited state, and radiative and collisional de-excitation.

In the troposphere and lower stratosphere collisional excitation of vibrational levels is sufficiently rapid to compensate for radiative de-excitation, and vibrational populations are in thermodynamic equilibrium with translation and rotation. At altitudes above 30 to 50 km, however, radiative processes assume a dominant role in determining vibrational populations. Resonant scattering and fluorescent mechanisms become the dominant processes producing infrared radiation. At wavelengths greater than $5\text{ }\mu\text{m}$, the most intense source of radiation is the thermal emission from the earth's surface and from carbon dioxide, water vapor and ozone present at lower levels in the atmosphere. During the day, the effects of solar radiation must be included since fluorescent processes may lead to significant excitation of vibrational states. At wavelengths shorter than $6\text{ }\mu\text{m}$, resonant absorption of solar infrared flux maintains vibrational temperatures above 200 K.

The following parts of this section discuss vibrational excitation and de-excitation and the model developed for computation of radiative processes.

B. Collisional Excitation

In computing radiances at high altitudes, it must be recognized that vibrational levels are not populated at thermal equilibrium values. Radiation de-populates vibrational states rapidly and collisional excitation is not sufficiently efficient to maintain a Boltzmann distribution of vibrational states. Collisional excitation and de-excitation rates are required to compute the balance of opposing mechanisms which determine actual vibrational populations.

Four collisional processes are considered which affect the vibrational state of a molecule. (1) translational-vibrational (T-V) interactions in which translational 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, (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, and (4) collisions between an atom and molecule in which there is either atom exchange or the formation of a collision complex which has a lifetime long enough to equilibrate kinetic and vibrational temperatures. This latter process may be considered a special case of the first mechanism.

When a molecule in the ground state collides with another molecule there is a finite probability P_{VT} 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_{TV} 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,

$$P_{TV} = P_{VT} \frac{g_u}{g_l} \exp \left(- \frac{hc}{\lambda_o kT} \right) \quad (3-1)$$

Similar expressions hold for intermolecular and intramolecular V-V transfer.

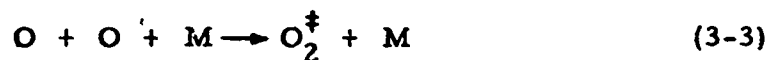
The rate at which a vibrational level is excited (k_{TV}) or de-excited (k_{VT}) is given by the product of the collision frequency Z and the transition probability. Experimental determinations of collisional excitation rates may be reported as probabilities, excitation rates or relaxation times. The latter are related to excitation and de-excitation rates by (Herzfeld and Litovitz, 1959)

$$\frac{1}{\tau} = k_{VT} + k_{TV} \quad (3-2)$$

Excitation rates depend on temperature and number density through the strong temperature variation of the transition probability and the square root temperature and linear density variation of the collision frequency.

The most important molecular collision partners in the atmosphere are nitrogen and oxygen, and a knowledge of their vibrational temperatures and effects is important. These will be treated first and followed by a brief discussion of the relative importance of collisional excitation mechanisms for nitric oxide.

Vibrationally excited oxygen can be produced in the atmosphere by recombination of atomic oxygen



by formation from ozone



by fluorescence in the discrete Schumann-Runge band system



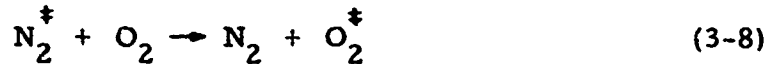
by transfer of kinetic energy to vibrational energy



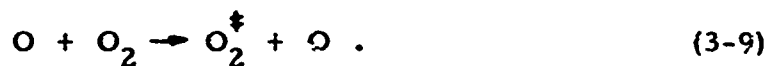
by electron impact



by exchange of vibrational quanta with molecular nitrogen (and other molecules)



and by the strong collisional interaction with atomic oxygen



Vibrationally excited oxygen can be quenched by the strong collisional interaction with atomic oxygen



by transfer of vibrational quanta to molecular nitrogen (and other molecules)



and by transfer of vibrational energy to kinetic energy



Above the mesopause, and up to 300 km, reactions (3-9) and (3-10) dominate production and loss of vibrational excitation and can be shown to maintain vibrational distributions close to thermal equilibrium values.

Bates and Moiseiwitsch (1956) proposed that the atom exchange reaction



is sufficiently rapid in quenching vibrationally excited oxygen that the reaction proposed by Krassovsky (1955)



cannot be important in the production of vibrationally excited OH. Bauer and Tsang (1963) suggested that reaction (3-14) would be important in relaxing the

vibrational levels of shock heated oxygen and estimated the rate coefficient to be about $2 \times 10^{-10} \exp(-8000/RT) \text{ cm}^3/\text{sec}$.

The rate for atom exchange reaction (3-14) has been measured near room temperature by Ogg and Sutphen (1953, 1954), Brennan and Niki (1965), Jaffe and Klein (1966) and Herron and Klein (1964). Johnston (1968) has reviewed these results and recommends a value of $10^{-12} \text{ cm}^3/\text{sec}$ for the rate coefficient at 273 K. Garnett, Kistiakowski and O'Grady (1969) have measured the rate coefficient near 1800 K. Clark, Garnett and Kistiakowski (1970) have combined the latter results with those of Jaffe and Klein (1966) to suggest a temperature dependence given by

$$k_{13} = 10^{-12.48 \pm 0.3} T^{1/2} \exp[-(960 \pm 700)/RT] \text{ cm}^3/\text{sec} \quad (3-15)$$

Kiefer and Lutz (1967) have measured the vibrational relaxation time of molecular oxygen in the presence of atomic oxygen at temperatures between 1600 and 3300 K. They find that over this temperature range the product of pressure P and relaxation time τ_{O_2} is given by

$$P\tau_{O_2} = (4.35 \pm 0.19) \times 10^{-8} - (7.75 \pm 0.81) \times 10^{-12} T \text{ atm-sec} \quad (3-16)$$

The prediction of Bauer and Tsang (1963) agrees with this result within a factor of two. The rate coefficient for atom exchange from Equation (3-15) is $9.8 \times 10^{-12} \text{ cm}^3/\text{sec}$ at 1600 K and the de-excitation rate coefficient derived from Equation (3-16) is $8.6 \times 10^{-12} \text{ cm}^3/\text{sec}$ at 1600 K. The close agreement

supports the assumption used here that the vibrational de-excitation rate is within a factor of two of the atom exchange rate, which may be considered well determined.

Further evidence is given by the theoretical computation of Breig (1969) who estimated the rate of de-excitation of high vibrational levels of molecular oxygen by reaction with atomic oxygen. He obtains total reaction rates of the $V = 5$ and $V = 25$ levels at 200 K. Breig's (1969) results may be used as a qualitative argument for the further assumption that higher vibrational levels of molecular oxygen will reach steady state populations in times of the order of that required by the $V = 1$ level.

Dalgarno (1963, 1967, 1970) has suggested that fluorescence in the discrete Schumann-Runge band system may be an important source of vibrationally excited oxygen in the mesosphere and thermosphere. The experimental results of Ackerman, Biaume and Nicolet (1969) and Hudson and Carter (1969) indicate that excitation to $V' > 3$ results in predissociation rather than fluorescence. If this conclusion is accepted, the only transition of importance is the $(2, 0)$ transition at $1971 \text{ \AA}$. This has an oscillator strength f in absorption of 2.34×10^{-4} according to the computations of Nicholls (1964). The solar flux πF_o at $1971 \text{ \AA}$ is about 1.2×10^{11} photons/($\text{cm}^2\text{-sec-\AA}$) according to Detwiler, et al. (1961) or 1.4×10^{11} photons/($\text{cm}^2\text{-sec-\AA}$) according to Ackerman (1970). At the top of the atmosphere, the absorption rate g per oxygen molecule in the $(2, 0)$ band is 9.6×10^{-7} photons per second from the relation

$$g = \frac{\pi e^2}{mc^2} \pi F_o f \lambda^2 \quad (3-17)$$

Using the Einstein A coefficients given by Nicholls (1964), it is found that the mean vibrational level following fluorescent emission is 11 so that about 10^{-5} vibrational quanta are produced each second per molecule.

The effective rates of vibrational excitation by electron impact may be determined from the electron cooling rates computed by Lane and Dalgarno (1969). Using the electron densities and temperatures observed by Spencer, et al. (1965), typical rates are 3.3×10^{-6} , 1.8×10^{-5} and 6.2×10^{-5} excitations per molecule per second at 150, 200 and 300 km, respectively.

The effects of the above processes on molecular oxygen vibrational temperatures were computed for the CIRA 1965 mean atmosphere for altitudes between 80 and 300 km. The rate coefficients given by Johnston (1968) were used for three body recombination of atomic oxygen and for the reaction of atomic oxygen with ozone. The maximum possible vibrational excitation of the products was assumed. The rate coefficients given by Gilmore, Bauer and McGowan (1969) were used for vibrational excitation by collisional transformation of kinetic energy and collisional exchange of vibrational quanta with molecular nitrogen. The de-excitation rate by reaction with atomic oxygen was taken to be $3.3 \times 10^{-13} T^{1/2} \exp(-485/T) \text{ cm}^3/\text{sec}$, the mean value suggested by Clark, Garnett and Kistiakowski (1970).

Even with the upper limits to nitrogen vibrational temperatures given by Walker, Stolarski and Nagy (1969) the rate of vibrational excitation of molecular oxygen by collisional transfer of energy with nitrogen is less than 10^{-6} excitations per molecule per second at any altitude above 80 km.

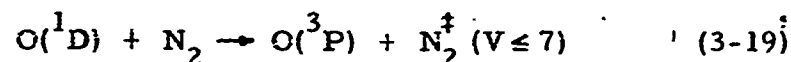
At all altitudes below 200 km, the vibrational population of molecular oxygen as computed is within two per cent of the thermal equilibrium value. At 300 km, excitation by electron impact increases the population of the first vibrational level by less than twenty per cent. Thus, at all altitudes where

polar molecules will be significantly excited by molecular oxygen, the vibrational temperature may be taken to be the kinetic temperature. This result agrees with the conclusions of Bauer, Kummeler and Bortner (1971).

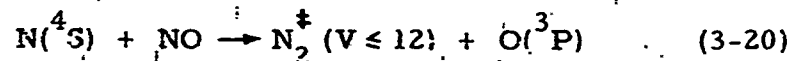
The vibrational temperature of molecular nitrogen at high altitudes remains uncertain. The principal processes which have been suggested to produce vibrationally excited nitrogen are transfer of kinetic energy to vibrational quanta



quenching of $\text{O}(^1\text{D})$ by collision



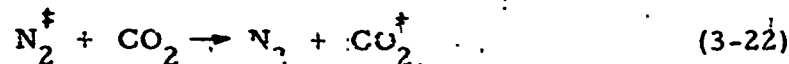
reaction of atomic nitrogen and nitric oxide



and excitation by electron impact



The principal process which quenches vibrationally excited nitrogen is collisional transfer of vibrational quanta to the ν_3 vibrational mode of carbon dioxide



Reaction (3-18) is insignificant above 70 or 80 km. If all the available energy were transformed to vibrational excitation, quenching of $O(^1D)$ would be the most important process above 80 km. However, Fisher and Bauer (1970) estimate that the process is only about five per cent efficient and Black and Eckstrom (1971) have obtained an experimental efficiency of 8.3 ± 6.5 per cent. Above 120 km, reaction (3-20) is the most important source of excitation, Morgan, et al. (1962) and Morgan and Schiff (1963) having determined that about one-third of the available energy goes into vibrational excitation. Reaction (3-21) is nearly as an important source, the electrons involved being those produced by photo-ionization in the E-region.

Walker (1968) and Walker, Stolarski and Nagy (1969) have computed theoretical estimates of nitrogen vibrational temperatures. The first paper neglected collisional transfer of vibrational energy to carbon dioxide and obtained an approximately constant vibrational temperature of 3000 K above 110 km. Corbin, et al. (1970) noted that this result was too high near 110 km, more energy being transferred to carbon dioxide than could be produced by the assumed excitation mechanisms. The second paper included quenching by carbon dioxide and recognized that the $O(^1D)$ excitation mechanism might be inefficient. Results were given for zero and 100 per cent excitation by this mechanism. The first assumption gives a maximum vibrational temperature of 1700 K at 300 km. The vibrational temperature is below 1000 K at all times for altitudes below 140 km. Minimum vibrational temperatures occur at dawn and rise steadily until sunset. Using these results, it is found that except for carbon dioxide band emissions originating from the ν_3 mode, excess vibrational excitation of nitrogen is unimportant for radiation emitted by the molecules studied here.

Degges (1971) has discussed the mechanisms which contribute to

vibrational excitation of nitric oxide. He proposes that above 80 km, the most important collisional process is the atom exchange mechanism



Using the low temperature measurement of Herron and Klein (1964) and an argument based on the adiabatic correlation diagram of Donovan and Husain (1970) and the temperature dependence of the similar atom exchange reaction of atomic oxygen with carbon monoxide, he suggests that the de-excitation rate of vibrationally excited nitric oxide by collision with atomic oxygen lies between $10^{-13} T^{1/2} \text{ cm}^3/\text{sec}$ and $5 \times 10^{-13} T^{1/2} \exp(-1000/RT) \text{ cm}^3/\text{sec}$. The latter rate has been adopted in the present computations. Figure III-1 compares the effect of this process with exchange of vibrational quanta with molecular oxygen and nitrogen. Even if the nitrogen vibrational temperature is 2000 K, nitrogen is a less efficient collision partner above 120 km.

The rates used for other vibrational excitation processes are largely the same as those used by Corbin, et al. (1969). Vibrational exchange rates between nitric oxide and nitrogen are those measured by Basco, et al. (1961, 1962) with the temperature dependence given by Gilmore, et al. (1969) which is based on the correlation scheme of Millikan and White (1963). Collisional excitation rates used for carbon dioxide and water vapor are taken from the review of Taylor and Bitterman (1969), with the addition of the experimental results of Stephenson et al. (1971) on intramolecular vibrational exchange in carbon dioxide. The rates used for nitrous oxide are those of Yardley (1968) and Simpson, et al. (1968). Estimated rates for ozone excitation were obtained by use of Millikan and White's (1963) correlation scheme, and comparison with other systems. Because of the lower temperatures at the mesopause (about 80 to 90 km) and the rapid decrease in number density with altitude, the only critical rates above 80 km are the oxygen atom exchange with nitric oxide and the vibrational exchange between carbon dioxide and molecular nitrogen.

Nitric Oxide Collisional Excitation Rates

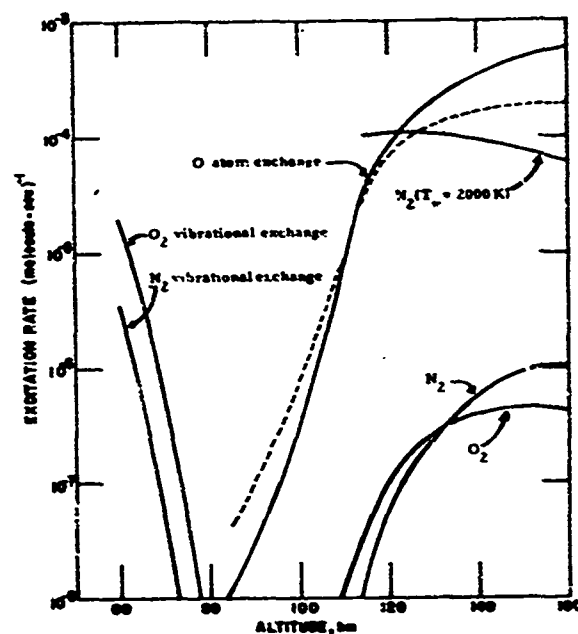
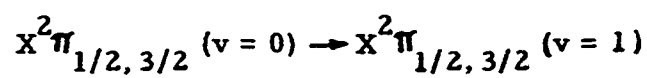


FIGURE III-1

C. Radiative Processes and Modeling

The physical processes and assumptions used in developing a model for transport of infrared radiation in the upper atmosphere are discussed below, beginning with the interaction of electromagnetic radiation with an isolated atom or molecule. 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 is 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 Doppler shape. Finally, a model for the transport of radiation for a single vibration-rotation band is described.

When treating a problem in which the number of photon 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 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 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 l).

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 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/cm³, the coefficient B_{lu} has units cm³/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

$$A_{ul} = 8\pi h B_{ul} / \lambda^3 \quad (3-24)$$

and

$$g_l B_{lu} = g_u B_{ul} \quad (3-25)$$

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,

$$S'_{lu} = N_l \frac{A_{ul} \lambda_{lu}^2}{8\pi} \frac{g_u}{g_l} \left[1 - \exp\left(-\frac{hc}{\lambda_{lu} kT}\right) \right] \quad (3-26)$$

or as line or band strengths

$$S_{lu} = S'_{lu}/pc \quad (3-27)$$

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

$$S'_{lu} = \frac{\pi e^2}{mc} N_l f_{lu} \left[1 - \exp \left(-\frac{hc}{\lambda_{lu} kT} \right) \right] \quad (3-28)$$

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

$$A_{ul} = \frac{64\pi^4}{3h\lambda^3} \frac{g_l}{g_u} |\mu_{ul}|^2 \quad (3-29)$$

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):

$$\begin{aligned} G &= \frac{\pi e^2}{mc} I_o f_o \lambda_o^2 \quad (\text{photons/sec}) \\ &= 8.852 \times 10^{-13} I_o f_o \lambda_o^2 \quad (\text{photons/sec}) \end{aligned} \quad (3-30)$$

where I_0 is the continuum source flux and f_0 is the oscillator strength. If the incident flux is expressed in photons/cm²-μm, and oscillator strength is replaced by the equivalent line or band strength S expressed as cm⁻²atm⁻¹,

$$G = 3.72 \times 10^{-24} I_0 S \lambda^2 \quad (3-31)$$

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

$$A_{ul} = 2.804 S (\text{cm}^{-2} \text{atm}^{-1}) / \lambda^2 (\mu\text{m}). \quad (3-32)$$

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,

$$G = 1.169 \times 10^{-23} N_\lambda S \lambda^2 \quad (3-33)$$

Another purely radiative mechanism of importance in upper atmosphere radiance computations is fluorescence. A striking example occurs for the weak 10.4 μm CO₂ band. Absorption of solar radiation by CO₂ 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 states is:

$$P_{un} = \frac{A_{un}}{\sum_{j=1}^k A_{uj}} \quad (3-34)$$

where the A_{uj} are the Einstein A coefficients for the spontaneous emission probabilities for a transition from the upper state to the j th 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 ($v'' = 0$) or to other vibrational levels ($v'' \neq 0$) in the ground electronic state. The rate of resonance fluorescent scattering in a given electronic vibrational transition may be obtained by combining Equations (3-30) and (3-34) and introducing appropriate subscripts:

$$G_{v'v''} = 8.853 \times 10^{-17} I_{v'o} f_{v'o} \lambda_{v'o}^2 \frac{A_{v'v''}}{\sum_{v''} A_{v'v''}} \quad (3-35)$$

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 $v'0$ 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.

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 escaping from an optically thick layer of gas. Thus, for large optical thicknesses (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 basic computational model therefore assumes a Doppler profile.

In addition to the use of Doppler shape for a single line, several other assumptions have been made about the radiative transfer within lines of a band. These are the following:

- (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. This assumption probably constitutes the largest source of error in computing transmission.

The major result is to make computed band radiances too small when determining escape of radiation from a warm layer overlaid by a cooler layer.

The above assumptions lead to the following treatment, which begins with consideration of a single line and from this develops the method used for treating radiative transfer in an entire band. The spectral absorption coefficient of a Doppler broadened line is (Mitcheli and Zemansky, 1961):

$$k(\nu) = \frac{1}{\Delta\nu_D} \sqrt{\frac{\ln 2}{\pi}} \frac{\pi e^2}{mc} N_l f_o \exp \left[-\left(\frac{\nu - \nu_o}{\Delta\nu_D} \sqrt{\ln 2} \right)^2 \right] \quad (3-36)$$

where the Doppler half width at half maximum is

$$\Delta \nu_D = \sqrt{\frac{2RT \ln 2}{Mc^2}} \nu_0 (\text{sec}^{-1})$$

The amount of light absorbed by the line per unit frequency interval is

$$\begin{aligned} I(\nu) A(\nu) d\nu &= I(\nu) (1 - T(\nu)) d\nu \\ &= I(\nu) (1 - e^{-k(\nu)}) d\nu \end{aligned} \quad (3-37)$$

and the total light absorbed is

$$\begin{aligned} \int_0^\infty I(\nu) A(\nu) d\nu &= \int_0^\infty I(\nu) (1 - T(\nu)) d\nu \\ &= \int_0^\infty I(\nu) \left(1 - \exp \left\{ -K_0 \exp \left[-\left(\frac{\nu - \nu_0}{\Delta \nu_D} \ln 2 \right)^2 \right] \right\} \right) d\nu \end{aligned} \quad (3-38)$$

where the integration is over all frequencies and

$$\kappa_o = \frac{1}{\Delta\nu_D} \sqrt{\frac{\ln 2}{\pi}} \frac{\pi e^2}{mc} N_1 f_o$$

The line will absorb only near the center frequency ν_o and $I(\nu)$ may be replaced by $I(\nu_o)$. Making the substitution

$$w = \frac{\nu - \nu_o}{\Delta\nu_D} \sqrt{\ln 2}$$

$$\int_0^\infty I(\nu) A(\nu) d\nu = I(\nu_o) \frac{\Delta\nu_o}{\sqrt{\ln 2}} \int_{-\infty}^\infty \left(1 - e^{-\kappa_o e^{-w^2}} \right) dw$$

$$= I(\nu_o) \frac{\pi e^2}{mc} N_1 f_o \left(1 - \frac{\kappa_o}{2! \sqrt{2}} + \frac{\kappa_o^2}{3! \sqrt{3}} + \dots \right) \quad (3-39)$$

The corresponding expression on a wavelength basis is

$$\begin{aligned} \int_0^\infty I(\lambda) A(\lambda) d\lambda &= I(\lambda_o) \frac{\pi e^2}{mc} N_1 f_o \lambda_o^2 \left(1 - \frac{\kappa_o}{2! \sqrt{2}} + \frac{\kappa_o^2}{3! \sqrt{3}} + \dots \right) \\ &= I(\lambda_o) \kappa_o \left(1 - \frac{\kappa_o}{2! \sqrt{2}} + \frac{\kappa_o^2}{3! \sqrt{3}} + \dots \right) \\ &= GN_1 \left(1 - \frac{\kappa_o}{2! \sqrt{2}} + \frac{\kappa_o^2}{3! \sqrt{3}} + \dots \right) \end{aligned} \quad (3-40)$$

The quantity in parentheses in Equation (3-40) is simply a factor by which the product of G and N_1 must be multiplied to account for the effects of total optical thickness κ_0 in computing the total absorption of radiation by a column containing N_1 molecules/cm² in the lower state. This quantity is termed S by Mitchell and Zemansky (1961) who present tabulations of its value. Penner (1959) gives an asymptotic expansion for large values of κ_0 . The astrophysical literature frequently uses the notation

$$L(\kappa_0) = \kappa_0 S(\kappa_0) = \frac{2}{\sqrt{\pi}} \int_0^{\infty} \left(1 - e^{-\kappa_0 e^{-w^2}} \right) dw \quad (3-41)$$

The factor by which G must be multiplied to obtain the true rate of absorption at an optical depth κ_0 may be found by differentiating Equation (3-41) with respect to κ_0 (Ivanov and Shcherbakov, 1965):

$$\begin{aligned} M_1(\kappa_0) &= \frac{dL(\kappa_0)}{d\kappa_0} \\ &= \frac{2}{\sqrt{\pi}} \int_0^{\infty} e^{-w^2} - \kappa_0 e^{-w^2} dw \end{aligned} \quad (3-42)$$

The factor by which G must be multiplied to obtain the actual absorption from an extended black body source is (Ivanov and Shcherbakov, 1965)

$$N_1(\kappa_0) = \int_0^{2\pi} d\phi \int_0^{\pi/2} \cos \theta \sin \theta M_1\left(\frac{\kappa_0}{\cos \theta}\right) d\theta \quad (3-43)$$

When the source of radiation is a Doppler profile exactly that of the absorbing line, the differential absorption at optical thickness κ_0 is given by

$$M_2(\kappa_0) = \frac{2}{\sqrt{\pi}} \int_0^{\infty} e^{-2w^2 - \kappa_0 w^2} dw \quad (3-44)$$

and G must be multiplied by this factor. Also, if the source is an infinite plane layer (not a surface) radiating with a Doppler profile, G must be multiplied by

$$N_2(\kappa_0) = \int_0^{2\pi} d\phi \int_0^{\pi/2} M_2\left(\frac{\kappa_0}{\cos \theta}\right) \sin \theta d\theta \quad (3-45)$$

Corbin, et al. (1970) evaluated the functions of Equations (3-41) through (3-45) by numerical integration and obtained polynomial approximations for the entire range of variation of x_0 from zero to infinity, using the known asymptotic forms as a guide at large values. They then used these functions to compute the corresponding functions for vibration-rotation bands of linear molecules. The resulting functions are conveniently parameterized by using the optical thickness at the center of the strongest line of a band. It was found that two sets of such functions were adequate, one set for parallel transitions and one set for perpendicular transitions.

In extending the modeling to non-linear molecules, it was found to be more convenient to use the number of molecules in the line of sight as the

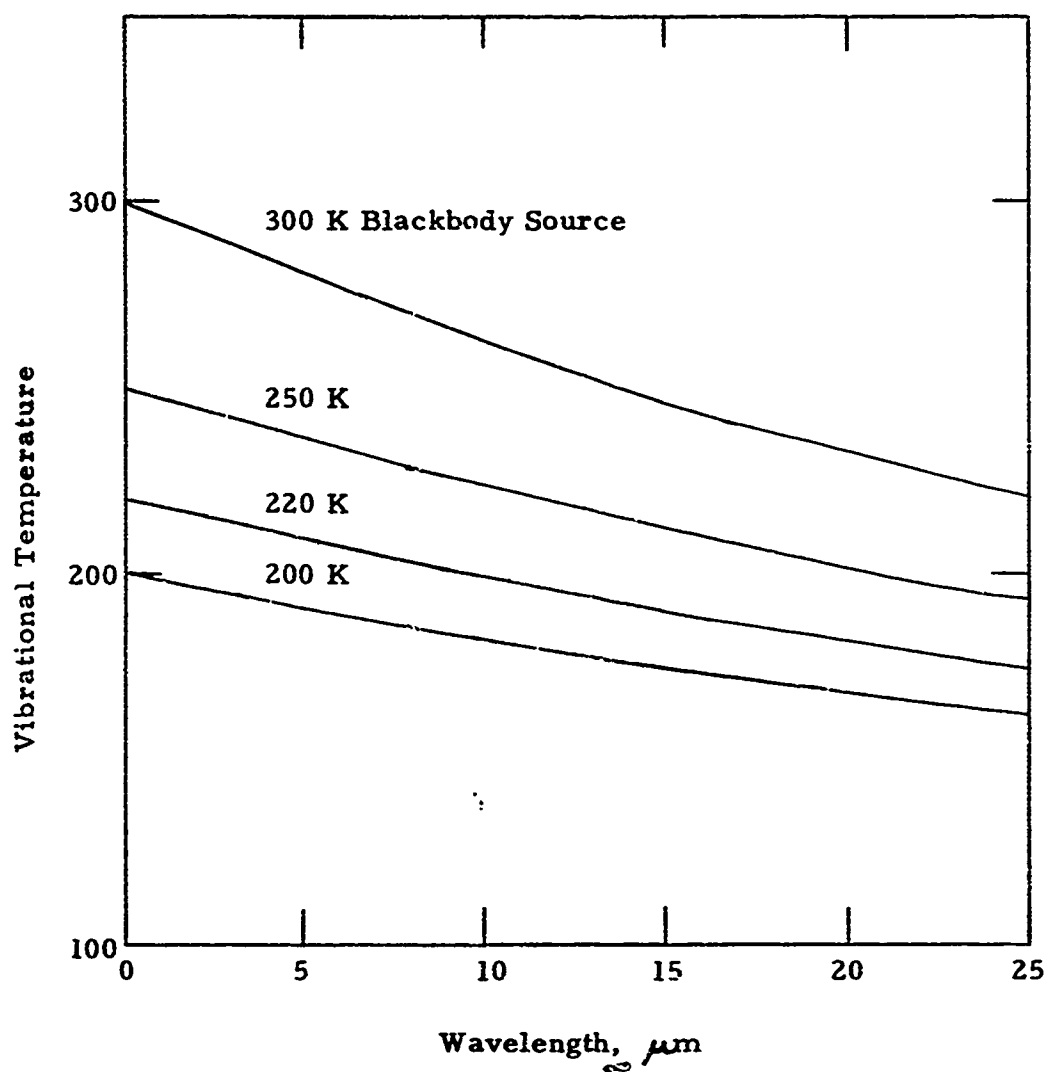
model parameter. Since upper atmosphere temperatures are near or below 250 K where optical thickness affects results, the functions used in the program for the water vapor ν_2 and ozone ν_1 and ν_3 bands were computer for a temperature of 250 K. Because ozone and water vapor are present in much smaller concentrations than carbon dioxide, the effects of optical thickness are much smaller.

At altitudes where collisional excitation rates are small, particularly between 80 and 100 km and above 120 km, the radiation from the lower atmosphere and the sun will control vibrational excitation of infrared emitting molecules. Figure III-2 shows vibrational temperatures which will be maintained by radiation from the earth's surface and lower atmosphere assuming that the upward radiation may be treated as originating from a blackbody at a given temperature. The effective temperature will vary with wavelength and lower atmosphere temperature and composition. Curves are drawn for blackbody temperatures of 200, 220, 250 and 300 K. Typical radiation temperatures between 5 and 25 μm lie between 190 and 310 K (E. G. Wark, et al., 1962). The main feature of importance in these curves is the decrease in vibrational temperature with increasing wavelength for a given blackbody temperature.

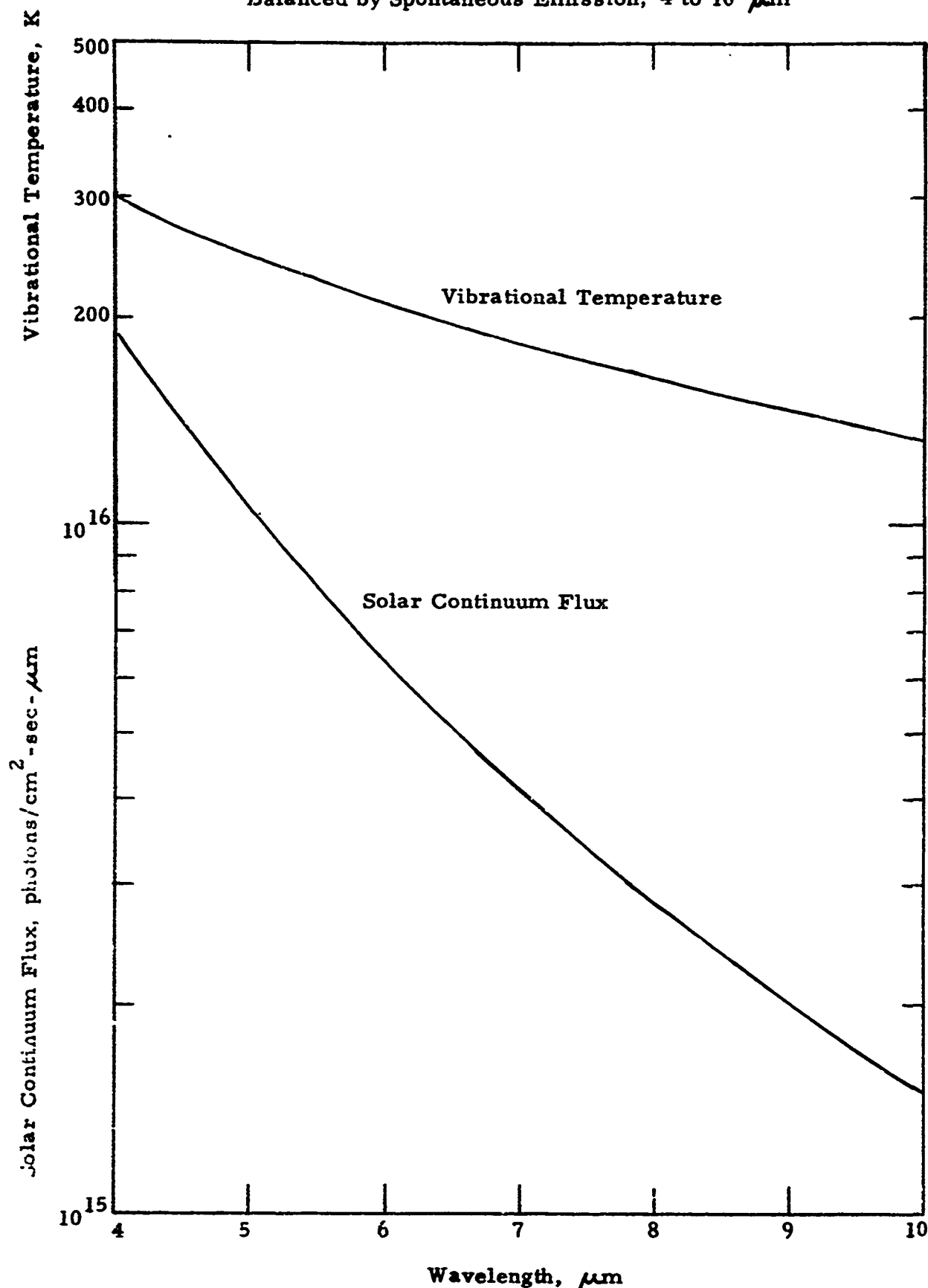
During the day, solar infrared flux is important in exciting the shorter wavelength transitions. Figure III-3 shows solar infrared fluxes at the top of the atmosphere given by Johnson (1964) and the resulting vibrational temperatures for a molecule at the top of the atmosphere. Absorption of radiation by molecules between the lower atmosphere radiation source or the sun decreases the effective incident radiation. This is partially, but not completely, offset by the radiation of these molecules.

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

Vibrational Temperature of Molecule with
Excitation by Infinite Plane Blackbody Source
Alone Balanced by Spontaneous Emission,
0 to 25 μm



Solar Continuum Flux and Corresponding Vibrational Temperature of a Molecule at the Top of the Atmosphere, Excitation by Solar Flux Alone
Balanced by Spontaneous Emission, 4 to 10 μm



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. The effective radiation from the lower boundary is determined by multiplying by the value of the band function $N_1^b(\kappa_0)$ corresponding to the line function of Equation (3-43). The effective radiation from the sun is determined by multiplying by the band function $M_1^b(\kappa_0)$ corresponding to the line function of Equation (3-42). Finally, the effective radiation from all atmospheric levels is obtained by integrating the product of emission at each level and the band function $N_2^b(\kappa_0)$, which corresponds to the line function of Equation (3-45). 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 $M_1^b(\kappa_0)$. 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 $L(\kappa_0)$, summing the line emission rates and scaling each line emission rate so that the total band intensity is obtained.

The use of band radiative transfer functions derived by assuming non-overlapping lines with Doppler shape does not lead to a completely satisfactory integration with the radiation field at lower altitudes. Gille and Ellingson (1968) have computed correction factors to make possible the inclusion of Doppler broadening effects in the computation of transmissivities of random exponential bands based on Lorentz line profiles. A rational approximation to their table of correction factors, obtained by a least squares gradient search on the parameters involved, has been investigated. Results were not good, partly because of the inaccuracy in computing transmissivity differences between

large values of absorber amount and partly because of the crudeness of the rational approximation, which in some cases led to differences with the wrong sign. Another problem with using such models is the difficulty of properly accounting for irradiation from an extended source (cf. Kuhn and London, 1969). Line-by-line integration with use of a Voigt line profile as done by Arnold, et al. (1969) is feasible when vibrational excitation is known beforehand and would be better but much slower than the present method of computing limb radiance spectral profiles.

Table III-1 lists the molecular bands included in the program. The band strengths listed for CO_2 , H_2O , O_3 and N_2O were adopted from a compilation by C. R. Calfee of ESSA which was provided by R. A. McClatchey (private communication, 1970) of AFCRL. These total band strengths agree with those obtained by summing the individual line strengths used by McClatchey, et al. (1970). The band strength of the fundamental of nitric oxide is an average of the values obtained by Abels and Shaw (1966), Feinberg and Camac (1967), Oppenheim, et al. (1967), Schurin and Clough (1963), and Varanasi and Penner (1967). The nitric oxide overtone value is that obtained by Schurin and Ellis (1966) and agrees with the theoretical calculation of Michels, et al. (1970).

D. Chemiluminescence

It is well known that some reactions such as that of atomic hydrogen with ozone to form OH lead to product molecules which are vibrationally excited and radiate in the infrared spectral region. Thus, Corbin, et al. (1969) assumed that nitric oxide produced by reaction R42 was produced with an average of one vibrational quantum. Hushfar, et. al. (1971) have determined the initial vibrational populations of nitric oxide in states higher than the first excited level. Their results indicate that at most one molecule in 50 is formed vibrationally excited to the second or higher vibrational level.

Table III-1

Molecular Band Strengths at 296K

Molecule	Band Center Wavelength, μm	Vibrational Transition*		Band Strength $\text{atm}^{-1}\text{cm}^{-2}$
		Upper State	Lower State	
CO_2	18.37	11102	10001	0.00675
	16.74	11102	02201	0.129
	16.18	10002	01100	3.56
	15.46	11102	10002	0.550
	14.98	01100	00000	205.0
	14.98	02200	01100	16.1
	14.97	03300	02200	0.948
	14.52	11101	10001	0.369
	13.87	10001	01100	4.59
	13.48	11101	02200	0.196
	12.64	11101	10002	0.0278
	10.41	00010	10001	0.0122
	9.40	00010	10002	0.0157
	5.18	11102	00000	0.0102
	4.82	11101	00000	0.0553
	4.26	00010	00000	2380.0
H_2O	6.27	010	000	290.0
O_3	9.60	001	000	320.0
	9.06	100	000	8.26
NO	5.33	1	0	112.0
	2.68	2	0	1.95
N_2O	17.26	0200	0110	1.73
	16.98	0110	0000	30.4
	16.98	0220	0110	1.73
	8.56	0200	0000	11.1
	7.78	1000	0000	216.0
	7.74	1110	0110	7.75
	4.58	0001	0000	1710.0

*Vibrational states are designated by ν , $\nu_1\nu_2\nu_3$, $\nu_1\nu_2l\nu_3$ or $\nu_1\nu_2l\nu_3f$, where ν_i is the quantum number of the i^{th} vibrational mode, l is the vibrational angular momentum quantum number and f (if greater than zero) identifies different components of a Fermi resonance multiplet.

If their fit of the distribution to a relative vibrational temperature of 5000 K holds for the first vibrational level also, then this reaction will be of negligible importance in producing infrared radiation. No experimental data exists on the vibrational state of nitric oxide formed by the more important reaction of $N(^2D)$ with molecular oxygen, and nitric oxide chemiluminescence is not included in the radiance model. Similarly, F. R. Gilmore (private communication to J. W. Carpenter, 1970) has suggested that ozone may be formed in highly excited vibrational states which may lead to radiation occurring in the ν_3 mode at longer wavelengths than the 001 to 000 transition. Again, there has been no experimental determination and refinement. Gilmore's estimates would require more information on both the initial state of the product ozone molecule and the relative transition probabilities of the competing radiative transitions that cascade to the ground vibrational state.

SECTION IV

DESCRIPTION OF PROGRAMS

This section describes the computer programs developed to implement the chemistry discussed in Section II and the computation of high altitude limb viewing infrared radiances employing the physical processes discussed in Section III. The chemistry program, including the effects of vertical eddy mixing and molecular diffusion, is completely new. The programs to compute radiance are modifications of those developed by Corbin, et al. (1969). An additional program has been written to selectively plot the output of the radiance program. The programs are described below and are listed in the Appendix.

A. Chemistry Program

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 Equation (2-18). 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 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 gravitational acceleration, cm/sec^2 , at computation location.
6. REFF Effective value of earth radius, km, for use in computing gravitational 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.

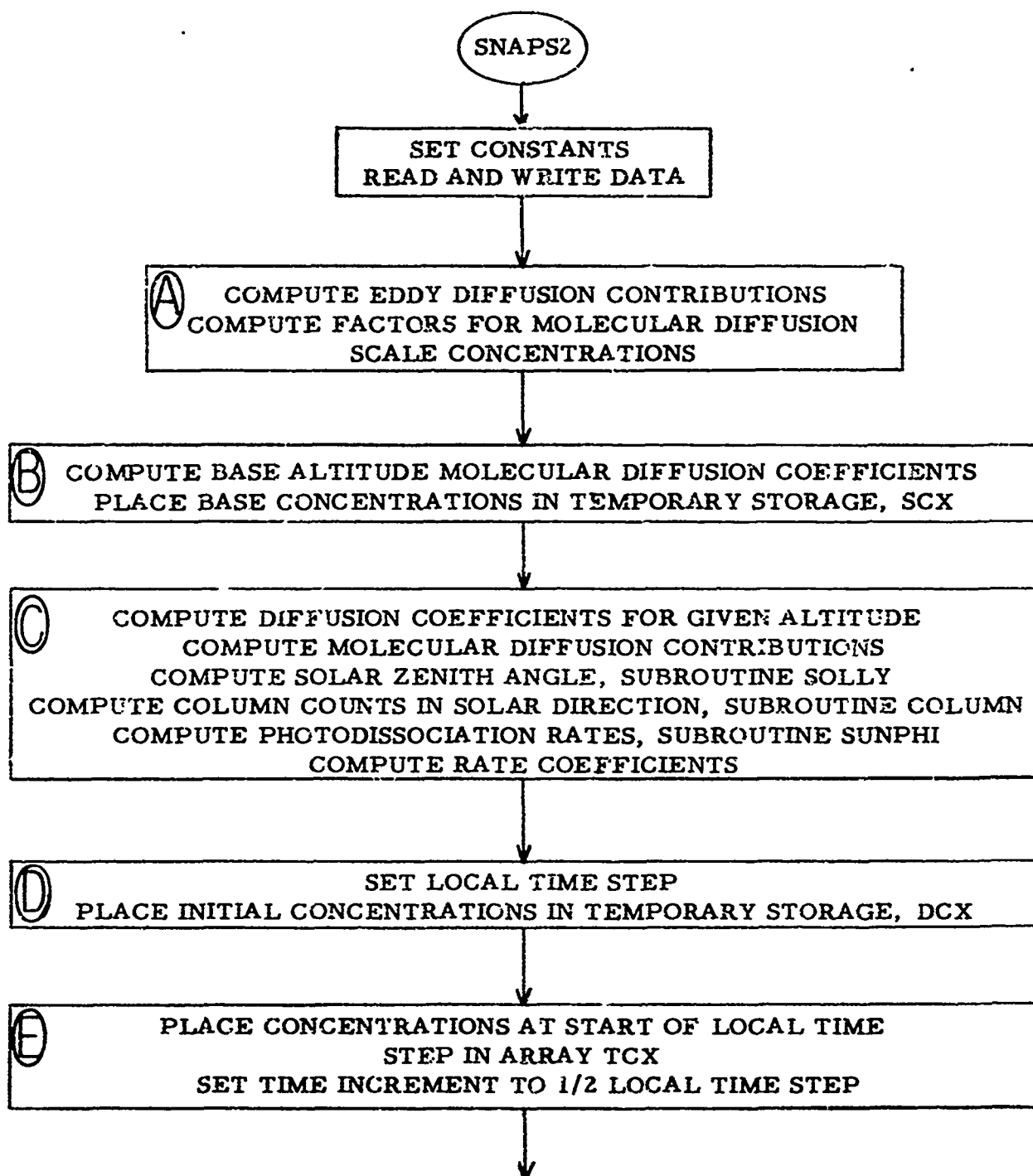
9. NBASE Index of base altitude for computations, altitude in km plus one.
10. NTOP Index of highest altitude in computations, altitude in km plus one, restricted to 201 or less in current program.
11. IHRMAX Number of times at which complete output is required.
12. MMAX Number of integration steps between 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; A.M. values are negative, P.M. values are positive.
15. CONSP The 603 cards containing concentrations and identifying indices; concentrations are read into array CONSP.

Program execution 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 array EDCDEF 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

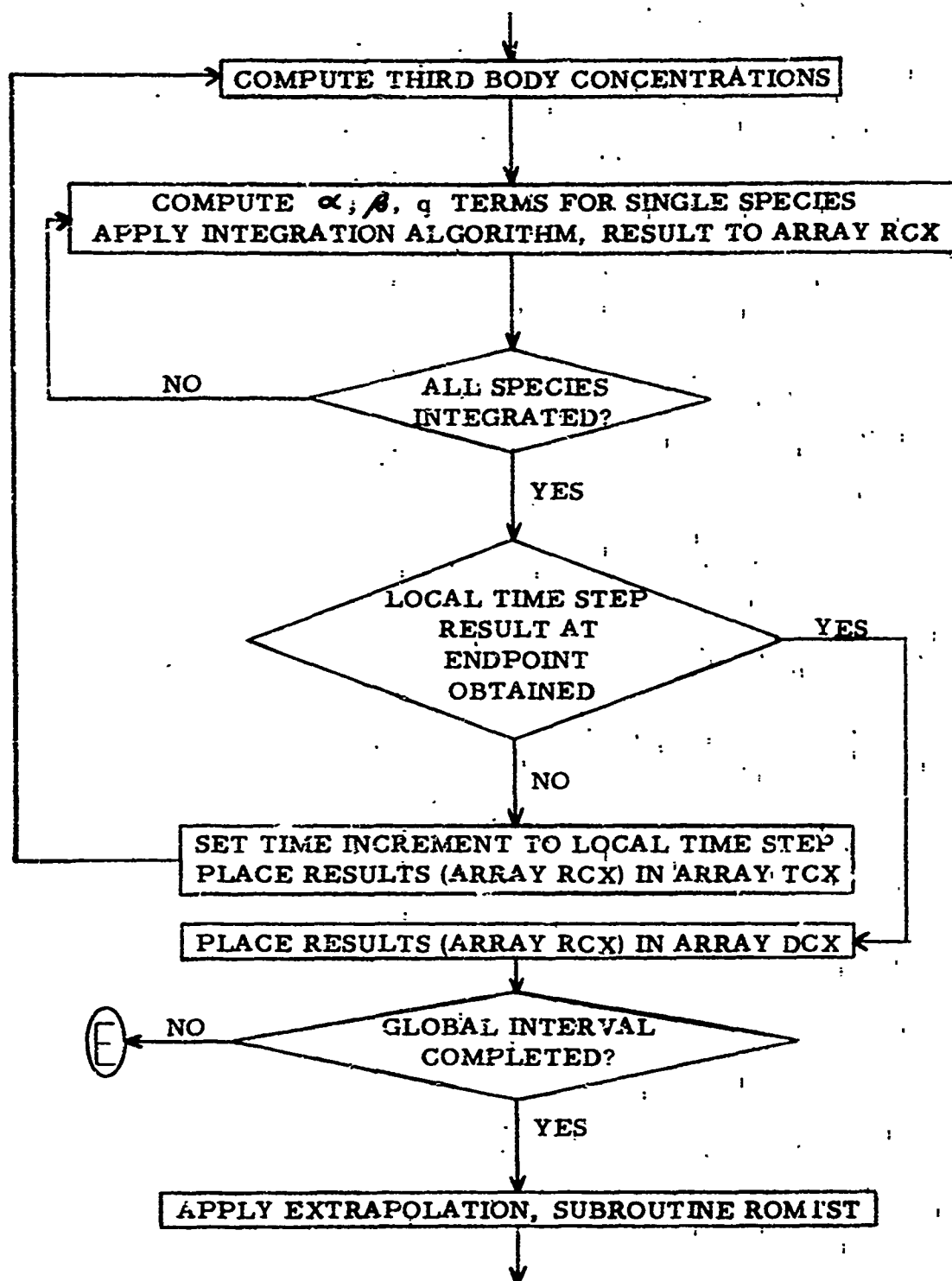
statements, and easy variation of the altitude of maximum turbulent transport. The data described above is then read in. The information on each data card is written to printer output channel 6 as it is read. Columns 13 through 80 of the second through fourteen data cards are printed in addition to the 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 following pages.

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 gravitational acceleration at each altitude is computed and local mixing scale heights H_{av} are computed. Coefficients for the eddy diffusion contributions to q and β terms in Equation (2-10) are determined from the finite difference approximation, Equation (2-8), 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.

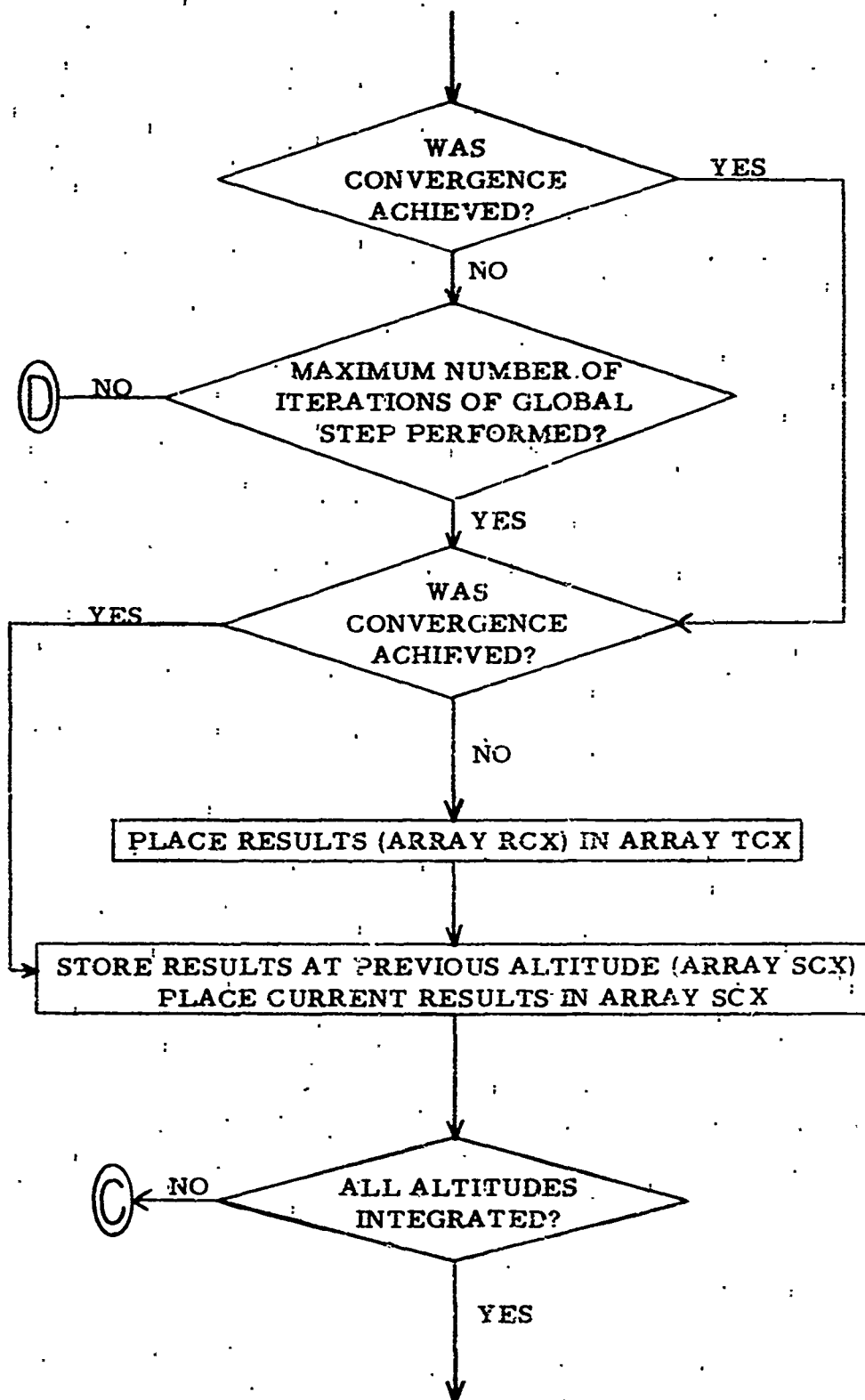
Flow Chart for Chemistry Program



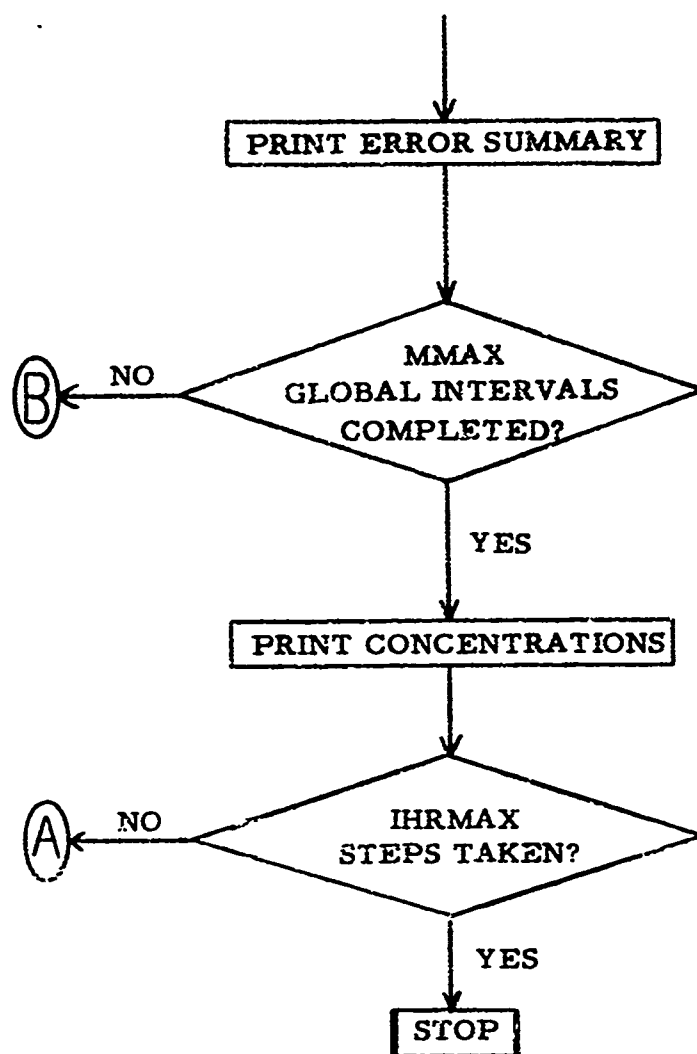
Flow Chart for Chemistry Program (continued)



Flow Chart for Chemistry Program (continued)



Flow Chart for Chemistry Program (continued)



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 q 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 MMAX 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 initial 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 q 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 endpoint 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 the 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 Laird, 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 subdivisions of the global time interval DLTIME. Further study is required before it will be possible to state optimum values of DLTIME and MLMAX. Since as many as 20 diurnal cycles may be required to reach concentrations which repeat from one simulated day to the next, a means of reducing running time is required. Possible methods are to rewrite the program and subroutines COLUMN, CONCEN and TKEL to use a larger altitude increment or to generate a starting atmosphere with the method of Colegrove, et al (1966). In its present form, the chemistry program SNAPS2 uses fixed concentrations of NO^+ , O_2^+ , O^+ and electrons during the day and assumes no ions to be present at night. The full ionic chemistry of Table 2-4 should be added and photoionization rates added to the photodissociation

rates computed by subroutine SUNPHI.

The names of the real arrays and their uses are:

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 q terms from next lower altitude
QJEDU	stores coefficients of eddy diffusion contributions to q terms from next higher altitude
QJEMD	stores values of contribution to q 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 q 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
TETOT	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
TINO	stores values of NO^+ concentrations at each altitude, set by DATA statement
TIO	stores values of O^+ concentrations at each altitude, set by DATA statement
TIO2	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

The names of the real variables and their uses are:

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, radians, computed by subroutine SOLLY
CHISAV	temporary storage for value of CHI
CMDSPN	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 Equation (2-10a)
FACM	temporary storage for value of $1 - \exp(-\delta t)$
FACRT	temporary storage for $4\alpha q/\beta^2$
FAC2AX	temporary storage for $2\alpha x_0$
FAC4AQ	temporary storage for $4\alpha q$

G	value of gravitational acceleration at altitude NALT
GL	value of gravitational acceleration at altitude NALT - 1
GREF	surface value of gravitational acceleration, read in as data
GU	value of gravitational acceleration at altitude NALT + 1
QCOHA	values of OH production rates at altitude
QCOHB	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 q 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
RII....	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	} storage for value of computation of computer running time
SECB	
SECC	
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
S11, ...	values of reverse ionic reaction rates
SOLDEC	solar declination angle, read in as data
SUM	} temporary storage used in forming mean molecular masses
SUML	
SUMU	
S1, ...	values of reverse neutral reaction rate coefficients
TCH4, ...	values of individual molecular concentrations in array TCX, names individually for convenience in forming α , β and q 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 q term for CO_2 and β term for CO

The names of integer variables and their uses are:

I	index in DO loop 95, auxiliary index in DO loop 110, and index in implied 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 MLth 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	} storage space used for sequence numbers of cards read in or punched out at a given altitude
N2	
N3	

B. Radiance Program

The high altitude radiance programs of Corbin, et al. (1969), BCKGND and SPCTRA, have been modified by the addition of six carbon dioxide

bands between 5 and 25 μm , and improved band modeling for the water vapor 6.3 μm band and the ozone 9.1 and 9.6 μm bands. The basic structure is unaltered and the division into two programs is retained. The organization of the CDC 6600 program control system makes this feasible and the overall saving in central memory storage space is greater than would be possible by using overlays. The output from the three lower altitude programs of Corbin, et al (1969), which are based on Anding's (1968) program, is required only to produce values of vertically directed radiances at the band centers. The programs INTERP and CATM were run to produce on cards a complete set of input data to RAD for each of the 47 lower atmosphere models. RAD was modified slightly to permit reading in the band data cards and to either write on tape or punch the band data cards with appropriate values of upwelling flux, TFLUX. The mid-latitude winter atmosphere model was used for the present computations.

Program BCKGND consists of the main program BCKGND and subroutines BANRAD, INTERP, SIMP, XFER, TAUMAX, SJMAX, SDM2, SM1, SLM1, QJPART, and EVAL. Program SPCTRA contains the main program SPCTRA and subroutines ROTATE, SFUN, QINT, FILTER, WATSTR, ROTSTR, ROTPOS, and SIMP. Overall operation is discussed below.

The main program BCKGND 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.

Data is read in on two different channels, one being used for number densities and collisional excitation rates, the other for band data which is needed again by program SPCTRA. The storage space used for methane data in the older version is now used for atomic oxygen concentrations. After input

data is checked, control is transferred to subroutine BANRAD.

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.

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 (wavenumber) stored in WAVE and the statistical weights stored in the ten's places in LSC. The relative populations are stored temporarily in real array VIEPOP. 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 configuration 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), G factors 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. Further computation is then 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 optical 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 total 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.

Computation of vibrational populations is done separately for each species. Control is transferred to statement 220 for CO_2 , statement 230 for H_2O , statement 240 for NO , statement 250 for N_2O or statement 230 for O_3 . The DO range following each of these statements is used to compute vibrational populations. After each set of vibrational populations is completed, new values of volume band radiances are computed in the nested DO ranges ending with statement 360. Following three iterations, vibrational populations are written to an output file for use by program SPCTRA and control is returned to the main program.

BANRAD returns volume emission radiance values in array RAD and local optical thickness values in array TAU. For each band, line of sight band radiances are computed in the DO range ending with statement 840. 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 endo-atmospheric viewing level. 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 SM1 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 both printed and written to the output file used by BANRAD, for input to SPCTRA.

Program SPCTRA computes the spectral radiances between 5 and $25\mu\text{m}$ corresponding to the band radiances computed by BCKGND. The input data to SPCTRA are the molecular vibrational level and band data also used by BCKGND, the vibrational population and band radiance data generated by BCKGND, and individual line data for the water vapor rotational lines and ν_3 band and the ozone ν_3 and ν_1 bands. The later data was obtained from R. A. McClatchey through the contract monitor.

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 individual 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 wavelengths at 0.1 μm intervals between 5 and 25 μm in real array AVLAM. These are the wavelengths at which spectral radiances are printed out. Again, more flexibility may be achieved by making the intervals and spacing dependent on values read in. If this is done, corresponding changes must be made in subroutine FILTER. The inner DO range ending with statement 105 sets the values of output spectral radiances equal to zero. Temperatures are set in the DO range ending with statement 110.

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 endoatmospheric 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 115 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, line positions are computed in subroutine ROTPOS. The DO range ending with statement 250 is entered and executed for each of the twelve altitudes for which spectra are computed. Line radiances are computed in subroutine ROTSTR for vertical endoatmospheric 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 endoatmospheric 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 molecule is non-linear (water vapor or ozone) control is transferred to statement 260 and the DO range ending with statement 270 is executed for each of the altitudes for which spectra are computed. For water vapor, spectra are generated from the pure rotational line radiances by calls to subroutine WATSTR and FILTER and for the $6.3\text{ }\mu\text{m}$ band by calls to WATSTR and FILTER, for each of the three viewing conditions. For ozone, spectra in the 9.1 and $9.6\text{ }\mu\text{m}$ bands are also generated by calls to WATSTR and FILTER.

Following the final execution of the DO ranges ending with statement 400, the spectra are printed out in the order endoatmospheric viewing in the upward direction, endoatmospheric viewing in the horizontal direction and limb viewing. Each of these sets is printed on four pages, successive pages covering $5\text{--}10\text{ }\mu\text{m}$, $10\text{--}15\text{ }\mu\text{m}$, $15\text{--}20\text{ }\mu\text{m}$ and $20\text{--}25\text{ }\mu\text{m}$. The first column on each page contains wavelengths and successive columns contain radiance values at altitudes or tangent heights 60, 65, 70, ... km. The data in arrays AVLAM, SPECA, SPECB and SPECC are written to a binary file, either for immediate use by the plotting program or for transfer to magnetic tape for later use.

C. Plotting Program

The plotting program, TRYCRT, is designed to read the binary output tape produced by the spectral radiance program, SPCTRA. It consists of the main program, TRYCRT, and six subroutines, LINSICA, LOGSCA, LINAXS, LOGAXS, GRID and NULINE. In addition, it uses the AFCRL/CDC 6600 system subroutines which are needed to produce microfilm plots. The version listed in the Appendix is set up to plot three sets of radiances, assuming that the SPCTRA output tape contains two sets of data, with vertical and horizontal endoatmospheric radiances and limb radiance data in each set. This program further assumes that there are 12 different altitude profiles in each data set. Graphic output is written on tape 39 and microfilm is produced from this. The microfilm must then be reproduced on paper.

The program places the date of the run in the first word of array IDENT, which is plotted with each graph. The other 2 words of the array contain the user name and project number, stored by a data statement. The x-axis length, SX, is set to 9 inches and the y-axis length, SY, is set to 14 inches. The CDC 6600 subroutines are initialized by calls to subroutines CRTPLT 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 LINSICA. 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. Line intensity is set to 16 by a call to subroutine STBEAM. 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. Line intensities for each curve produced in this loop are set by calls to STBEAM and use values stored in array JBEAM through a data statement. After each set of curves on a graph is produced, a call to subroutine FRAME 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 LNSCA 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.

SECTION V

RESULTS AND DISCUSSION

Representative limb viewing radiance profiles are shown in Figures V-1 through V-5. These profiles result from the use of the nitric oxide, water vapor, ozone and carbon dioxide concentrations of Figures II-1 to II-4 and an assumed nitrous oxide volume mixing ratio of 5×10^{-7} in the upper stratosphere (Keneshea 1967). The model atmosphere for the computations, used primarily for the temperature and total density profiles, is taken from the U. S. Standard Atmosphere Supplements, 1966 and uses the 30° latitude January model below 120 km and the Spring/Fall model with exospheric temperature 1000 K above 120 km.

The results shown are not qualitatively different from those of Corbin, et al. (1969). The quantitative differences result primarily from the use of different molecular abundances. The use of a lower exospheric temperature has negligible effect because at the higher altitudes where the temperature is lower, the radiation is determined by absorption of radiation from the lower atmosphere or the sun and temperature dependent collisional processes play a negligible role. The nitric oxide fundamental band at $5.3 \mu\text{m}$ is an exception, because of the excitation by collisions with atomic oxygen. Even for this molecule, results are not appreciably different during the day. At night, however, limb radiances at $5.3 \mu\text{m}$ do not fall off as rapidly with altitude below 200 km. The carbon dioxide bands added do not make an appreciable difference in the limb radiances because the basic excitation mechanism for these bands is absorption of solar or terrestrial radiation in the weak combination and overtone bands at 5.18 and $4.82 \mu\text{m}$.

Limb Viewing Spectral Radiances, 5 to 8.5 μm
Solid Line Day, Dashed Line Night

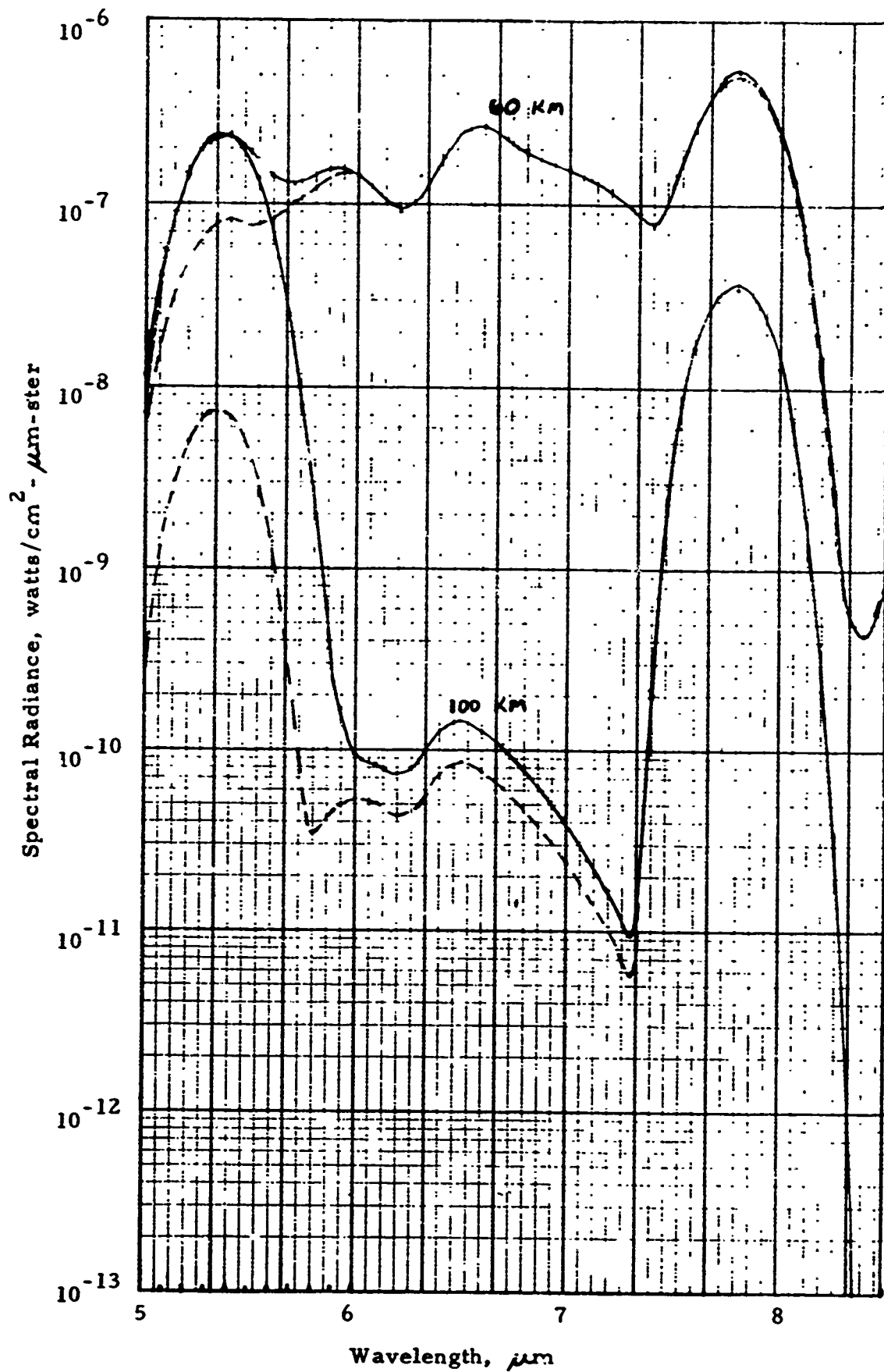


FIGURE V-1

Limb Viewing Spectral Radiances, 8 to 11.5 μm

Solid Line Day, Dashed Line Night

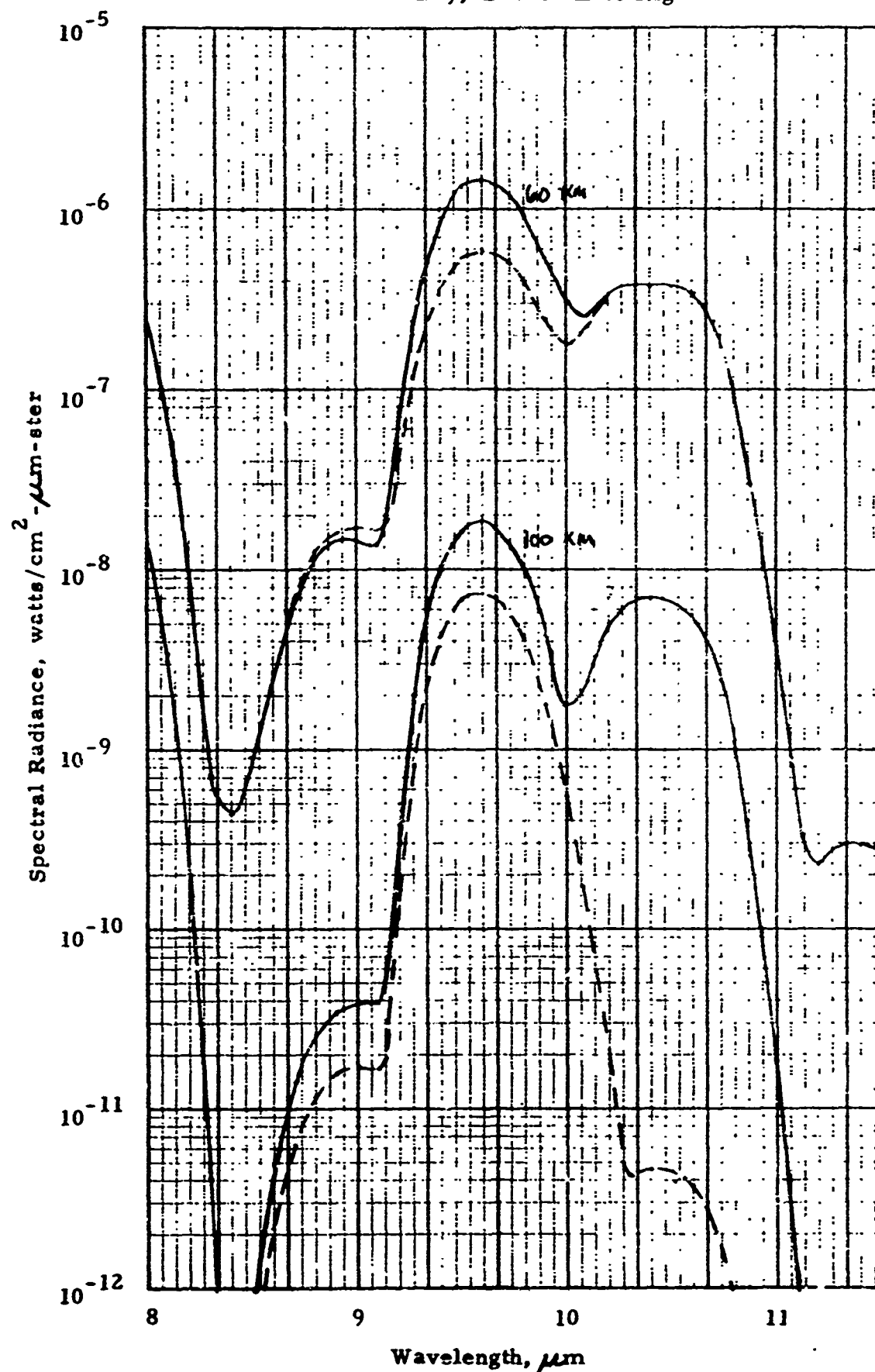


FIGURE V-2

Limb Viewing Spectral Radiances, 11 to 14.5 μm

Solid Line Day, Dashed Line Night

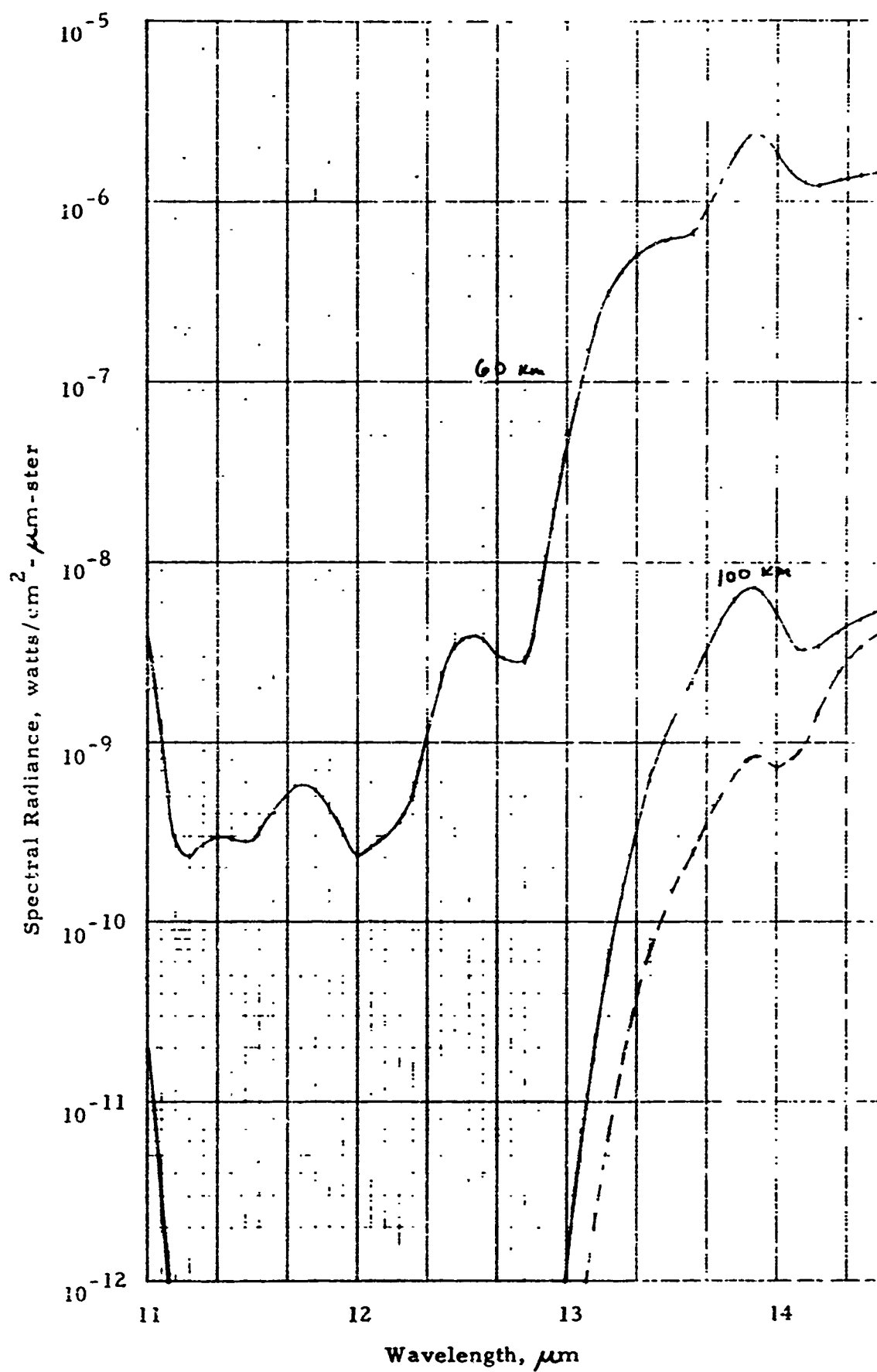


FIGURE V-3

Limb Viewing Spectral Radiances, 14 to 17.5 μm
Solid Line Day, Dashed Line Night

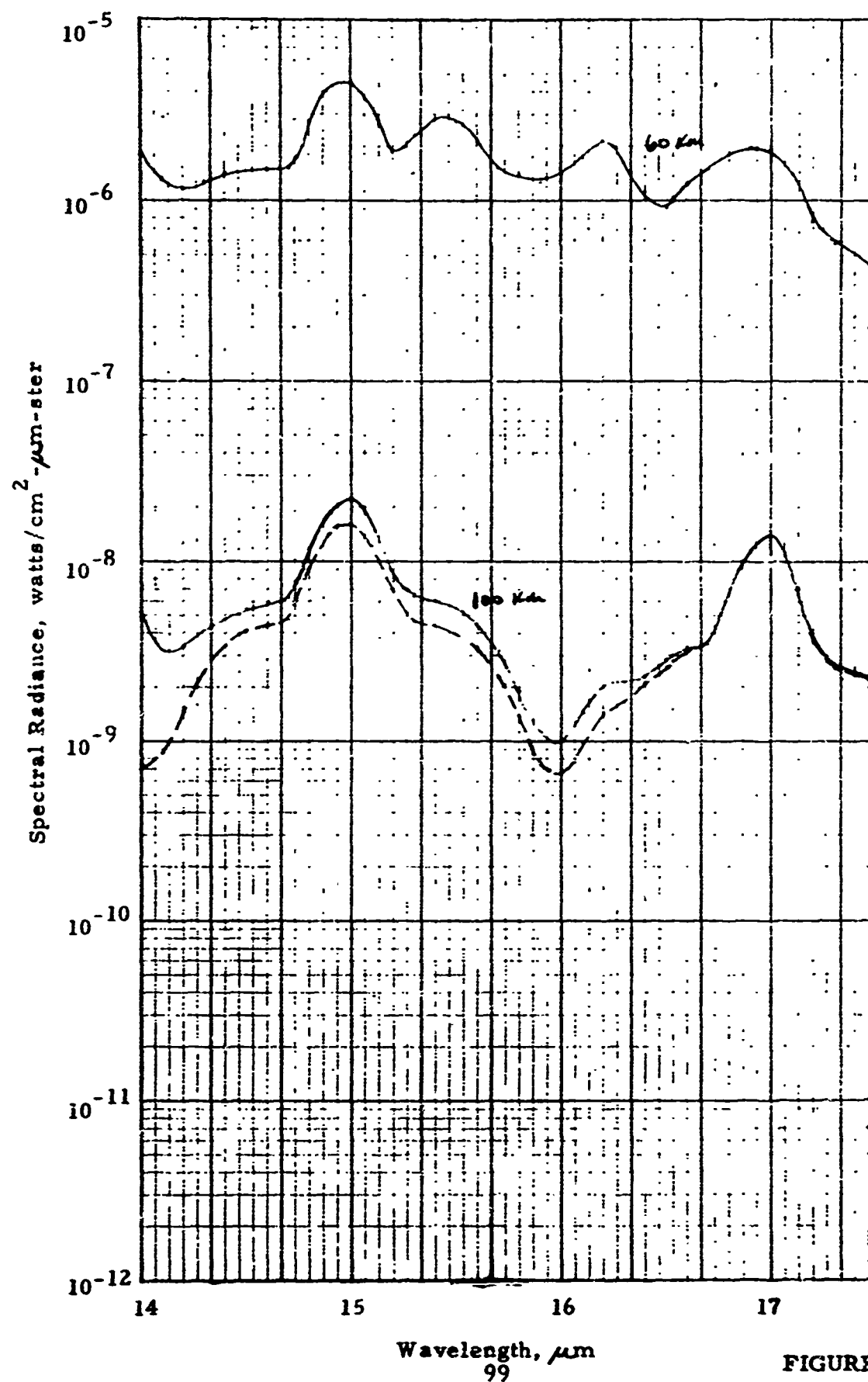
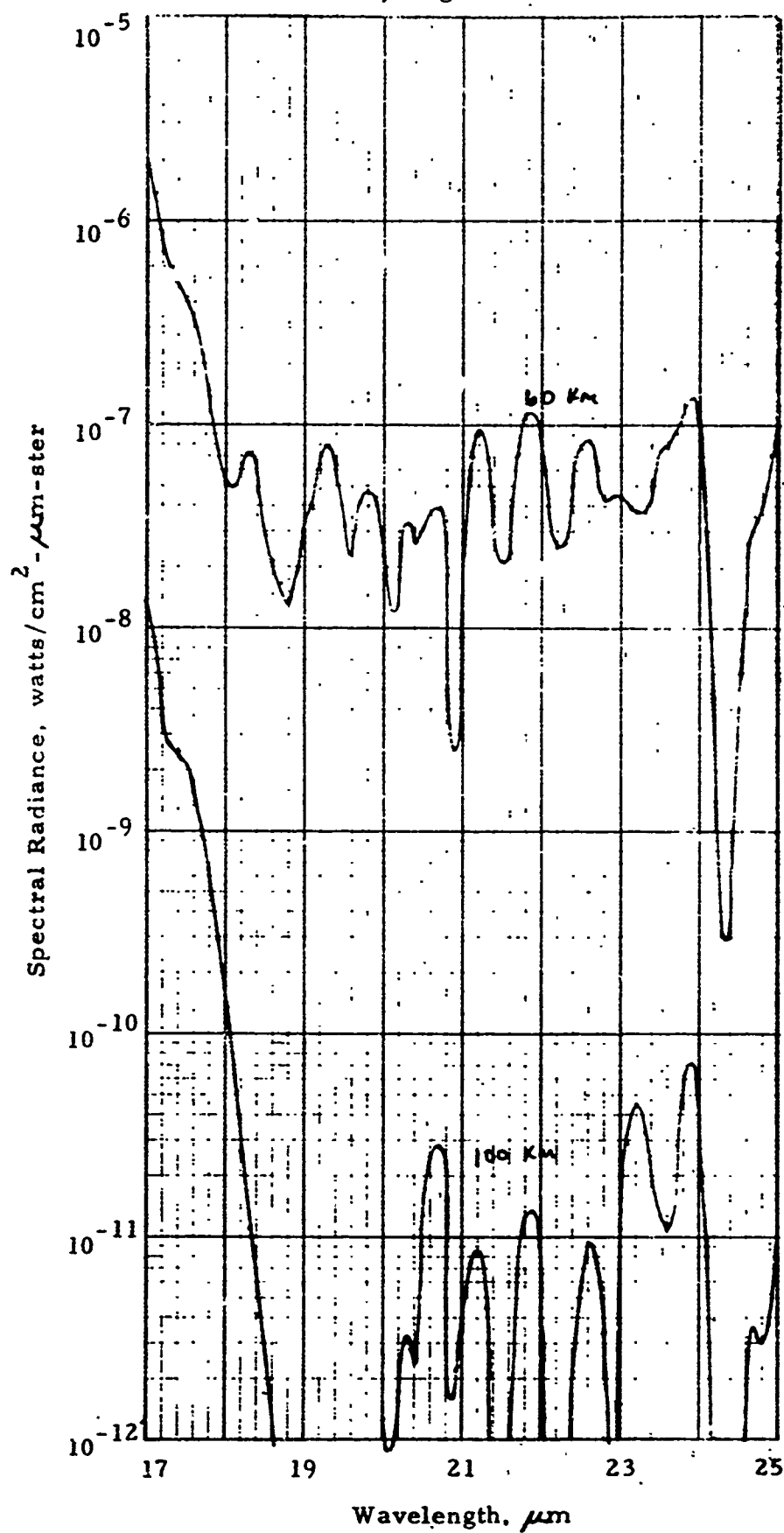


FIGURE V-4

Limb Viewing Spectral Radiances, 17 to 25 μm
No Day-Night Variation



The three general areas of investigation in this research are (a) the chemistry of the upper atmosphere and its relation to production of infrared radiation, (b) the physical mechanisms which excite and quench molecular quantum levels which can radiate in the infrared, and (c) improvement of models of the transport of infrared radiation.

The principal results of the current program have been the development of a program to compute the abundances of infrared emitting molecular species insofar as knowledge of chemical reaction rates permits, and an extension of the limb viewing infrared radiance program of Corbin, et al. (1969). This work is incomplete, but permits the definition of areas in which future work will be profitable.

In principle, especially at the higher altitudes considered here, the chemistry and diffusive mechanisms if properly described should permit the predicted concentrations to approach a diurnal steady-state. That is, the various concentrations should be reattained in every twenty-four hour period. Likewise there should be other steady-state cycle achieved, namely, those varying with the seasons and the 11-year solar cycle. These latter effects require, of course, much more elaborate modeling than is provided in the present context and must consider variation in the total amount of solar energy reaching the various portions of the earth's atmosphere and the large scale wind patterns generated by these variable inputs.

It was not possible in the present program to exercise it sufficiently to allow the diurnal steady state behavior to be investigated over the desired range of altitudes, species, and uncertain reaction rates. One run through a 24-hour cycle was performed over a restricted altitude regime to determine how much the solution varied during a single day from the specified initial conditions. A separate run was performed that covered a total of 12-days real time. These results were not acceptable due to the longer time steps that were required in the program. The exercise of the code must be continued to determine the permissible time steps that will allow steady-state to be

reached within a reasonable machine running time. Investigations must also be made of the effects on this process of the uncertainties in the experimental reaction-rates.

At altitudes above 100 km, the principal source of error in computed limb radiance profiles is the knowledge of molecular abundances. The individual lines in the molecular bands are weak enough that the neglect of collision broadening effects which lead to a Lorentz or similar shape in the wings of lines are negligible. At lower altitudes, the effects of the line wings will increase actual radiances, and further work is required to include this effect in band modeling. The effect of neglecting the far wings of the lines is expected to be greatest in the carbon dioxide bands, and here the problem is complicated by the non-Lorentz wing shape. Further work should be done on band modeling.

Measurements of radiation in spectral regions outside the nominal limits five to twenty-five microns are available and can aid in defining the processes important in producing radiation in that spectral region. The capabilities of the limb viewing atmospheric radiance model should be expanded to include the carbon dioxide and water vapor bands at 2.7 micrometers and other known bands for which sufficient data exists in the spectral range between 2.7 and 25 micrometers. Estimates of global variations of infrared radiation are required and will require at least a qualitative estimate of the effects of horizontal transport (winds, global circulation patterns) on chemistry.

Further work suggested may be summarized in the context of the present problem areas:

Chemistry. The main effort here has been to develop an efficient and reliable program to integrate the photochemical reaction rate equations, including vertical transport by eddy mixing and molecular diffusion, for atmospheric atoms and molecules which radiate in the infrared. At present

eighteen different neutral atoms and molecules are included. These are molecular nitrogen, N_2 , which is treated as a background gas with concentrations determined only by transport; two states of atomic nitrogen, $N(^4S)$ and $N(^2D)$; two states of atomic oxygen, $O(^3P)$ and $O(^1D)$; molecular oxygen, O_2 ; ozone, O_3 ; the hydroxyl radical, OH ; the hydroperoxyl radical, HO_2 ; water vapor, H_2O ; hydrogen peroxide, H_2O_2 ; atomic and molecular hydrogen, H and H_2 ; nitric oxide, NO ; nitrogen dioxide, NO_2 ; methane, CH_4 ; carbon dioxide, CO_2 ; and carbon monoxide, CO . In addition, argon is included to serve as a further check on transport calculations. Running time is slower than desired, and further development should include a search for a more efficient integration scheme, as well as the necessary addition of ionized chemical species. Qualitative estimates of the effects of horizontal transport on atmospheric chemistry should be made.

Excitation Mechanisms. Data on excitation of vibrational levels of infrared emitting molecules by collision with neutral molecules, ionized molecules and electrons need to be continually reviewed. Excitation of vibrational and electronic states of other molecules, particularly nitrogen and metastable states of atomic and molecular oxygen, which can transfer energy to infrared emitting molecules should be included. As relevant new rates are found, they should be added to the chemistry and radiance computer programs.

Radiative Transport Modeling. The addition of new bands requires further modeling of individual bands. The current method should be improved and compared with other methods.

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 Univ. 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., Kummier, 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).
- Bércecs, T. and Förgeteg, S., Trans. Far. Soc. 66, 633-639 (1970a).
- Bércecs, T. and Förgeteg, S., Trans. Far. Soc. 66, 640-647 (1970b).
- Bércecs, 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 N₂ 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, 4539-4547 (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., *J. Chem. Phys.* 54, 1718-1725 (1971).
- Degges, T. C., *Applied Optics* 10, 1856-1860 (1971).
- 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., J. 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., J. 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).
- 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 Ellingson, 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).

- 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).
- Hushfar, F., Rogers, J. W. and Stair, A. T., Jr., *Appl. Optics* 10, 1843-1847 (1971).
- Ivanov, V. V. and Shcherbakov, V. T., *Astrophys. J.* 1, 10-21 (1965).
- Jaffe, S., and Klein, F. S., *Trans. Far. Soc.* 62, 3135-3141 (1966).
- Johanson, 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).
- 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).
- 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, Ca, 23-24 June 1969.
- Malkmus, W., 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., J. Chem. Phys. 39, 3209-3213 (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. NBS 68A, 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 von 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, 853-861 (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, Ma. (1959).
- 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).
- Schurin, 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).
- Stephenson, J. C., Wood, R. E. and Moore, C. B., J. Chem. Phys. 54,
3097-3102 (1971).
- Strobel, D. F., J. Geophys. Res. 76, 2441-2452 (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).
- 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., " $\text{NI}(\text{S}^2\text{D})$ 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).
- 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

Preceding page blank

1	(TCX(4),T03),	(TCX(5),T04),	(TCX(6),TH),	(TCX(7),TH2C),	SNAP 370
2	(TCX(8),THC2),	(TCX(9),TH202),	(TCX(10),TH2),	(TCX(11),TCH4),	SNAP 380
3	(TCX(12),T002),	(TCX(13),T00),	(TCX(14),T01C),	(TCX(15),TN),	SNAP 390
4	(TCX(16),TNO),	(TCX(17),TNC2),	(TCX(18),TN2C),	(TCX(19),TN2C),	SNAP 400
					SNAP 410
					SNAP 420
					SNAP 430
					SNAP 440
					SNAP 450
					SNAP 460
					SNAP 470
					SNAP 480
					SNAP 490
					SNAP 500
					SNAP 510
					SNAP 520
					SNAP 530
					SNAP 540
					SNAP 550
					SNAP 560
					SNAP 570
					SNAP 580
					SNAP 590
					SNAP 600
					SNAP 610
					SNAP 620
					SNAP 630
					SNAP 640
					SNAP 650
					SNAP 660
					SNAP 670
					SNAP 680
					SNAP 690
					SNAP 700
					SNAP 710
					SNAP 72

DATA TST /500*0.0/	
CATA TINO /50*10.0,20.0,40.0,80.0,1.6E2,3.2E2,6.5E2,30*1.0E3,	
1 1.5E3,2.0E3,2.5E3,3.0E3,3.5E3,4.0E3,4.5E3,5.0E3,5.5E3,6.0E3,	
2 1.1E4,1.6E4,2.1E4,2.6E4,3.2E4,3.8E4,4.1E4,4.3E4,4.7E4,	
3 395*5.0E4/	
DATA TETOT /50*10.0,20.0,40.0,80.0,1.6E2,3.2E2,6.5E2,26*1.0E3,	
1 1.1E3,1.2E3,1.4E3,1.5E3,2.4E3,3.3E3,4.2E3,5.1E3,6.0E3,6.8E3,	
2 7.6E3,8.4E3,9.2E3,1.0E4,1.9E4,2.8E4,3.7E4,4.6E4,5.5E4,6.4E4,	
3 7.3E4,8.2E4,9.1E4,396*1.0E5/	
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,	
1 10.0,15.0,20.0,25.0,30.0,35.0,40.0,50.0,60.0,80.0,100.0,	
2 120.0,140.0,160.0,180.0,220.0,300.0,400.0,500.0,600.0,	
3 700.0,750.0,800.0,850.0,900.0,1.0E3,1.2E3,1.4E3,1.6E3,	
4 1.8E3,2.0E3,2.2E3,2.4E3,2.7E3,3.0E3,3.5E3,4.0E3,4.5E3,	
5 5.0E3,5.5E3,6.0E3,7.0E3,8.0E3,9.0E3,1.0E4,1.1E4,1.2E4,	
6 1.3E4,1.4E4,1.5E4,1.8E4,2.1E4,2.4E4,2.7E4,3.0E4,3.3E4,	
7 3.6E4,3.9E4,4.2E4,341*4.5E4/	
DATA TIO2 /81*0.0,100.0,200.0,300.0,400.0,600.0,900.0,	
1 1.2E3,1.6E3,2.0E3,2.5E3,2.8E3,3.1E3,3.4E3,3.7E3,4.0E3,	
2 5.0E3,7.0E3,1.0E4,1.5E4,2.0E4,2.3E4,2.6E4,2.9E4,3.2E4,	
3 3.5E4,3.8E4,4.1E4,4.4E4,4.7E4,391*5.0E4/	

DATA LALT,LTIME,LCHI,LCOL02,LCOL03,LJ02,LJ03,LQCOMA,	
1 LQCOMB,LQZCHA,LQZCHB,LML /6H ALT,6H TIME,6H CHI,6H COL02,	
2 6H COLN2,6H COL03,6H J02,6H J03,6HQA(OH),6HGB(CH),6HIA(OH),	
3 6HIB(OH),6H ML/	

DATA AMNT / 28.0134, 31.9988, 15.9994, 47.9982, 17.0074,	
1 1.00797, 18.0153, 33.0067, 34.0147, 2.01594, 16.0430, 44.0099,	
2 28.0105, 15.9994, 14.0067, 30.0061, 46.0055, 44.0128, 14.0067,	

```

C      3 39.9480, 30*0.0 /
      DATA EDCOF / 51*1.0E5,1.09E5,1.202E5,1.318E5,1.445E5,1.585E5,
1 1.738E5,1.905E5,2.095E5,2.291E5,2.512E5,2.754E5,3.02E5,3.31E5,
2 3.63E5,3.98E5,4.36E5,4.79E5,5.25E5,5.75E5,6.31E5,6.92E5,7.59E5,
3 8.32E5,9.12E5,1.0E6,1.09E6,1.202E6,1.318E6,1.445E6,1.585E6,
4 1.738E6,1.905E6,2.095E6,2.291E6,2.512E6,2.754E6,3.02E6,3.31E6,
5 3.63E6,3.98E6,4.36E6,4.78E6,5.25E6,5.75E6,6.31E6,6.92E6,7.59E6,
6 8.32E6,9.12E6,1.0E7,1.09E7,1.20E7,1.24E7,1.27E7,1.24E7,
7 1.18E7,1.05E7,8.20E6,3.162E6,1.0E6,3.162E5,1.0E5,3.162E4,
8 1.0JE4,3.162E3,1.0JE2,3.162E2,1.0JE1,3.162E1,1.0E1,380*1.0/
      DATA EDCF / 12*1.0E5,7.943E4,6.310E4,5.012E4,3.981E4,3.162E4,
1 2.512E4,1.995E4,1.585E4,1.259E4,1.1*1.0E4,1.259E4,1.585E4,
2 1.995E4,2.512E4,3.162E4,3.981E4,5.012E4,6.310E4,7.943E4/

C      DATA LSP / 6H N2 , 6H O2 , 6H O(3P) , 6H O3 , 6H CH , 6H H ,
1 6H H2O , 6H H2 , 6H H2O2 , 6H H2 , 6H CH4 , 6H CC2 , 6H CO ,
2 6H O(1D) , 6H N(4S) , 6H NO , 6H NO2 , 6H N2O , 6H N(2C) ,
3 6*6H /

C      DATA TFMT / 10H (/ 5X, 5HSTA, 10HRT, 6X, 3HEN, 10HD, 7X, 5FTCT,
1 10HAL, 6X, 5HTI, 10HMES / 1X, 3F1, 5H0.3 / ) /

C      STATEMENT FUNCTION TC CONVERT DEGREES TO RADIANS
      RAD(DEG) = 1.7453292519943E-2*DEG

C      STATEMENT FUNCTION TC COMPLETE 1.0 - EXP(-X)
      EXPM(X) = (((0.008333*X - 0.04166667)*X + 0.1666666667)*X
1 - 0.5)*X + 1.0)*X

C      EDDY DIFFUSION COEFFICIENTS ARE STORED IN ARRAY EDCCF
C      KINETIC TEMPERATURES ARE STORED IN ARRAY TEMPK
C      NUMBER DENSITIES FOR ALL ALTITUDES AND SPECIES ARE
C      STORED IN ARRAY CONSP

```

```

SNAP 730
SNAP 740
SNAP 750
SNAP 760
SNAP 770
SNAP 780
SNAP 790
SNAP 800
SNAP 810
SNAP 820
SNAP 830
SNAP 840
SNAP 850
SNAP 860
SNAP 870
SNAP 880
SNAP 890
SNAP 900
SNAP 910
SNAP 920
SNAP 930
SNAP 940
SNAP 950
SNAP 960
SNAP 970
SNAP 980
SNAP 990
SNAP1000
SNAP1010
SNAP1020
SNAP1030
SNAP1040
SNAP1050
SNAP1060
SNAP1070
SNAP1080

```

```

C C      NUMBER DENSITIES FOR ALTITUDE NALT ARE STORED IN ARRAY CCX
C C      NUMBER DENSITIES FOR NALT - 1 AT TIME T ARE STORED IN ARRAY SCX
C C      ARRAYS RCX AND TCX STORE APPROXIMATIONS DURING INTEGRATION
C C      DIFFUSION CONTRIBUTIONS TO THE Q COEFFICIENTS ARE
C C      STORED IN ARRAY GJED
C C      DIFFUSION CONTRIBUTIONS TO THE BETA COEFFICIENTS ARE
C C      STORED IN ARRAY ETED
C C      TERMS USED TO COMPUTE QJED ARE STORED IN ARRAYS GJECL AND QJEDU
C C
C C      INITIALIZATION OF CONSTANTS
C C      NW = 6
C C      NPAGE = -1
C C      PI = 3.1415926535898
C C      RE = 6371.0
C C      DELZ = 1.0E5
C C      RDELZ = 0.5/DELZ
C C      R2DELZ = DELZ**(-2)
C C      NPRNT = 1
C C      DO 90 NALT = 1, 150
C C          EDCOEFF(NALT) = EDCOEFF(NALT + 9)
C C          CONTINUE
C C      DO 95 I = 1, 41
C C          EDCOEFF(I) = EDCF(I)
C C          CONTINUE
C C
C C      FOR CONVENIENCE IN CHECKING FORMATION OF COEFFICIENTS IN THE
C C      DIFFERENTIAL EQUATIONS, INDEX VALUES OF THE FORM K#2, K02,
C C      ETC ARE USED IN PLACE OF SPECIFIC VALUES CF NSP.
C C      KN2 = 1
C C      K02 = 2
C C      K0 = 3
C C      K03 = 4
C C      K0H = 5
C C      KH = 6

```

SNAP1450
 SNAP1460
 SNAP1470
 SNAP1480
 SNAP1490
 SNAP1500
 SNAP1510
 SNAP1520
 SNAP1530
 SNAP1540
 SNAP1550
 SNAP1560
 SNAP1570
 SNAP1580
 SNAP1590
 SNAP1600
 SNAP1610
 SNAP1620
 SNAP1630
 SNAP1640
 SNAP1650
 SNAP1660
 SNAP1670
 SNAP1680
 SNAP1690
 SNAP1700
 SNAP1710
 SNAP1720
 SNAP1730
 SNAP1740
 SNAP1750
 SNAP1760
 SNAP1770
 SNAP1780
 SNAP1790
 SNAP1800

KH20 = 7
 KHC2 = 8
 KH202 = 9
 KH2 = 10
 KCH4 = 11
 KC02 = 12
 KCO = 13
 K01D = 14
 KN = 15
 KNO = 16
 KN02 = 17
 KN20 = 18
 KN2D = 19

READ (5,21) IDENT,IDENT
 WRITE (4,21) IDENT,IDENT
 WRITE (6,20) IDENT,NPAGE
 NPAGE = NPAGE + 1

PRINT INPUT PARAMETERS

WRITE (6,21) IDENT,IDENT
 READ (5,15) GTHETA,CARD
 WRITE (6,15) DTHETA,CARD
 WRITE (4,15) DTHETA,CARD
 READ (5,15) DPHI,CARC
 WRITE (6,15) DPHI,CARC
 WRITE (4,15) DPHI,CARC
 READ (5,15) SOLDEC,CARD
 WRITE (6,15) SOLDEC,CARD
 WRITE (4,15) SOLDEC,CARD
 READ (5,15) GREF,CARC
 WRITE (6,15) GREF,CARC
 WRITE (4,15) GREF,CARC
 READ (5,15) REFF,CARD

C
 C C C C

SNAP1810
 SNAP1820
 SNAP1830
 SNAP1840
 SNAP1850
 SNAP1860
 SNAP1870
 SNAP1880
 SNAP1890
 SNAP1900
 SNAP1910
 SNAP1920
 SNAP1930
 SNAP1940
 SNAP1950
 SNAP1960
 SNAP1970
 SNAP1980
 SNAP1990
 SNAP2000
 SNAP2010
 SNAP2020
 SNAP2030
 SNAP2040
 SNAP2050
 SNAP2060
 SNAP2070
 SNAP2080
 SNAP2090
 SNAP2100
 SNAP2110
 SNAP2120
 SNAP2130
 SNAP2140
 SNAP2150
 SNAP2160

WRITE (6,15) REFF,CARC
 WRITE (4,15) REFF,CARC
 READ (5,15) EPS,CARC
 WRITE (6,15) EPS,CARC
 WRITE (4,15) EPS,CARC
 READ (5,16) HLMAX,CARD
 WRITE (6,16) HLMAX,CARD
 WRITE (4,16) HLMAX,CARD
 READ (5,16) NBASE,CARD
 WRITE (6,16) NBASE,CARD
 WRITE (4,16) NBASE,CARD
 READ (5,16) NTOP,CARC
 WRITE (6,16) NTOP,CARC
 WRITE (4,16) NTOP,CARC
 READ (5,16) IHRMAX,CARD
 WRITE (6,16) IHRMAX,CARD
 WRITE (4,16) IHRMAX,CARD
 READ (5,16) MMAX,CARC
 WRITE (6,16) MMAX,CARC
 WRITE (4,16) MMAX,CARC
 READ (5,15) OLTIME,CARD
 WRITE (6,15) OLTIME,CARD
 WRITE (4,15) OLTIME,CARD
 READ (5,15) TIME,CARC
 WRITE (6,15) TIME,CARC

WRITE (6,20) IDENT,NPAGE
 NPAGE = NPAGE + 1
 DO 100 NALT = 1, 201
 READ (5,34) NXX,N1,(CONSP(NALT,J), J = 1, 7)
 WRITE (6,35) NXX,N1,(CONSP(NALT,J), J = 1, 7)
 IF (NXX.NE.NALT) GO TO 500
 IF (N1.NE.1) GO TO 500
 READ (5,34) NXX,N2,(CONSP(NALT,J), J = 8, 14)

C C C


```

RMWT = 0.0
SUMU = 0.0
SUM = 0.0
I = 0
DO 110 NALT = NEASE, NTOP
  I = I + 1
  NA = NALT - 1
  NB = NALT - 2
  TEMPL = TEMP
  TEMP = TEMPU
  SHL = SH
  SH = SHU
  GL = G
  G = GU
  RMWT = RMWT
  RMWTU = RMWTU
  SUML = SUM
  SUM = SUMU
  SUMU = 0.0
  RMWTU = 0.0
DO 102 NSP = 1, 20
  SUMU = SUMU + CONSP(NALT,NSP)
  RMWTU = RMWTU + AMWT(NSP)*CCNSP(NALT,NSP)
102 CONTINUE
  RMWTU = RMWTU/SUMU
  RCBS = NA
  TEMPU = TKEL(ROBS)
  ROTEMP(NALT) = 2.68684E19*(TEMPU/273.15)**0.75
  TEMPK(NALT) = TEMPU
  GU = GREF*(REFF/(REFF + ROBS))**2
  SHU = 8.31432E7*TEMPU/(RMWTU*GU)
  IF (I.EQ. 2) SHFACL = EXP(1.0E5/SHU)
  IF (I.LE. 2) GC TO 110
  IF (NALT.EQ. NTCF - 1) SHFAC = EXP(-1.0E5/SHU)
  GTOZ = (TEMPU - TEMPL)*RDELZ
  DKDZ = (EDCCEF(NALT) - EDCCEF(NB))*RCELZ

```

SNAP2530
 SNAP2540
 SNAP2550
 SNAP2560
 SNAP2570
 SNAP2580
 SNAP2590
 SNAP2600
 SNAP2610
 SNAP2620
 SNAP2630
 SNAP2640
 SNAP2650
 SNAP2660
 SNAP2670
 SNAP2680
 SNAP2690
 SNAP2700
 SNAP2710
 SNAP2720
 SNAP2730
 SNAP2740
 SNAP2750
 SNAP2760
 SNAP2770
 SNAP2780
 SNAP2790
 SNAP2800
 SNAP2810
 SNAP2820
 SNAP2830
 SNAP2840
 SNAP2850
 SNAP2860
 SNAP2870
 SNAP2880


```

DHCZ = (SHU - SHL)*R3ELZ
TFAC = QTDZ/TEMP
AFAC(NA) = TFAC
SFAC = (4.0 - (TEMPU + TEMPL)/TEMP)*R2DELZ + TFAC*TFAC
BFAC(NA) = SFAC
FAC = 1.0/SH + TFAC
ED = EDOEF(NA)
BTEC(NA) = (SFAC + DHCZ/(SH*SH))*ED - FAC*DKDZ
CFAC(NA) = (TEMPL/GU - TEMPL/GL)*G*RDELZ/(8.31432E7*TEMP)
CFAC(NA) = G/(8.31432E7*TEMP)
FAC = (FAC*ED + CKCZ)*RDELZ
QJEDU(NA) = R2CELZ*ED + FAC
QJEDL(NA) = R2CELZ*ED - FAC
IF (IHR.GT. 1) GC TC 110
CONFAC = SUML*(QJEDL(NA)/QJEDU(NA) + 1.0)/((BTEC(NA)/QJEDU(NA)
1 + (TEMPU - TEMPL)/TEMP + 2.0*DELZ/SH)*SUM)
SUM = CONFAC*SUM
IF (NALT.EC. NTCF) CON = (BTEU(NA)*SUM - QJEDL(NA)*SUML)
1 / (QJEDU(NA)*SUMU)
DO 104 NSP = 1, NSPM
CCNSP(NA,NSF) = CCNSP(NA,NSF)*CONFAC
IF (NALT.EC. NTCF) CCNSP(NTCF,NSP) = CON*CONFAC(NTCF,NSF)
104 CONTINUE
WRITE (6,6) NB,CCNSP(NA,1),CCNSP(NA,2),GL,SH,CTCZ,CKCZ,DHDZ,
1 STEU(NA),QJEDL(NA),QJEDU(NA)
WRITE (6,6) N1,PMWT,TFAC,SFAC,CFAC(NA),DFAC(NA),RTEMP(NA)
WRITE (6,6) NALT,CCNC(1),CCNC(2),CCNC(4),CCNC(7),CCNC(9),CCNC(10)
CONTINUE
NBASE = NBSAV
NBASE = NBASE + 1
NBOP = NTOP - 1
NMIN = NBASE - 1
CC 400 M = 1, MPAX
NTIME = (0.5*CLTIME + TIME)/60.0 + 720.1
RTIME = FLOAT(NTIME)/60.0
NTIME = 100*(NTIME/60) + MOD(NTIME,60)

```

SNAP28
 SNAP29
 SNAP291
 SNAP292
 SNAP293
 SNAP294
 SNAP295
 SNAP296
 SNAP297
 SNAP298
 SNAP299
 SNAP300
 SNAP301
 SNAP302
 SNAP303
 SNAP304
 SNAP305
 SNAP306
 SNAP307
 SNAP308
 SNAP309
 SNAP310
 SNAP311
 SNAP312
 SNAP313
 SNAP314
 SNAP315
 SNAP316
 SNAP317
 SNAP318
 SNAP319
 SNAP320
 SNAP321
 SNAP322
 SNAP323
 SNAP324

```

C
C
120
IF (MCD(M,NPRNT) .NE. 0) GO TO 120
WRITE (6,20) IDENT,NPAGE
NPAGE = NPAGE + 1
WRITE (6,6) NTIME,RTIME
WRITE (6,19)
WRITE (6,22) LALT,LTIME,LCHI,LCOLO2,LCOLN2,LCCLC3,LJC2,LJ03
WRITE (6,22) LALT,LQCHA,LCCCH8,LOZCHA,LOZOH8,(LSP(I),I=1,6)
WRITE (6,22) LML,(LSP(I),I=7,14)
WRITE (6,19)
CONTINUE
RFAC = ROTEMP(NMIN)
C1 = CONSP(NMIN,1)
C2 = CONSP(NMIN,2)
C3 = CONSP(NMIN,3)
CMDL(1) = RFAC/(C1/U.180 + C2/U.181 + C3/U.260)
CMDL(2) = RFAC/(C1/U.181 + C2/U.189 + C3/U.260)
CMDL(3) = RFAC/(C1/U.260 + C2/U.260 + C3/U.370)
RFAC = ROTEMP(NBASE)
C1 = CONSP(NBASE,1)
C2 = CONSP(NBASE,2)
C3 = CONSP(NBASE,3)
CMD(1) = RFAC/(C1/U.180 + C2/U.181 + C3/U.260)
CMD(2) = RFAC/(C1/U.181 + C2/U.189 + C3/U.260)
CMD(3) = RFAC/(C1/U.260 + C2/U.260 + C3/U.370)
DO 130 NSP = 1,NSPM
SCX(NSP) = CONSP(NBASE-1,NSP)
IF (NSP .LT. 4) GO TO 130
CMDL(NSP) = U.707*CMDL(2)*SQRT(31.9988/AMWT(NSP) + 1.0)
CMD(NSP) = U.707*CMD(2)*SQRT(31.9988/AMWT(NSP) + 1.0)
CONTINUE
130
DZOH8 = 0.0
DZOH8 = 0.0
DO 350 NALT = NBASE, NTOP
NPLU = NALT + 1
NMIN = NALT - 1
RFAC = ROTEMP(NPLU)

```

```

SNAP3250
SNAP3260
SNAP3270
SNAP3280
SNAP3290
SNAP3300
SNAP3310
SNAP3320
SNAP3330
SNAP3340
SNAP3350
SNAP3360
SNAP3370
SNAP3380
SNAP3390
SNAP3400
SNAP3410
SNAP3420
SNAP3430
SNAP3440
SNAP3450
SNAP3460
SNAP3470
SNAP3480
SNAP3490
SNAP3500
SNAP3510
SNAP3520
SNAP3530
SNAP3540
SNAP3550
SNAP3560
SNAP3570
SNAP3580
SNAP3590
SNAP3600

```

SNAP3610
SNAP3620
SNAP3630
SNAP3640
SNAP3650
SNAP3660
SNAP3670
SNAP3680
SNAP3690
SNAP3700
SNAP3710
SNAP3720
SNAP3730
SNAP3740
SNAP3750
SNAP3760
SNAP3770
SNAP3780
SNAP3790
SNAP3800
SNAP3810
SNAP3820
SNAP3830
SNAP3840
SNAP3850
SNAP3860
SNAP3870
SNAP3880
SNAP3890
SNAP3900
SNAP3910
SNAP3920
SNAP3930
SNAP3940
SNAP3950
SNAP3960

```

C1 = CONSP(NPLU,1)
C2 = CONSP(NFLU,2)
C3 = CONSP(NPLU,3)
CMDU(1) = RFAC/(C1/U.181 + C2/U.181 + C3/U.260)
CMDU(2) = RFAC/(C1/U.181 + C2/U.181 + C3/U.260)
CMDU(3) = RFAC/(C1/U.260 + C2/U.260 + C3/U.370)
AFACNA = AFAC(NALT)
BFACNA = BFAC(NALT)
CFACNA = CFAC(NALT)
DFACNA = DFAC(NALT)
QJEDUN = QJEDU(NALT)
QJEDLN = QJEDL(NALT)
BTEDNA = BTED(NALT)
DO 140 NSP = 1,NSPM
CCNSPN = CONSP(NALT,NSP)
DCX(NSP) = CONSPN
TCX(NSP) = CONSPN
CMDNSP = CMD(NSP)
IF (NSP.LT. 4) GO TO 135
CMDU(NSP) = U.707*CMDU(2)*SCRT(31.9986/AMWT(NSP) + 1.0)
CONTINUE
FAC = AMWT(NSP)*CFACNA + AFACNA
DDZ = (CMDU(NSP) - CMDL(NSP))*RDELZ
BTEND(NSP) = (BFACNA + AMWT(NSP)*CFACNA)*CMDNSP - FAC*DDCZ
1 + BTEDNA
FAC = (FAC*CMDNSP + DCDZ)*RCELZ
QJEMD(NSP) = (R2DELZ*CMDNSP + FAC + QJEDUN)*CCNSP(NFLU,NSP)
1 + (R2DELZ*CMDNSP - FAC + QJEDLN)*CCNSP(NAMIN,NSP)
CMDL(NSP) = CMDNSP
CMD(NSP) = CMDL(NSP)
CONTINUE
TEMP = TEMPK(NALT)
RCBS = FLOAT(NALT - 1)
CALL SOLLY (THETA,ROES,TIME,SOLDEC,CHI,ETA)
IF (ETA.LE. 0.0) GO TO 160
CHISAV = 1.0
140

```

135

140

```

IF (CHI .GT. 1.0) GO TO 145
CHISAV = 1.0/COS(CHI)
CHI = 0.0
CONTINUE
CALL COLUMN (ROBS,CHI,COL,ETA)
COL(1) = CHISAV*COL(1)
COL(2) = CHISAV*COL(2)
COL(3) = CHISAV*COL(3)
COL(4) = CHISAV*COL(4)
CONTINUE
CONTINUE
CALL SUNPHI (COL,ETA,RATE)
TEMP1 = 1.0/TEMP
TEMP12 = SQRT(TEMP1)
TEMP32 = TEMP1*TEMP12
TEMP52 = TEMP1*TEMP32
FORWARD REACTIONS
REVERSE REACTIONS
REACTION 1  N + O2 = NC + C
R1 = 1.1E-14*TEMP*EXP(-3140.0*TEMP1)
S1 = 2.6E-15*TEMP*EXP(-19500.0*TEMP1)
REACTION 2  N + NO = N2 + C
R2 = 5.1E-11*EXP(-168.0*TEMP1)
S2 = 2.3E-10*EXP(-37500.0*TEMP1)
REACTION 3  NO + NO = N2 + C2
R3 = 1.3*TEMP52*EXP(-42800.0*TEMP1)
S3 = 15.2*TEMP52*EXP(-64300.0*TEMP1)
REACTION 4  O + O + M = O2 + M
R4 = 3.80E-30*TEMP1*EXP(-171.0*TEMP1)
S4 = 4.57E-5*TEMP1*EXP(-59700.0*TEMP1)
REACTION 5  N + N + M = N2 + M
R5 = 1.4E-33*EXP(-500.0*TEMP1)
S5 = 3.2E-7*TEMP12*EXP(-112000.0*TEMP1)
REACTION 6  N + N + N2 = N2 + N2
R6 = 7.3E-32*TEMP12
S6 = 7.8E-7*TEMP12*EXP(-112000.0*TEMP1)

```

```

SNAP3970
SNAP3980
SNAP3990
SNAP4000
SNAP4010
SNAP4020
SNAP4030
SNAP4040
SNAP4050
SNAP4060
SNAP4070
SNAP4080
SNAP4090
SNAP4100
SNAP4110
SNAP4120
SNAP4130
SNAP4140
SNAP4150
SNAP4160
SNAP4170
SNAP4180
SNAP4190
SNAP4200
SNAP4210
SNAP4220
SNAP4230
SNAP4240
SNAP4250
SNAP4260
SNAP4270
SNAP4280
SNAP4290
SNAP4300
SNAP4310
SNAP4320

```

C	REACTION 7	N + N + N = N2 + N	SNAP4330
	R7 =	9.25E-27*TEMP32	SNAP4340
C	S7 =	6.75E-2*TEMP32*EXP(-112000.0*TEMP)	SNAP4350
	REACTION 8	N + O + M = NC + M	SNAP4360
	R8 =	1.1E-32*SQRT(273.0*TEMP)	SNAP4370
C	S8 =	3.8E-7*TEMP12*EXP(-74500.0*TEMP)	SNAP4380
	REACTION 9	NO + O = NC2 + H-NU	SNAP4390
	R9 =	6.5E-17*TEMP/296.0	SNAP4400
	S9 =	RATE(4)	SNAP4410
C	REACTION 10	C + NO2 = NO + O2	SNAP4420
	R10 =	1.6E-11*EXP(-300.0*TEMP)	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 = NC2 + M	SNAP4510
	R13 =	2.9E-33*EXP(940.0*TEMP)	SNAP4520
	S13 =	1.8E-8*EXP(-32800.0*TEMP)	SNAP4530
C	REACTION 14	NO + NC + O2 = NO2 + NC2	SNAP4540
	R14 =	6.6E-39*EXP(530.0*TEMP)	SNAP4550
	S14 =	6.6E-12*EXP(-13500.0*TEMP)	SNAP4560
C	REACTION 15	O + C = O2 + H-NU	SNAP4570
	S15 =	RATE(1)	SNAP4580
C	REACTION 16	N + O = NO + H-NU	SNAP4590
	S16 =	RATE(3)	SNAP4600
C	REACTION 21	NC + O3 = O2 + NC2	SNAP4610
	R21 =	9.5E-13*EXP(-1240.0*TEMP)	SNAP4620
	S21 =	1.0E-13*EXP(-23500.0*TEMP)	SNAP4630
	REACTION 31	O + O2 + M = C3 + M	SNAP4640
	R31 =	4.63E-35*EXP(1060.00*TEMP)	SNAP4650
	S31 =	1.65E-9*EXP(-11920.0*TEMP)	SNAP4660
	REACTION 32	C + C3 = O2 + C2	SNAP4670
	R32 =	2.00E-11*EXP(-2410.0*TEMP)	SNAP4680

SNAP4690
 SNAP4700
 SNAP4710
 SNAP4720
 SNAP4730
 SNAP4740
 SNAP4750
 SNAP4760
 SNAP4770
 SNAP4780
 SNAP4790
 SNAP4800
 SNAP4810
 SNAP4820
 SNAP4830
 SNAP4840
 SNAP4850
 SNAP4860
 SNAP4870
 SNAP4880
 SNAP4890
 SNAP4900
 SNAP4910
 SNAP4920
 SNAP4930
 SNAP4940
 SNAP4950
 SNAP4960
 SNAP4970
 SNAP4980
 SNAP4990
 SNAP5000
 SNAP5010
 SNAP5020
 SNAP5030
 SNAP5040

S32 = 2.12E-11*EXP(-50700.0 * TEMPI)
 REACTION 33 H + O3 = OH + C2
 P33 = 2.6E-11
 S33 = 0.
 REACTION 34 O + OH = H + O2
 R34 = 2.06E-11
 S34 = 4.19E-10 * EXP(-8450.0*TEMPI)
 R35 H02 + O = OH + C2
 R35 = 1.0E-11
 R36 H02 + H = OH + CH
 R36 = 2.0E-10*EXP(-2000.0*TEMPI)
 R37 H + O2 + M = H02 + M
 R37 = 3.0E-32*(273.0*TEMPI)**1.3
 R38 OH + C3 = HC2 + O2
 R38 = 5.0E-13
 R39 H02 + O3 = OH + O2 + C2
 R39 = 1.0E-14
 R40 OH + CH = H2C + O
 R40 = 9.3E-12*EXP(-390.0*TEMPI)
 S40 = 9.3E-11*EXP(-9000.0*TEMPI)
 R41 OH + H02 = H2O + O2
 R41 = 1.0E-11
 R42 H + HC2 = H2 + C2
 R42 = 2.0E-13
 R43 O + H2 = CH + H
 R43 = 7.0E-11*EXP(-5100.0*TEMPI)
 R44 H02 + H02 = H2O2 + O2
 R44 = 1.5 E-12
 R45 OH + H2C2 = H2O + H02
 R45 = 4.0E-13
 R46 O + H2C2 = OH + H02
 R46 = 1.0E-15
 R47 H + H2C2 = H2 + H02
 R47 = 1.0E-13
 R50 O(ID) + C3 = O2 + O2
 R50 = 5.0E-11

SNAP5050
 SNAP5060
 SNAP5070
 SNAP5080
 SNAP5090
 SNAP5100
 SNAP5110
 SNAP5120
 SNAP5130
 SNAP5140
 SNAP5150
 SNAP5160
 SNAP5170
 SNAP5180
 SNAP5190
 SNAP5200
 SNAP5210
 SNAP5220
 SNAP5230
 SNAP5240
 SNAP5250
 SNAP5260
 SNAP5270
 SNAP5280
 SNAP5290
 SNAP5300
 SNAP5310
 SNAP5320
 SNAP5330
 SNAP5340
 SNAP5350
 SNAP5360
 SNAP5370
 SNAP5380
 SNAP5390
 SNAP5400

C R51 O(ID) + N2 = O + N2
 R51 = 5.0E-11
 C R52 O(IC) + H2 = OH + H
 R52 = 1.0E-11
 C R53 O(ID) + H2O = OH + OH
 R53 = 3.0E-11
 C R54 O(ID) + CH4 = PRODUCTS
 R54 = 1.0E-11
 C R55 CH4 + OH = H2O + CH3
 R55 = 1.2E-10*EXP(-2960.0*TEMP)
 C R56 O + CH4 = CH + CH3
 R56 = 3.5E-11*EXP(-4550.0*TEMP)
 C R60 CO + OH = CO2 + H
 R60 = 1.0E-12*EXP(-540.0*TEMP)
 C R61 CO + C + M = CO2 + M
 R61 = 1.4E-33*EXP(-1250.0*TEMP)
 C IONIC REACTION 1 O2+ + E = C + O
 RI1 = 6.0E-5*TEMP
 SI1 = 0.0
 C IONIC REACTION 2 N2+ + E = N + N
 RI2 = 5.0E-5*TEMP
 SI2 = 0.0
 C IONIC REACTION 3 N2+ + O2 = N2 + O2+
 RI3 = 1.0E-10
 SI3 = 0.0
 C IONIC REACTION 4 N2+ + E = N2 + H-NU
 RI4 = 1.0E-16
 SI4 = 1.0E-16
 C WRITE (6,19)
 NA = NALT - 1
 C
 C 170. CONTINUE
 C
 C CO 300 ML = 1, MLMAX

```

180      MLSAVE = ML
      MMF = (MOD(ML + 1,2) + 2)*2*((ML - 1)/2)
      IF (MLMAX.EC. 1) MMF = 1
      DO 180 NSP = 1,NSPM
      DCX(NSP) = CONSP(NALT,NSP)
      CONTINUE
180      DO 290 MMS = 1,MMF
      CO 190 NSP = 1,NSPM
      TCX(NSP) = DCX(NSP)
      CCCONTINUE
190      DTIME = 0.5*DLTIME/FLCAT(MMF)
      ITER = 1
      CONTINUE
200      XM4 = TN2/3.0 + T02 + 3.0*TC
      XM5 = T02 + TNO + T0
      XM8 = XM4
      XM13 = XM4 - TAC + TNC2
      XM31 = 0.39*TN2 + 0.44*T02
      XM37 = TN2 + T02
      XM61 = XM37 + T0
      DO 255 NSP = 1, NSPM
      DCT = DCX(NSP)
      GC TO (201,202,203,204,205,206,207,208,209,210,211,212,213,214,
1      215,216,217,218,219,220,221,222,223,224,225), NSP
      MOLECULAR NITROGEN
      CONTINUE
      RCX(KN2) = DCT
      APT = 0.0
      BTT = 0.0
      OJT = 0.0
      GC TO 251
      MOLECULAR OXYGEN
      CONTINUE
      APT = 2.0*S32
202

```

SNAP5410
 SNAP5420
 SNAP5430
 SNAP5440
 SNAP5450
 SNAP5460
 SNAP5470
 SNAP5480
 SNAP5490
 SNAP5500
 SNAP5510
 SNAP5520
 SNAP5530
 SNAP5540
 SNAP5550
 SNAP5560
 SNAP5570
 SNAP5580
 SNAP5590
 SNAP5600
 SNAP5610
 SNAP5620
 SNAP5630
 SNAP5640
 SNAP5650
 SNAP5660
 SNAP5670
 SNAP5680
 SNAP5690
 SNAP5700
 SNAP5710
 SNAP5720
 SNAP5730
 SNAP5740
 SNAP5750
 SNAP5760

SNAP5770
SNAP5780
SNAP5790
SNAP5800
SNAP5810
SNAP5820
SNAP5830
SNAP5840
SNAP5850
SNAP5860
SNAP5870
SNAP5880
SNAP5890
SNAP5900
SNAP5910
SNAP5920
SNAP5930
SNAP5940
SNAP5950
SNAP5960
SNAP5970
SNAP5980
SNAP5990
SNAP6000
SNAP6010
SNAP6020
SNAP6030
SNAP6040
SNAP6050
SNAP6060
SNAP6070
SNAP6080
SNAP6090
SNAP6100
SNAP6110
SNAP6120

BTT = S4*XM4 + R31*XM31*TO + (S34 + R37*XM37)*TH + RATE(1)
QJT = ((R32*TO + R39*THO2 + R50*TO1C)*2.U + RATE(5)

1 + S31*XM31 + R33*TH + R38*TCH)*TO3
2 + (R4*XM4*TO + R34*TCH + R35*THO2)*TO
3 + (R41*TOH + R42*TH + R44*THO2)*THC2
GC TO 251

C
C
ATOMIC OXYGEN, O(3P)

CONTINUE

203

APT = 2.0*R4*XM4

BTT = R31*XM31*TO2 + R32*TO3 + R34*TOH + R35*THC2

1 + R43*TH2 + R46*TH2O2 + R56*TCH4 + R61*XM61*TOC

QJT = (2.0*S4*XM4 + S32*TO2 + S34*TH + RATE(8))*TOC

1 + (S31*XM31 + RATE(10))*TOC3 + R40*TOH*TOH + R51*TH2*TO1D

GC TO 251

C

C

OZONE

CONTINUE

204

APT = 0.0

BTT = S31*XM31 + R32*TO + R33*TH + R38*TOH + R39*THC2

1 + R50*TO1D + RATE(5)

QJT = (R31*XM31*TO + S32*TO2)*TO2

GC TO 251

C

C

HYDROXYL

CONTINUE

205

APT = 2.0*R40

ETT = R34*TO + R38*TO3 + R41*THO2 + R45*TH2O2 + R60*TOC

1 + R55*TCH4

QJT = (R35*THO2 + R43*TH2 + R46*TH2O2 + R56*TCH4)*TC

1 + (R52*TH2 + 2.0*R53*TH2O + R54*TCH4)*TO1D + R39*THC2*TOC3

2 + (R33*TO3 + S34*TO2 + 2.0*R36*THO2)*TH + 2.0E-5*RATE(5)*TH2O2

3 + RATE(6)*TH2O

GC TO 251

C

C

ATOMIC HYDROGEN

206 CONTINUE
 APT = 0.0
 BTT = (S34 + R37*XM37)*T02 + (R42 + R36)*TH02 + R33*T03
 1 + R47*TH202
 QJT = (R34*TOH + R43*TH2)*TC + R52*TO1D*TH2 + R60*TCH*TCC
 1 + RATE(6)*TH20 + RATE(7)*TCH4
 GO TO 251

C C WATER VAPOR
 207 CONTINUE
 APT = 0.0
 BTT = R53*TC1C + RATE(6)
 QJT = (R40*TOH + R41*TH02 + R45*TH2C2 + R55*TCH4)*TCH
 GO TO 251

C C FERHYDROXYL
 208 CONTINUE
 APT = 2.0*R44
 BTT = R35*TC + R39*T03 + R41*TOH + (R36 + R42)*TH
 QJT = (R45*TOH + R46*TO + R47*TH)*TH202 + R38*TCH*TC3
 1 + R37*XM37*TH*T02
 GO TO 251

C C HYDROGEN PEROXIDE
 209 CONTINUE
 APT = 0.0
 BTT = R45*TOH + R46*TC + R47*TH + 1.0E-2*RATE(5)
 QJT = R44*TH02*THC2
 GO TO 251

C C MOLECULAR HYDROGEN
 210 CONTINUE
 APT = 0.0
 BTT = R43*TC + R52*TO1D
 QJT = (R42*TH02 + R47*TH202)*TH
 GO TO 251

SNAP6130
 SNAP6140
 SNAP6150
 SNAP6160
 SNAP6170
 SNAP6180
 SNAP6190
 SNAP6200
 SNAP6210
 SNAP6220
 SNAP6230
 SNAP6240
 SNAP6250
 SNAP6260
 SNAP6270
 SNAP6280
 SNAP6290
 SNAP6300
 SNAP6310
 SNAP6320
 SNAP6330
 SNAP6340
 SNAP6350
 SNAP6360
 SNAP6370
 SNAP6380
 SNAP6390
 SNAP6400
 SNAP6410
 SNAP6420
 SNAP6430
 SNAP6440
 SNAP6450
 SNAP6460
 SNAP6470
 SNAP6480

SNAP6490
SNAP6500
SNAP6510
SNAP6520
SNAP6530
SNAP6540
SNAP6550
SNAP6560
SNAP6570
SNAP6580
SNAP6590
SNAP6600
SNAP6610
SNAP6620
SNAP6630
SNAP6640
SNAP6650
SNAP6660
SNAP6670
SNAP6680
SNAP6690
SNAP6700
SNAP6710
SNAP6720
SNAP6730
SNAP6740
SNAP6750
SNAP6760
SNAP6770
SNAP6780
SNAP6790
SNAP6800
SNAP6810
SNAP6820
SNAP6830
SNAP6840

```

C      METHANE
C      CCNTINUE
C      211  APT = 0.0
           BTT = RATE(7) + R54*TC10 + R55*TC11 + R56*TC12
           GJT = 0.0
           GO TO 251

C      CARBON DIOXIDE
C      CCNTINUE
C      212  APT = 0.0
           BTT = RATE(2) + 1.0E-9*ETA*TC10(NALT)
           XX12 = R60*TC10 + R61*XM61*TC10
           QJT = XX12*TC10
           GO TO 251

C      CARBON MONOXIDE
C      CCNTINUE
C      213  APT = 0.0
           BTT = XX12
           QJT = (RATE(2) + 1.0E-9*ETA*TC10(NALT))*TC02
           GO TO 251

C      ATOMIC OXYGEN, O(10)
C      CCNTINUE
C      214  APT = 0.0
           BTT = R50*TC3 + R51*TC2 + R52*TC1 + R53*TC0 + R54*TC4
           QJT = RATE(2)*TC02 + RATE(9)*TC02 + RATE(11)*TC3
           GO TO 251

C      ATOMIC NITROGEN, N(45)
C      CCNTINUE
C      215  GJT = ((S5*XM5 + S6*TC2 + S7*TC1)*2.0 + S2*TC0
           1  + S11*TC02)*TC2 + (S1*TC0 + S8*XM8 + S12*TC0 + S16)*TC0
           3  + 1.0E-14*TC2*TC2
           BTT = (R11 + R12)*TC02 + R8*XM8*TC0 + R2*TC0 + R1*TC2

```

SNAP6650
 SNAP6660
 SNAP6670
 SNAP6680
 SNAP6690
 SNAP6700
 SNAP6710
 SNAP6720
 SNAP6730
 SNAP6740
 SNAP6750
 SNAP6760
 SNAP6770
 SNAP6780
 SNAP6790
 SNAP7000
 SNAP7010
 SNAP7020
 SNAP7030
 SNAP7040
 SNAP7050
 SNAP7060
 SNAP7070
 SNAP7080
 SNAP7090
 SNAP7100
 SNAP7110
 SNAP7120
 SNAP7130
 SNAP7140
 SNAP7150
 SNAP7160
 SNAP7170
 SNAP7180
 SNAP7190
 SNAP7200

APT = (R5*XM5 + R6*TN2 + R7*TN)*2.0
 GO TO 251

NITRIC OXIDE

216 CONTINUE

QJT = ((R12*TN + S14*TAO2)*2.0 + S9 + R10*TO
 1 + S13*XM13)*TNO2 + (R1*TO2 + R8*XM8*TO)*TN

2 + (S2*TO + 2.0*S3*TO2)*TN2

3 + 5.0E-12*TN20*TO2

BYT = (R13*XM13 + R9 + S1)*TC + R2*TN + R21*TC3

1 + S8*XM8 + S10*TO2 + S16

2 + 0.1*RATE(7)

APT = (R14*TO2 + S12 + R3)*2.0

GO TO 251

NITROGEN DIOXIDE

217 CONTINUE

QJT = ((2.0*R14*TN + R13*XM13 + R9)*TO + R21*TC3
 1 + S12*TN + S10*TO2)*TNO + S11*TN2*TO2

BYT = (R11 + R12)*TN + S9 + R10*TO + S13*XM13

APT = 0.0

GO TO 251

NITROUS OXIDE

218 CONTINUE

QJT = 0.0

BYT = 0.0

APT = 0.0

GO TO 251

ATOMIC NITROGEN, N(2C)

219 CONTINUE

QJT = 1.5E-4*TEMPI*ETA*TNO(NALT)*TETOT(NALT)

BYT = 1.0E-14*TN2 + 5.0E-12*TO2

APT = 0.0

GO TO 251

SNAP7210
 SNAP7220
 SNAP7230
 SNAP7240
 SNAP7250
 SNAP7260
 SNAP7270
 SNAP7280
 SNAP7290
 SNAP7300
 SNAP7310
 SNAP7320
 SNAP7330
 SNAP7340
 SNAP7350
 SNAP7360
 SNAP7370
 SNAP7380
 SNAP7390
 SNAP7400
 SNAP7410
 SNAP7420
 SNAP7430
 SNAP7440
 SNAP7450
 SNAP7460
 SNAP7470
 SNAP7480
 SNAP7490
 SNAP7500
 SNAP7510
 SNAP7520
 SNAP7530
 SNAP7540
 SNAP7550
 SNAP7560

```

C 220 CCNTINUE
    QJT = 0.0
    BTT = 0.0
    APT = 0.0
    GO TO 251

C 221 CONTINUE
    GO TO 253
222 CONTINUE
    GO TO 253
223 CONTINUE
    GO TO 253
224 CONTINUE
    GO TO 253
225 CONTINUE
    GO TO 253
251 CONTINUE
    BTT = BTT + BTEM(DNSP)
    QJT = QJT + QJEM(DNSP)
    IF (APT .LE. 0.0) GO TO 252
    FAC4AG = 4.0*APT*QJT
    BTSQ = BTT*BTT
    FACRT = SQRT(FAC4AG + BTSQ)
    DLT = SQRT(FAC4AG + BTSQ)
    FACDMB = DLT - BTT
    IF (FACRT .LT. 0.01) FACDME = (((U.027344 - 0.0205*FACRT)
1 *FACRT - 0.0390625)*FACRT + 0.0625)*FACRT - 0.125)*FACRT
2 + 0.5)*FACRT*BTT
    FAC2AX = 2.0*APT*DLT
    FACB = DLT*DLT
    FAC = EXP(AMAX1(-675.0,AMIN1(-FACB,675.0)))
    FACM = 1.0 - FAC
    IF (FACB .LT. 0.01) FACM = EXP(FACB)
    RCX(DNSP) = ((BTT + DLT)*FAC + FACDMB)*DCT + 2.0*QJT*FACM
1 /((FAC2AX + BTT)*FAC + (FAC + 1.0)*DLT)
  
```

SNAP7570
 SNAP7580
 SNAP7590
 SNAP7600
 SNAP7610
 SNAP7620
 SNAP7630
 SNAP7640
 SNAP7650
 SNAP7660
 SNAP7670
 SNAP7680
 SNAP7690
 SNAP7700
 SNAP7710
 SNAP7720
 SNAP7730
 SNAP7740
 SNAP7750
 SNAP7760
 SNAP7770
 SNAP7780
 SNAP7790
 SNAP7800
 SNAP7810
 SNAP7820
 SNAP7830
 SNAP7840
 SNAP7850
 SNAP7860
 SNAP7870
 SNAP7880
 SNAP7890
 SNAP7900
 SNAP7910
 SNAP7920

```

252 GO TO 253
    CONTINUE
    FACB = OTT*OTIME
    FAC = EXP(AMAX1(-675.0,AMIN1(-FACB,675.0)))
    FACM = 1.0 - FAC
    IF (FACB.LT. 0.01) FACM = EXPH(FACB)
    RCX(NSP) = OCT*FAC + FACM*OJT/BTT
    CONTINUE
    CONTINUE
    ITER = ITER + 1
    IF (ITER.GT. 2) GO TO 260
    DO 258 NSP = 1, NSPM
    TCX(NSP) = RCX(NSP)
    CONTINUE
    OTIME = 2.0*OTIME
    IF (ITER.EQ. 2) GC TC 200
    CONTINUE
    CONTINUE
    JO 280 NSP = 1, NSPM
    DCX(NSP) = ABS(RCX(NSP))
    CONTINUE
    CONTINUE
    IF (MLMAX.EQ. 1) GO TO 310
    CALL ROMTST (ML, NSPM, CCX, TST, TCX, KTEST, EPS)
    IF (KTEST.EQ. 0) GO TC 310
    CONTINUE
    CONTINUE
    ML = 100*KTEST + MLSAVE
    MLPRNT(NALT) = ML
    IF (KTEST.EQ. 0) GO TO 318
    DO 315 NSP = 1, NSPM
    TCX(NSP) = RCX(NSP)
    CONTINUE
    CONTINUE
    DO 320 NSP = 1, NSPM
    CCNSP(NAMIN, NSP) = SCX(NSP)
  
```

```

SCX(NSP) = ABS(TCX(NSP))
IF(NALT.EQ. NTOP) CONSP(NALT,NSP) = SCX(NSP)
IF(NALT.EG. NTOP)
1  CONSP(NPLU,NSP) = SCX(NSP)*EXP(-1.0E5*AMWT(NSP)*CFACNA)
320 CONTINUE
QCOHA = R33*TCX(KH)*TCX(KO3)
QCOHB = R35*TCX(KH02)*TCX(KC)
OZOH A = OZOH A + 1.0E5*QCOHA
OZOH B = OZOH B + 1.0E5*QCOHB
IF(MOD(M,NPRNT).NE. 0) GC TO 35U
WRITE(6,6) NA,QCOHA,QCOHB,CZOH A,DZCHB,(TCX(I),I=1,6)
WRITE(6,6) ML,(TCX(I),I=7,16)
350 CONTINUE
IF(MOD(M,NPRNT).EQ. 0)
1  WRITE(6,30) (MLPRNT(I), I = NBASE, NTOP)
TIME = TIME + DLTIME
IF(TIME.GT. 4.32E4) TIME = TIME - 8.64E4
400 CONTINUE
TIME = TIME - 0.5*DLTIME
WRITE(4,15) TIME,CARD
NB = NBASE - 1
NT = NTOP + 1
DO 420 NALT = 1, NT
WRITE(4,34) NALT,N1,(CONSP(NALT,I), I = 1, 7)
WRITE(4,34) NALT,N2,(CONSP(NALT,I), I = 8, 14)
WRITE(4,34) NALT,N3,(CONSP(NALT,I), I = 15, 20)
420 CONTINUE
IS = 1
DO 440 IP = 1, 2
IF = IS + 9
NS = 1
DO 435 IX = 1, 5
NF = NS + 50
WRITE(6,20) IDENT,NPAGE
NPAGE = NPAGE + 1
WRITE(6,6) NTIME,RTIME

```

SNAP7930
 SNAP7940
 SNAP7950
 SNAP7960
 SNAP7970
 SNAP7980
 SNAP7990
 SNAP8000
 SNAP8010
 SNAP8020
 SNAP8030
 SNAP8040
 SNAP8050
 SNAP8060
 SNAP8070
 SNAP8080
 SNAP8090
 SNAP8100
 SNAP8110
 SNAP8120
 SNAP8130
 SNAP8140
 SNAP8150
 SNAP8160
 SNAP8170
 SNAP8180
 SNAP8190
 SNAP8200
 SNAP8210
 SNAP8220
 SNAP8230
 SNAP8240
 SNAP8250
 SNAP8260
 SNAP8270
 SNAP8280

```

WRITE (6,19)
WRITE (6,22) LALT,(LSF(I), I = IS, IF)
WRITE (6,19)
DO 430 NALT = NS, NF
  NA = NALT - 1
  WRITE (6,6) NA,(CONSP(NALT,I), I = IS, IF)
  CONTINUE
  NS = NF
  CONTINUE
  IS = 11
  CONTINUE
  NBASE = NBASE - 1
  NTOP = NTOP + 1
  CALL SECOND (SECB)
  SECC' = SECB - SECA
  WRITE (NH,TFMT) SECA,SECB,SECC
  CONTINUE
  WRITE (6,20) IDENT,NPAGE
  NPAGE = NPAGE + 1
  CONTINUE
  WRITE (6,20) IDENT,NPAGE
  NPAGE = NPAGE + 1
  STOP
  6  FORMAT (1X,I6,1P10E12.3)
  14  FORMAT (I6,1P8E9.3)
  15  FORMAT (1X,F11.4,11A6,A2)
  16  FORMAT (1X,I11,11A6,A2)
  19  FORMAT (1X)
  20  FORMAT (50H1***** SNAP S2 CHEMISTRY ***** HARF DEVELOPMENT,
1  8H ***** ,10A6,13H ***** PAGE ,X3/)
  21  FORMAT (10A6,4A5)
  22  FORMAT (1X,A6,1U(3X,A6,3X))
  24  FORMAT (I6,3X,6(1P9.3,1H),3X,A3,I3,I2)
  26  FORMAT (7X,1P6E18.9)
  30  FORMAT (/1X,30I4)
  34  FORMAT (I4,I2,1X,1P7E10.4)
  35  FORMAT (5X,I4,I2,1X,1F7E12.4)
END

```

SNAP8250
 SNAP8300
 SNAP8310
 SNAP8320
 SNAP8330
 SNAP8340
 SNAP8350
 SNAP8360
 SNAP8370
 SNAP8380
 SNAP8390
 SNAP8400
 SNAP8410
 SNAP8420
 SNAP8430
 SNAP8440
 SNAP8450
 SNAP8460
 SNAP8470
 SNAP8480
 SNAP8490
 SNAP8500
 SNAP8510
 SNAP8520
 SNAP8530
 SNAP8540
 SNAP8550
 SNAP8560
 SNAP8570
 SNAP8580
 SNAP8590
 SNAP8600
 SNAP8610
 SNAP8620
 SNAP8630
 SNAP8640
 SNAP8650
 SNAP8660


```

C C SUBROUTINE COLU44 (HA,GHI,COL,ETA)
C C COMPUTES THE INTEGRATED COLUMNAR NUMBER DENSITIES FOR
C C COL(1) THE COLUMNAR NUMBER DENSITY FOR 02
C C COL(2) THE COLUMNAR NUMBER DENSITY FOR N2
C C COL(3) THE COLUMNAR NUMBER DENSITY FOR O3
C C ALONG A PATH BEGINNING AT ALTITUDE HA AND EXTENDING TO INFINITY
C C AND MAKING AN ANG-E CHI(RADIANS) WITH RESPECT TO THE LOCAL
C C VERTICAL AT HA
C C CHI=SOLAR ZENITH ANGLE
C C COMMON CONSP(501,25)
C C DIMENSION CONC(20),COL(5),TERM(5)
C C DATA RE /6371.0/
C C DATA A23,A43 / 0.66666666666667, 1.33333333333333 /
C C RHA = RE + HA
C C RHIN = RHA*SIN(CHI)
C C IF (RHIN .LE. RE) RHIN = RE + 0.0001
C C RHINSQ = RHIN*RHIN
C C COSCHI = COS(CHI)
C C HMIN = ABS(RHA*COSCHI)
C C NPT = HA + 1.001
C C DO 110 I = 1,4
C C TERM(I) = 0.0
C C CONTINUE
C C ISW = 1
C C R1 = RHA + 1.0
C C HSUB = HMIN
C C IF (COSCHI .GE. 0.0) GO TO 215
C C ISW = 0
C C LPT = RHIN + 1.0 - RE
C C R1 = FLOAT(LPT) + RE
C C H1 = SQRT(R1*R1 - RHINSQ)
C C HSUB = H1
C C R1 = R1 + 1.0
C C NA = LPT + 1
C C CALL CONCEN(RMIN - RE,CONC,ETA)
C C IF (100(NPT-LPT,2) .EQ. 0) GO TO 130
110

```

```

H13 = H1/3.0
DO 120 I = 1,4
  IF (I .EQ. 3) GO TO 120
  TERM(I) = (2.0*CONC(I) + CONSP(NA,I))*H13
  CONTINUE
120 NS = LPT + 1
    GO TO 150
130 CONTINUE
  H2 = SQRT(R1*R1 - RMINSQ) - HSUB
  HSUB = HSUB + H2
  R1 = R1 + 1.0
  H12 = 0.5*H1/H2
  H21 = 0.5*H2/H1
  HS3 = (H1 + H2)/3.0
  NB = NA + 1
  DO 140 I = 1,4
    IF (I .EQ. 3) GO TO 140
    YA = CONC(I)
    YB = CONSP(NA,I)
    YC = CONSP(NB,I)
    TERM(I) = ((YB - YC)*H12 + (YB - YA)*H21 + YA + YB + YC)*HS3
    CONTINUE
140 NS = LPT + 2
    CONTINUE
150 IF (NS .EQ. NPT) GO TO 205
    NF = NPT - 2
    CONTINUE
170 DO 200 VA = NS, VF, 2
    NB = NA + 1
    NC = NB + 1
    H1 = SQRT(R1*R1 - RMINSQ) - HSUB
    R1 = R1 + 1.0
    HSUB = HSUB + H1
    H2 = SQRT(R1*R1 - RMINSQ) - HSUB
    R1 = R1 + 1.0
    HSUB = HSUB + H2

```

```

CLMN 370
CLMN 380
CLMN 390
CLMN 400
CLMN 410
CLMN 420
CLMN 430
CLMN 440
CLMN 450
CLMN 460
CLMN 470
CLMN 480
CLMN 490
CLMN 500
CLMN 510
CLMN 520
CLMN 530
CLMN 540
CLMN 550
CLMN 560
CLMN 570
CLMN 580
CLMN 590
CLMN 600
CLMN 610
CLMN 620
CLMN 630
CLMN 640
CLMN 650
CLMN 660
CLMN 670
CLMN 680
CLMN 690
CLMN 700
CLMN 710
CLMN 720

```

```

H12 = 0.5*H1/H2
H21 = 0.5*H2/H1
HS3 = (H1 + H2)/3.0
DO 180 I = 1,4
  IF (I.EQ. 3) GO TO 180
  YA = CONSP(NA,I)
  YB = CONSP(NB,I)
  YC = CONSP(NC,I)
  TERM(I) = ((YB - YC)*H12 + (YB - YA)*H21 + YA + YB + YC)*HS3
  + TERM(I)
180 CONTINUE
200 CONTINUE
  IF (ISM.EQ. 1) GO TO 220
205 CONTINUE
  ISM = 1
DO 210 I = 1, 4
  TERM(I) = 2.0*TERM(I)
210 CONTINUE
215 CONTINUE
  NS = NPT
  NF = MIN0(NPT + 30,198 + MOD(NPT,2))
  IF (NPT.GT. 199) GO TO 220
  IF (CHI.EQ. 0.0) GO TO 250
  GO TO 170
220 CONTINUE
  COL(1) = 1.0E5*TERM(2)
  COL(2) = 1.0E5*TERM(1)
  COL(3) = 1.0E5*TERM(4)
  RETURN
250 CONTINUE
DO 260 I = 1, 4
  TERM(I) = 0.333333333333*CONSP(NPT,I)
260 CONTINUE
DO 280 NA = NS, N, 2
  NB = NA + 1
  NC = NB + 1

```

```

CLMN 730
CLMN 740
CLMN 750
CLMN 760
CLMN 770
CLMN 780
CLMN 790
CLMN 800
CLMN 810
CLMN 820
CLMN 830
CLMN 840
CLMN 850
CLMN 860
CLMN 870
CLMN 880
CLMN 890
CLMN 900
CLMN 910
CLMN 920
CLMN 930
CLMN 940
CLMN 950
CLMN 960
CLMN 970
CLMN 980
CLMN 990
CLMN1000
CLMN1010
CLMN1020
CLMN1030
CLMN1040
CLMN1050
CLMN1060
CLMN1070
CLMN1080

```

CLMN1090
CLMN1100
CLMN1110
CLMN1120
CLMN1130
CLMN1140
CLMN1150

```
DO 270 I = 1, 4  
  IF (I.EQ. 3) GO TO 270  
  TERM(I) = A43*CONSP(N3,I) + A23*CONSP(NC,I) + TERM(I)  
  270 CONTINUE  
  280 CONTINUE  
      GO TO 220  
      END
```

SUBROUTINE CONCEN(H, CONC, ETA)

PROVIDES AMBIENT CONCENTRATIONS AS A FUNCTION OF ALTITUDE

CONC(1) = NO.DENSITY OF N2 IN PARTICLES/CC

CONC(2) = NO.DENSITY OF O2 IN PARTICLES/CC

CONC(3) = NO.DENSITY OF O IN PARTICLES/CC

CONC(4) = NO.DENSITY OF O3 IN PARTICLES/CC

THIS VERSION (1 MARCH 1971) IS FOR SHAPS2 ONLY.

COMMON CONSP(501,25)

DIMENSION CONC(25)

DATA JMAX /10/

IF (1.LE. 200.0) GO TO 115

DO 110 J = 1, JMAX

CONC(J) = 0.0

CONTINUE

GO TO 200

CONTINUE

K = 1AX0(2, IFIX(H + 1.5))

K = 1IN0(200, K)

HX = H - FLOAT(K - 1)

DO 130 I = 1, 4

CX = CONSP(K, I)

BX = 0.5*(CONSP(K+1, I) - CONSP(K-1, I))

AX = BX + CONSP(K-1, I) - CX

CONC(I) = (AX*HX + BX)*HX + CX

CONTINUE

CONTINUE

RETURN

END

CONC 10
CONC 20
CONC 30
CONC 40
CONC 50
CONC 60
CONC 70
CONC 80
CONC 90
CONC 100
CONC 110
CONC 120
CONC 130
CONC 140
CONC 150
CONC 160
CONC 170
CONC 180
CONC 190
CONC 200
CONC 210
CONC 220
CONC 230
CONC 240
CONC 250
CONC 260
CONC 270
CONC 280
CONC 290
CONC 300
CONC 310
CONC 320
CONC 330
CONC 340
CONC 350
CONC 360
CONC 370

SUBROUTINE ROMTST (M,N,DINT,T,RINT,KTEST,EPS)

M IS INDEX OF M-TH ENTRY TO ROMTST DURING SINGLE INTERVAL
 N IS NUMBER OF DIFFERENTIAL EQUATIONS
 DINT IS VECTOR OF ENDPPOINT INTEGRATION VALUES
 T IS ARRAY IN WHICH ROMBERG EXTRAPOLATION IS COMPUTED
 RINT IS VECTOR IN WHICH FINAL INTEGRALS ARE RETURNED
 KTEST IS SET TO 1 IF CONVERGENCE CRITERION IS NOT SATISFIED
 EPS IS RELATIVE ERROR ALLOWED

DIMENSION DINT(50)
 DIMENSION RINT(50)
 DIMENSION T(50,10)

IF (M .GT. 1) WRITE (6,6) (DINT(I), I = 1, N)

KTEST = 0

TIK = 1.0E14

DO 150 I = 1, N

T(I,4) = 0.0

K = 1

CONTINUE

K = K + 1

IF (K .GT. M) GO TO 140

IF (K .GT. 1) GO TO 130

TA = T(I,1)

CA = DINT(I)

T(I,1) = CA

WA = CA - TA

SUM = CA

FAC = 1.0

GO TO 110

CONTINUE

TB = TA

TA = T(I,K)

RTST 10
 RFST 20
 RTST 30
 RTST 40
 RTST 50
 RTST 60
 RTST 70
 RTST 80
 RTST 90
 RTST 100
 RTST 110
 RTST 120
 RTST 130
 RTST 140
 RTST 150
 RTST 160
 RTST 170
 RTST 180
 RTST 190
 RTST 200
 RTST 210
 RTST 220
 RTST 230
 RTST 240
 RTST 250
 RTST 260
 RTST 270
 RTST 280
 RTST 290
 RTST 300
 RTST 310
 RTST 320
 RTST 330
 RTST 340
 RTST 350
 RTST 360

RTST 370
RTST 380
RTST 390
RTST 400
RTST 410
RTST 420
RTST 430
RTST 440
RTST 450
RTST 460
RTST 470
RTST 480
RTST 490
RTST 500
RTST 510
RTST 520
RTST 530
RTST 540
RTST 550
RTST 560
RTST 570
RTST 580

```

FAC = 4.0*FAC
FACB = FAC*TB
DEN = WA/(FACB - CA)
JOV = _EGVAR(DEN)
IF (JOV .NE. 0) DEN = 0.0
TIK = CA*DEN
T(I,<) = TIK
CA = FACB*DEN
WA = CA - TA
SUM = SUM + TIK
IF (TIK .EQ. 0.0) GO TO 145
GO TO 110
CONTINUE
IF (ABS(TIK) .GT. EPS*(1.0E5 + SUM)) KTEST = 1
CONTINUE
RINT(I) = SUM
CONTINUE
WRITE (6,7) M,KTEST,(RINT(I), I = 1, N)
RETURN
FORMAT (7X,1P6E13.9)
FORMAT (1X,2I3,1P6E13.9)
END

```

140
145
150
C
C
C

```

SUBROUTINE SOLLY (OT-ETA,DALT,TIME,SOLDEC,CHI,ETA)
DATA PI /3.1415926535898/
DATA RE /6371.0/
PID2 = PI*0.5
T = TIME
IF (T.GT. 4.32E4) T = T - 8.64E4
XCRIT = ACOS(RE/(RE + DALT)) + PID2
XMIN = XCRIT-PI/720.0
XMAX = XCRIT+PI/720.0
OLAT = PID2-OT-ETA
SINLAT = SIN(OLAT)
COSLAT = COS(OLAT)
SINDEC = SIN(SOLDEC)
COSDEC = COS(SOLDEC)
CHI = ACOS(SINLAT*SINDEC+COSLAT*COSDEC*COS(2.*PI*T/86400.))
IF(CHI.LE.XMIN) GO TO 1
IF (CHI.GE. XMAX) GO TO 2
HDR = (CHI-XMIN)*720.0/PI
ETA = 1.-(ACOS(1.-HDR)-(1.-HDR)*SQRT(2.*HDR-HDR*HDR))/PI
GO TO 3
1 ETA = 1.0
GO TO 3
2 ETA = 0.0
3 RETURN
END

```

SOLL 10
SOLL 20
SOLL 30
SOLL 40
SOLL 50
SOLL 60
SOLL 70
SOLL 80
SOLL 90
SOLL 100
SOLL 110
SOLL 120
SOLL 130
SOLL 140
SOLL 150
SOLL 160
SOLL 170
SOLL 180
SOLL 190
SOLL 200
SOLL 210
SOLL 220
SOLL 230
SOLL 240
SOLL 250

1
2
3


```

C
C
SUBROUTINE SUNPHI(DENSCO,ETA,RATE)
      THIS SUBROUTINE COMPUTES THE PHOTODISSOCIATION RATES AT A
      POINT IN THE ATMOSPHERE DUE TO THE SOLAR FLUX
      DIMENSION PHIZ(30)
      DIMENSION DENSQ(10)
      DIMENSION RATE(30)
      DIMENSION PHITOP(30)
      DIMENSION XS(30,10)
      DATA PHITOP / 4.59E03,2.62E09,2.86E09,2.00E09,2.59E09,1.30E10,
1 1.35E09,3.10E11,7.97E09,1.44E10,2.28E10,2.07E10,4.05E10,6.96E10,
2 1.15E11,2.06E11,5.52E11,1.28E12,2.57E12,5.90E12,1.56E13,6.39E13,
3 1.05E14,1.26E14,2.97E14,6.68E14,1.93E15,3.67E15,5.95E15,8.68E15/
      DATA (XS(I),I=1,30)
1 1.51E-18,1.03E-18,7.91E-19,5.51E-19,
2 6.03E-19,1.13E-18,2.59E-19,1.03E-20,2.91E-19,5.84E-19,3.33E-18,
3 1.37E-17,1.40E-17,1.10E-17,6.54E-18,2.96E-18,7.10E-19,2.25E-20,
4 1.25E-22,2.50E-23,1.90E-23,7.30E-24,4.12E-24,1.00E-24,6*0.0/
      DATA (XS(I),I=31,60)
1 6*0.0,1.59E-17,2.29E-17,6.20E-18,
2 1.23E-17,1.47E-17,6.78E-18,5.16E-18,3.44E-18,1.86E-18,9.36E-19,
3 8.25E-19,7.95E-19,6.14E-19,3.87E-19,4.07E-19,1.13E-18,3.88E-18,
4 8.82E-18,1.03E-17,5.22E-18,7.35E-19,3.65E-20,1.03E-21,0.0/
      DATA (XS(I),I=61,90)
1 7*0.0,1.44E-17,6*0.0,2.65E-18,
2 4.23E-18,3.80E-18,1.11E-18,6.60E-20,11*0.0/
      DATA (XS(I),I=91,120)
1 7*0.0,7.32E-20,6*0.0,
2 2.6E-19,8.6E-20,2.0E-20,6.7E-21,12*0.0/
      DATA XS8CH4 /1.50E-17/
      XS(I,I) ARE MOLECULAR OXYGEN TOTAL CROSS SECTIONS.
      VALUES FOR I = 1, 30 ARE TAKEN FROM ACKERMAN (1970).
      VALUES FOR I = 1,6 WILL BE REVISED LATER
      XS(I,2) ARE OZONE CROSS SECTIONS FROM ACKERMAN (1970).
      XS(I,3) ARE WATER VAPOR CROSS SECTIONS.
      THE ONLY CROSS SECTION USED FOR CH4 IS XS8CH4 AT 1215.7 A.
      RATE( 1) IS TOTAL O2 PHOTODISSOCIATION RATE
      RATE( 2) IS TOTAL O3 PHOTODISSOCIATION RATE
      RATE( 3) IS TOTAL NO PHOTODISSOCIATION RATE
      RATE( 4) IS TOTAL N2 PHOTODISSOCIATION RATE
      RATE( 5) IS TOTAL O3 PHOTODISSOCIATION RATE
      SPHI 10
      SPHI 20
      SPHI 30
      SPHI 40
      SPHI 50
      SPHI 60
      SPHI 70
      SPHI 80
      SPHI 90
      SPHI 100
      SPHI 110
      SPHI 120
      SPHI 130
      SPHI 140
      SPHI 150
      SPHI 160
      SPHI 170
      SPHI 180
      SPHI 190
      SPHI 200
      SPHI 210
      SPHI 220
      SPHI 230
      SPHI 240
      SPHI 250
      SPHI 260
      SPHI 270
      SPHI 280
      SPHI 290
      SPHI 300
      SPHI 310
      SPHI 320
      SPHI 330
      SPHI 340
      SPHI 350
      SPHI 360

```

```

C      RATE( 6) IS TOTAL H2O PHOTODISSOCIATION RATE
C      RATE( 7) IS TOTAL CH4 PHOTODISSOCIATION RATE
C      RATE( 8) IS RATE OF O(3P) PRODUCTION FROM O2 PHOTODISSOCIATION
C      RATE( 9) IS RATE OF O(1D) PRODUCTION FROM O2 PHOTODISSOCIATION
C      RATE(10) IS RATE OF O(3P) PRODUCTION FROM O3 PHOTODISSOCIATION
C      RATE(11) IS RATE OF O(1D) PRODUCTION FROM O3 PHOTODISSOCIATION
C      FLUX VALUES FOR INTERVALS 1-6 WILL BE REVISED LATER
C      INPUT- DENS2CO(I)=3 COLUMN NUMBER DENSITY FROM SOURCE TO OBSERVER
C      IN PARTICLES PER SQ CM.
C      I=1 IS O2, I=2 IS N2 AND I=3 IS O3.
C      ETA IS THE FRACTION OF THE SOLAR DISK SHOWING.
C      IT COMES FROM SUBROUTINE SOLLY AND RANGES FROM 0 TO 1.
C      OUTPUT- RATE(I) = REACTION RATES IN REACTIONS PER SEC. I=1 IS O2, I=2 IS O2, I=3 IS NO AND I=4 IS NO2 AT THE OBSERVER.
C      RATE(I) IS O3 PHOTODISSOCIATION RATE
C      SOLAR FLUX AT Z= INFINITY IN PHOTONS PER SQ CM PER SEC. PHITOP(I)
C      FOR I=1,30 IS THIS FLUX BROKEN INTO 30 EQUAL WINDOWS OF 0.25
C      RECIPROCAL MICRONS CENTERED AT 9.875 (I=1) TO 2.625 (I=30)
C      RECIPROCAL MICRONS.
C      SET RATES=0 IF ETA=0.
C      DO 110 I = 1, 11
C      RATE(I) = 0.0
C      110 CONTINUE
C      IF (ETA .LE. 0.0) GO TO 500
C      COMPUTATION OF SOLAR FLUX REACHING HEIGHT Z IN PHOTONS PER SQ CM
C      PER SEC. THIS IS THE PORTION OF THE SOLAR FLUX THAT HAS NOT BEEN
C      ABSORBED ENROUTE. IT IS MODULATED BY ETA.
C      PHIZ(I) FOR I=1,30 ARE THE THIRTY WINDOWS OF THE SOLAR FLUX
C      REACHING THE OBSERVER
C      COLO2 = ABS(DENS2CO(1))
C      COLO3 = ABS(DENS2CO(3))
C      DO 120 I = 7, 29

```

```

SPHI 370
SPHI 380
SPHI 390
SPHI 400
SPHI 410
SPHI 420
SPHI 430
SPHI 440
SPHI 450
SPHI 460
SPHI 470
SPHI 480
SPHI 490
SPHI 500
SPHI 510
SPHI 520
SPHI 530
SPHI 540
SPHI 550
SPHI 560
SPHI 570
SPHI 580
SPHI 590
SPHI 600
SPHI 610
SPHI 620
SPHI 630
SPHI 640
SPHI 650
SPHI 660
SPHI 670
SPHI 680
SPHI 690
SPHI 700
SPHI 710
SPHI 720

```

```

120  ATTE4 = COLO2*XS(I,1) + COLO3*XS(I,2)
      PHIZ(I) = ETA*PHITOP(I)*EXP(-ATTEN)
      CONTINUE
      DJ1 = 0.0
      DO 133 I = 7, 17
        DJ1 = DJ1 + PHIZ(I)*XS(I,1)
130  CONTINUE
      DJ2 = 1.46E-7*ETA/((COLO2*5.71821E-19)**0.793455 + 1.0)
      DO 140 I = 21, 24
        DJ2 = DJ2 + PHIZ(I)*XS(I,1)
140  CONTINUE
      DJ3 = PHIZ(8)*XS(8,2)
      DO 150 I = 18, 27
        DJ3 = DJ3 + PHIZ(I)*XS(I,2)
150  CONTINUE
      DJ4 = PHIZ(28)*XS(28,2) + PHIZ(29)*XS(29,2)
      DJ5 = PHIZ(8)*XS(8,3)
      DO 160 I = 15, 19
        DJ5 = DJ5 + PHIZ(I)*XS(I,3)
160  CONTINUE
      DJ6 = PHIZ(8)*XS(8,4)
      DO 170 I = 15, 18
        DJ6 = DJ6 + PHIZ(I)*XS(I,4)
170  CONTINUE
      DJ7 = (PHIZ(7) + PHIZ(8))*1.0E-19
      DO 180 I = 9, 19
        DJ7 = DJ7 + 1.0E-18*PHIZ(I)
180  CONTINUE
      RATE(1) = DJ1 + DJ2
      RATE(2) = DJ6
      RATE(3) = DJ7
      RATE(4) = DJ7
      RATE(5) = DJ3 + DJ4
      RATE(6) = DJ5
      RATE(7) = XS8CH4*PHIZ(8)
      RATE(8) = RATE(1) + DJ2

```

SPHI 730
 SPHI 740
 SPHI 750
 SPHI 760
 SPHI 770
 SPHI 780
 SPHI 790
 SPHI 800
 SPHI 810
 SPHI 820
 SPHI 830
 SPHI 840
 SPHI 850
 SPHI 860
 SPHI 870
 SPHI 880
 SPHI 890
 SPHI 900
 SPHI 910
 SPHI 920
 SPHI 930
 SPHI 940
 SPHI 950
 SPHI 960
 SPHI 970
 SPHI 980
 SPHI 990
 SPHI1000
 SPHI1010
 SPHI1020
 SPHI1030
 SPHI1040
 SPHI1050
 SPHI1060
 SPHI1070
 SPHI1080

SPHI1090
SPHI1100
SPHI1110
SPHI1120
SPHI1130
SPHI1140

RATE(9) = DJ1
RATE(10) = DJ4
RATE(11) = DJ3
500 CONTINUE
RETURN
END

```

C C C C C C C
FUNCTION TKEL (Z)
RETURNS TEMPERATURES FOR WINTER 30 DEGREE LATITUDE MODEL
ATMOSPHERE IF Z IS LESS THEN 120 KM, AND APPROXIMATE
TEMPERATURES FOR HIGH ALTITUDE WINTER MODEL ATMOSPHERE WITH
EXOSPHERE TEMPERATURE 1000 K IF Z IS GREATER THAN 120 KM.
Z IS INPUT ALTITUDE, KM.
TKEL RETURN VALUE IS ABSOLUTE TEMPERATURE, DEGREES KELVIN.
LOWER ATMOSPHERE VALUES COMPUTED BY LINEAR INTERPOLATION.
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, 86.107, 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 CCNTINUE
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

```

```

PROGRAM BCKGND (INPUT,OUTPUT,TAPE5=INPLT,TAPE6=OUTPUT,
1 TAPE1,TAPE2)
DIMENSION XIO(25,4),XIGT(4),ALT(30),STORE(30),IALI(25)
DIMENSION CATATY(25),CDEXT(25),CDEXE(25),NLEVEL(25),NEANC(25)
DIMENSION LSC(25,10),LCC(25,10),RCM(25,10),BV(25,10),AMASS(25)
DIMENSION ZANG(7),STAU(7),SRAL(8),RGN(8),RATIO(8)
DIMENSION LEC(25,20),LBU(25,20),STR(25,20),SFLX(25,25),
1 7FLUX(25,25,3)
DIMENSION CC2(18,2),WATVA(18,2)
REAL LAMCA
COMMON /VERCOM/ I,CO2,WATVA,CDEXT,CDEXE
COMMON /TECCOM/ NLTE
COMMON/INOLT/NR,NP,NH
COMMON CCNC(161,20),TAU(161,20),RAC(161,20),T(161),LAMBDA(25),
1 TOTJEN(161)
DATA(ZANG(1),I=1,7)/0.0,20.0,40.0,60.0,70.0,80.0,90.0/
DATA(CATATY(1),I=1,25)/8METHANE,8HCAREON D,8WATER VA,8NITRIC
1,8NITROUS,8OZONE PR,8HCZONE NO,18*8H
NN = 5
NP = 2.
NTY = 9
NW = 0
PCODE = 1.0
REWIND 1
REWIND 2
10 FCRMAT(12)
READ(NR,10) IGAS
20 FCRMAT(14,22)NUMBER OF GAS FRCFILES, I3)
WRITE(NW,20) IGAS
IGAS=IGAS+1
DO 170 J=1,IGAS
30 FCRMAT(4A6,I3,F7.0)
READ(NR,30)(XIO(J,K),K=1,4), ICHK,ALIM
IF(J.EQ.1CHK) GC TO 50
40 FCRMAT(7F10 CARD, A6, 20)OUT OF ORDER FOR GAS, I3)
WRITE(NW,40)(XIO(J,K),K=1,4), J

```

```

50 DO 105 K=1,6
   K1=(K-1)*5+1
   K2=K1+4
   60 FORMAT (5(F5.0,E8.1,2X),12,I3)
   READ(NR,60) (ALT(L),STORE(L),L=K1,K2), ICHK, KCHK
   IF (ICLK.EC.J) GO TO 80
   70 FORMAT (30HCATA CARD CUT CF CRDER FCR GAS,I3, 3X, 8+CARD NC., I3)
   IFIVE = 5
   WRITE (NH,70) J,IFIVE
   80 IF (KCHK.EC.K) GO TO 90
   WRITE(NH,70) J,K
   90 DO 100 L=K1,K2
     IF (ALT(L).LT.ALTH) GO TO 99
     KSET=L
     GC TC 120
     99 CONTINUE
     100 CONTINUE
     105 CONTINUE
     110 FORMAT (42HMAX ALT. NOT FOUND CN DATA CARDS - GAS NC., I3)
     WRITE(NH,110) J
     120 DO 130 L=1,KSET
       STORE(L)=ALCG(STORE(L))
       130 STORE(L)=ALCG(STORE(L))
       IF (ALTH.GT.150.) GO TC 140
       IALT(J)=ALTH-55.
       GO TO 150
       140 IALT(J)=51.+(ALTH-150.)/5.
       150 CALL INTERF(ALT,STORE,KSET,1,J)
       L1=IALT(J)
       DO 160 L=1,L1
         CONC(L,J)=EXP(CCNC(L,J))
       160 CONC(L,J)=EXP(CCNC(L,J))
       170 CONTINUE
       DO 175 J=1,161
         TGTDEN(J)=CCNC(J,IGAS)
       175 TGTDEN(J)=CCNC(J,IGAS)
       READ(NR,30) (XICT(K),K = 1,4), ICHK, KCHK
       IF (ICLK.EC.1) GO TO 190
       180 FORMAT (7HIC CARD, 4A6,26+CLT CF CADER FOR TEMP DATA)

```

```

EGND 370
EGND 380
EGND 390
EGND 400
EGND 410
EGND 420
EGND 430
EGND 440
EGND 450
EGND 460
EGND 470
EGND 480
EGND 490
EGND 500
EGND 510
EGND 520
EGND 530
EGND 540
EGND 550
EGND 560
EGND 570
EGND 580
EGND 590
EGND 600
EGND 610
EGND 620
EGND 630
EGND 640
EGND 650
EGND 660
EGND 670
EGND 680
EGND 690
EGND 700
EGND 710
EGND 720

```

```

150 WRITE(NW,160) (XIOT(K),K=1,4)
    CC 245 K=1,6
    K1=(K-1)*5+1
    K2=K1+4
    READ(NR,60)(ALT(L),STORE(L),L=K1,K2), ICHK, KCHK
    IF(ICHK.EQ.1) GO TO 210
    FCRMAT(31HTEMP DATA CARD CUT CF CRDER NO.,13)
    WRITE(NW,200) K
    210 IF(KCHK.EQ.K) GO TO 230
    220 WRITE(NW,200) K
    230 DO 240 L=K1,K2
        IF(ALT(L).LT.ALIM) GO TO 240
        KSET=L
    GC TC 260
    240 CONTINUE
    245 CONTINUE
    250 FCRMAT(37HMAX ALI. NOT FOUND CN TEMP DATA CARDS)
        WRITE(NW,250)
    260 CALL INTERF(ALT,STORE,KSET,2,1)
    261 FORMAT(I5,SE10.2)
        WRITE(NW,261) (J,(CONC(J,I),I=1,IGAS), T(J),J = 1,161)
        IGAS=IGAS-1
        DO 331 I=1,IGAS
            READ(NR,310)TYPE,NLEV ,NBAK ,AMAS ,CEXT,DEXB
            FCRMAT (A8,22X,I2,3X,I2,3X,F4.2,4X,2(E12.4,3X))
            DO 311 J=1,25
                IF(TYPE.EQ.CATATY(J)) GO TC 312
            CONTINUE
            WRITE(NW,370)
            370 FCRMAT(36H THE GAS IS NCT LISTED IN THE PROGRAM)
            STCP 1
        C 312 NSP=J
            NLEVEL(NSP)=NLEV
            NBAND(NSP)=NBAK
            AMASS(NSP)=AMAS
            CCEXT(NSP)=CEXT

```

```

EGND 730
EGND 740
EGND 750
EGND 760
EGND 770
EGND 780
EGND 790
EGND 800
EGND 810
EGND 820
EGND 830
EGND 840
EGND 850
EGND 860
EGND 870
EGND 880
EGND 890
EGND 900
EGND 910
EGND 920
EGND 930
EGND 940
EGND 950
EGND 960
EGND 970
EGND 980
EGND 990
EGND1000
EGND1010
EGND1020
EGND1030
EGND1040
EGND1050
EGND1060
EGND1070
EGND1080

```



```

CCEXB(NSP)=CXXE
IF(NSP.EG.2)REAC(NR,320) (C02(I,1),C02(I,2),I=1,16)
320 FCRMAT (6(F5.2,F8.2))
IF(NSP.EG.3)REAC(NR,320) (MATVA(I,1),MATVA(I,2),I=1,12)
ILEVL=0
DO 313 IJ=1,NLEV
ILEVL=ILEVL+1
READ(NR,330)LSC(NSP,IJ),LCC(NSP,IJ),RCM(NSP,IJ),BV(NSP,IJ),JGAS,
1JLEVEL
330 FCRMAT (2I6,2E12.4,36X,2I3)
IF(NSP.EG.JGAS) GO TO 314
WRITE(NH,340) NSP
340 FCRMAT(64HTPE CCDE ON THE DATA DOES N E WITH THE FIRST CARC
1WHICH IS,I3)
C STOP 2
314 IF(ILEVL.EG.JLEVEL) GO TO 313
WRITE(NH,350) NSP
350 FCRMAT(42HTHERE IS A CARD CUT OF SEQUENCE IN GAS AC,I3)
C STOP 3
313 CONTINUE
ILEVL=0
CO 316 IJ=1,NBAN
ILEVL=ILEVL+1
READ(NR,360)LBC(NSP,IJ),LEU(NSP,IJ),STR(NSP,IJ),SFLUX(I,J,NSP),
1(TFLUX(IJ,NSP,J),J=1,3),JGAS,JLEVEL
360 FCRMAT (2I6,5E12.4,2I3)
IF(NSP.EG.JGAS) GO TO 322
WRITE(NH,340) NSP
C STOP 4
322 IF(ILEVL.EG.JLEVEL) GO TO 316
WRITE(NH,350) NSP
C STOP 5
316 CONTINUE
321 CCNTINUE
DO 900 NLTE = 3, 4
DO 850 I = 2, 7

```

```

EGND1090
EGND1100
EGND1110
EGND1120
EGND1130
EGND1140
EGND1150
EGND1160
EGND1170
EGND1180
EGND1190
EGND1200
EGND1210
EGND1220
EGND1230
EGND1240
EGND1250
EGND1260
EGND1270
EGND1280
EGND1290
EGND1300
EGND1310
EGND1320
EGND1330
EGND1340
EGND1350
EGND1360
EGND1370
EGND1380
EGND1390
EGND1400
EGND1410
EGND1420
EGND1430
EGND1440

```

```

IF (I,GT,3) FCCFE = 0.0
IF ((I,EG, 6) .AND. (NLTE,EG, 4)) GC TC 850
IF ((I,EG, 7) .AND. (NLTE,EG, 3)) GC TC 850
CALL BANRAC (NEVEL(I),NEANC(I),IALT(I),CCEXT(I),CCEXB(I),
1 AMASS(I),LSC(I,1),LOC(I,1),LEC(I,1),LEU(I,1),EV(I,1),STR(I,1),
2 RCM'I,1),CCNC(I,1),SFLUX(I,1),TFLUX(I,1,I,1)
KK=IALT(I)
JSET=NEAND(I)
DO 840 J=1,JSET
700 FCRMAT(1,1,4,6,8,BAND NO.,13,5X,7,PLAMDA =,F6,2,7,HPICRC M,4X,4AE//)EGND1540
WRITE(NW,700)(XID(I,K),K=1,4),J,LAPCA(J),(XIDT(K),K=1,4)EGND1540
IF (NLTE,GT, 1) GO TO 707EGND1550
704 FCRMAT(40H ASSUMES LOCAL THERMODYNAMIC EQUILIBRIUM//)EGND1560
705 WRITE(NW,704)EGND1570
GC TC 715EGND1580
706 FCRMAT(40H LACK OF LOCAL THERMODYNAMIC EQUILIBRIUM//)EGND1590
707 WRITE (NW,706)EGND1600
710 FCRMAT(118H EXOATMOSPHERIC VIEWING $$$ - - - - - EN CEGND1610
1 C A T Y C S F E R I C V I E W I N G - - - - - )EGND1620
715 WRITE (NW,710)EGND1630
720 FCRMAT(118H TAN HT LIME BAND RAD $$$ ALTD - - - - - EAND RACIEGND1640
1 ANCE AS A FUNCTION OF ZENITH ANGLE (WATTS/CM2-STER) - - - - - )EGND1650
WRITE (NW,720)EGND1660
730 FCRMAT(113H (KM) (WATTS/CM2-STER) $$$ (KM) C CEG - LP 20 LEEGND1670
1 40 DEG 60 DEG 70 DEG 80 DEG 90 DEG -PCRZ )EGND1680
WRITE (NW,730)EGND1690
740 FCRMAT(F5.0,E14.4,F12.0,1X7E12.4)EGND1700
KSET=IALT(I)-1EGND1710
DO 830 K=1,KSET,5EGND1720
IF (K,GT,51) GC TC 750EGND1730
HT=K+FSEGND1740
GC TC 760EGND1750
750 HT=150+(K-51)*5EGND1760
760 C1=HT+6371.0EGND1770
DO 810 L=1,7EGND1780
M1=K+1EGND1790
EGND1800

```

EGND1810
EGND1820
EGND1830
EGND1840
EGND1850
EGND1860
EGND1870
EGND1880
EGND1890
EGND1900
EGND1910
EGND1920
EGND1930
EGND1940
EGND1950
EGND1960
EGND1970
EGND1980
EGND1990
EGND2000
EGND2010
EGND2020
EGND2030
EGND2040
EGND2050
EGND2060
EGND2070
EGND2080
EGND2090
EGND2100
EGND2110
EGND2120
EGND2130
EGND2140
EGND2150
EGND2160

```

PHI=ZANG(L)*.017453292
SAVTAU=0.0
STAU(L)=0.0
SRAD(L)=0.0
RANG=0.0
MSET=IAL1(I)
DO 790 M=M1,MSET
IF (M.GT.51) GO TO 770
D2=M+59+6371
GO TO 780
770 D2=150+(M-51)*5+6371
780 ANG=ASIN(SIN(PHI)*D1/D2)
DS=SQRT(D1*D1+D2*D2-2.*D1*D2*CCS(PHI-ANG))
DELS=(DS-RANG)*1.0E+05
RANG=DS
STAU(L)=STAL(L)+DELS*(0.5*(TAU(M,J)+TAU(M-1,J)))
SRAD(L)=SRAD(L)+DELS*0.5*(RAD(M,J)+SM1(MOD(LBU(I,J),10),STAU(L))
1+RAD(M-1,J)+SM1(MOD(LBU(I,J),10),SAVTAU))
SAVTAU=STAL(L)
790 CONTINUE
810 CONTINUE
SAVTAU=0.0
STAU(7)=0.0
VRANG = 0.0
SRAD(8)=0.0
MSET=IAL1(I)
DO 815 M=M,MSET
MSELS = 2*(MSET-K) + 1
L=MSET-M+K
IF (L.GT.51) GO TO 811
D2=L+59+6371
GO TO 812
811 D2=150+(L-51)*5+6371
812 DS=SQRT(D2*D2-D1*D1)
D3 = D2 - 6371.
IF (L.LT.MSET) GC TO 813

```

```

RANG=DS
SAVT1=1.0
GC TC 815
813 DELS=(RANG-DS)*1.0E+05
VRANG = VRANG + DELS
RANG=DS
STAU(7)=STAU(7)+DELS*(0.5*(TAU(L,J)+TAU(L+1,J)))
SAVT2=SM1(PCD(LEU(I,J),10),STAU(7))
SRAD(8)=SRAD(8)+DELS*(0.5*(RAD(L,J)*SAVT2+RAD(L+1,J)*SAVT1))
IF(K.GT.1) GC TC 819
819 SAVT1=SAVT2
815 CONTINUE
M1=K+1
RANG=0.0
MSET=IAL1(I)
DO 818 M=M1,MSET
IF(M.GT.91) GO TO 816
D2=M+59+6371
GO TO 817
816 D2=150+(M-91)*5+6371
817 DS=SQRT(D2*D2-01*01)
D3 = D2 - 6371.
DELS=(DS-RANG)*1.0E+05
VRANG = VRANG + DELS
RANG=DS
STAU(7)=STAU(7)+DELS*(0.5*(TAU(M,J)+TAU(M+1,J)))
SAVT2=SM1(MOD(LEU(I,J),10),STAU(7))
SRAD(8)=SRAD(8)+DELS*(0.5*(RAD(M,J)*SAVT2+RAD(M+1,J)*SAVT1))
IF(K.GT.1) GO TC 820
820 SAVT1=SAVT2
818 CONTINUE
C
WRITE(NP,4444) HT,SRAD(8),SRAD(1),SRAD(7)
4444 FCRMAT (F6.1,3E17.4)
830 CONTINUE
840 CONTINUE

```

```

EEND2170
EEND2180
EEND2190
EEND2200
EEND2210
EEND2220
EEND2230
EEND2240
EEND2250
EEND2260
EEND2270
EEND2280
EEND2290
EEND2300
EEND2310
EEND2320
EEND2330
EEND2340
EEND2350
EEND2360
EEND2370
EEND2380
EEND2390
EEND2400
EEND2410
EEND2420
EEND2430
EEND2440
EEND2450
EEND2460
EEND2470
EEND2480
EEND2490
EEND2500
EEND2510
EEND2520

```

85A CONTINUE
900 CCNTINUE
REWIND 1
REWIND 2
STOP 77
END

EGND2530
EGND2540
EGND2550
EGND2560
EGND2570
EGND2580

```

SUBROUTINE INTERP(ALT,STORE,KSET,MCDE,I)
  DIMENSION ALT(30),STORE(30)
  COMMON CCNC(161,10),TAU(161,10),RAD(161,10),T(161),LAMBDA(25),
  1TCIDEN(161)
  J=1
  A=60.
  DO 120 K=1,KSET
    10 IF(A-ALT(K))60,20,20
    20 GO TC (30,40),MCDE
    30 CONC(J,I)=STORE(K)
    GC TC 50
    40 T(J)=STORE(K)
    50 IF(K.LT.KSET) GO TO 60
    RETURN
    60 IF(J.GE.91) GO TO 70
    A=A+1.
    GO TO 110
    70 A=A+5.
    GC TO 110
    80 FR=(A-ALT(K-1))/(ALT(K)-ALT(K-1))
    GC TC (90,100),MCDE
    90 CONC(J,I)=STORE(K-1)+FR*(STORE(K)-STORE(K-1))
    91 J=J+1
    IF(J.GT.91) GO TO 92
    A=A+1.
    GO TO 10
    92 A=A+5.
    GC TO 10
    100 T(J)=STORE(K-1)+FR*(STORE(K)-STORE(K-1))
    GO TO 91
    110 J=J+1
    120 CONTINUE
    RETURN
  END

```

116P 10
 116P 20
 116P 30
 116P 40
 116P 50
 116P 60
 116P 70
 116P 80
 116P 90
 116P 100
 116P 110
 116P 120
 116P 130
 116P 140
 116P 150
 116P 160
 116P 170
 116P 180
 116P 190
 116P 200
 116P 210
 116P 220
 116P 230
 116P 240
 116P 250
 116P 260
 116P 270
 116P 280
 116P 290
 116P 300
 116P 310
 116P 320
 116P 330
 116P 340

```

SUBROUTINE EANGRAD (NLEVEL,NEAND,NALT,TEXCC,VEXCC,FM,LSC,LCC,
1 LBC,LBL,EV,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(10),POWER(10),KE(10),KL(10)
DIMENSION EAHALF(10),TFLUX(10),SFLUX(10)
DIMENSION GUP(10),GDOWN(10),GFAC(10)
DIMENSION TR2(161),TTH2(161),TDCAN2(161),TTUP2(161)
DIMENSION TTH(161,10),TTUP(161,10),TTDCWN(161,10)
DIMENSION CONC(161)
DIMENSION TVSIG(161),VTSIG(161)
DIMENSION WVUSIG(161),VVSIG(161)
DIMENSION VNSIG(161),VVSIG(161)
DIMENSION WAVE(250),LSC(250),LCC(250),LBC(250),LEU(250),STR(250)
DIMENSION GNDSTE(161)
DIMENSION C2(18,2),WATVA(18,2)
DIMENSION COEXT(25),COEXE(25)
DIMENSION CHEM(161)
COMMON /VERCOM/ I,CO2,WATVA,COEXT,COEXE
COMMON /FCFCGM/ VIBPOP(161,10)
COMMON /KCEP/KCHEM
COMMON /INOLT/NR,NP,NH
EQUIVALENCE (TDCWN2(1),TTDCWN(1,9)),(TTUP2(1),TTUP(1,9))
EQUIVALENCE (GNDSTE(1),VIBPOP(1,1))
COMMON /TECCOM/ NLTE
COMMON ZDEN(161,10),BNDTAU(161,10),ENDRAD(161,10),TEMP(161)
1 ,BNOLAM(25),DENTOT(161)
AAA = 0.0
IF (NLTE .EQ. 4) AAA = 2.0
IF (KCHEM .EQ. 1) READ(NR,50) CHEML
50 FORMAT (6E12.4)
IF (KCHEM .EQ. 1) KCHEP = 0
DO 130 NZ = 1, NALT
SUM = 0.0
TMP = TEMP(NZ)

```

```

ERAD 10
ERAD 20
ERAD 30
ERAD 40
ERAD 50
ERAD 60
ERAD 70
ERAD 80
ERAD 90
ERAD 100
ERAD 110
ERAD 120
ERAD 130
ERAD 140
ERAD 150
ERAD 160
ERAD 170
ERAD 180
ERAD 190
ERAD 200
ERAD 210
ERAD 220
ERAD 230
ERAD 240
ERAD 250
ERAD 260
ERAD 270
ERAD 280
ERAD 290
ERAD 300
ERAD 310
ERAD 320
ERAD 330
ERAD 340
ERAD 350
ERAD 360

```

```

DO 110 NL = 2, NLEVEL
  JL = (NL - 1)*25 + 1
  XVA = PCE(LSC(JL)/10,1)
  AA = EXP(-1.43826*WAVE(JL)/TEMP)*XVA
  VIBFCF(NZ,NL) = AA
  SUM = SUM + AA
  CCNTINUE
  SUM = SUM + 1.0
  GNDSTE(NZ) = CONC(NZ)/SUM
  CC 120 NL = 2, NLEVEL
  VIBFCF(NZ,NL) = CONC(NZ)*VIBFCF(NZ,NL)/SUM
  CCNTINUE
  DO 135 NL = 2, NLEVEL
    JL = (NL - 1)*25 + 1
    IF (LCC(JL).EG. 1) VVWAVE = 2331. - WAVE(JL)
    IF (LCC(JL).EG. 2) VVWAVE = 1556.4 - WAVE(JL)
    FAC = 2331.
    IF (LCC(JL).EG. 2) FAC = 1556.4
    CCNTINUE
    IF (NLTE.EG.1) GO TO 170
    DC 160 NZ = 1,NALT
    IF (I.EG. 3) GO TO 136
    IF (I.NE. 2) GO TO 140
    VVSIG(NZ) = 10.0*(EVAL(CC2(1,1),CC2(1,2),TEMP(NZ))*(-1.0/3.0),E)
    VVRSIG(NZ) = 10.0*(EVAL(CC2(7,1),CC2(7,2),TEMP(NZ),E)
    VVNSIG(NZ) = VVRSIG(NZ)/EXP(1.43886/TEMP(NZ)*18.0)
    VVDSIG(NZ) = 10.0*(EVAL(CC2(11,1),CC2(11,2),TEMP(NZ))*(-1.0/3.0),E)
    1 6)*CENTCF(NZ)
    VVUSIG(NZ) = VVDSIG(NZ)*VIBFCF(NZ,8)/VIBPCF(NZ,5)
    GO TO 150
  136 CCNTINUE
    VVNSIG(NZ) = 10.0*(EVAL(WATVA(7,1),WATVA(7,2),TEMP(NZ)
    1 *(-1.0/3.0),E)
    VVRSIG(NZ) = VVNSIG(NZ)*EXP(-1.43879/TEMP(NZ)*736.0)
    VVSIG(NZ) = 3.0E-14

```

```

ERAD 370
ERAD 380
ERAD 390
ERAD 400
ERAD 410
ERAD 420
ERAD 430
ERAD 440
ERAD 450
ERAD 460
ERAD 470
ERAD 480
ERAD 490
ERAD 500
ERAD 510
ERAD 520
ERAD 530
ERAD 540
ERAD 550
ERAD 560
ERAD 570
ERAD 580
ERAD 590
ERAD 600
ERAD 610
ERAD 620
ERAD 630
ERAD 640
ERAD 650
ERAD 660
ERAD 670
ERAD 680
ERAD 690
ERAD 700
ERAD 710
ERAD 720

```



```

GC TC 15"
CONTINUE
140 V1SIG(NZ) = CDEXT(I)
    VVNSIG(NZ) = CDEXB(I)
    VVRSIG(NZ) = VVNSIG(NZ)*EXP(-1.43886/TEMP(NZ)*WAVE)
150 CCNTINLF
    TVSIG(NZ) = V1SIG(NZ)/GNCSTE(NZ)*VIEPOP(NZ,2)
    V1SIG(NZ) = V1SIG(NZ)*DENTCT(NZ)
    TVSIG(NZ) = TVSIG(NZ)*DENTCT(NZ)
    TANNV = TEMP(NZ)
    IF (NLTE .LT. 5) GO TO 154
    IF (NZ .GT. 41 .AND. NZ .LT. 51) TANNV = TEMP(NZ)+FLCAT(NZ-41)
    1 *280.0
    IF (NZ .GE. 51) TANNV = 3009.0
154 CONTINUE
    FACTR = EXP(-1.43879*FAC/TANNV)
    VVNSIG(NZ) = VVNSIG(NZ)*DENTCT(NZ)*FACTOR/(1.0 + FACTR)
    VVRSIG(NZ) = VVRSIG(NZ)*DENTCT(NZ)/(1.0 + FACTR)
160 CONTINUE
170 CONTINUE
    WRITE (NF,36)
    DO 175 IX = 1, NALT
    WRITE (NF,16) (VIBPOP(IX,JX), JX = 1, NLEVEL)
175 CCNTINUE
    CO 180 NE = 1, NRAND
    NEX = (NE - 1)*25 + 1
    NEC = LEC(NEX)
    NI = MOC(NEC/100,100)
    NF = MOC(NEC,100)
    JI = (NI - 1)*25 + 1
    JF = (NF - 1)*25 + 1
    KL(NB) = MOC(LEU(NBX)/10,10)
    KE(NB) = MOC(LEU(NBX),10)
    BADLAN (NE) = 1.0E+4/(WAVE*JI) - WAVE(JF)
    XYP = MOC(LSC(JF)/10,10)
    XYM = MOC(LSC(JI)/10,10)

```

```

ERAD 730
ERAD 740
ERAD 750
ERAD 760
ERAD 770
ERAD 780
ERAD 790
ERAD 800
ERAD 810
ERAD 820
ERAD 830
ERAD 840
ERAD 850
ERAD 860
ERAD 870
ERAD 880
ERAD 890
ERAD 900
ERAD 910
ERAD 920
ERAD 930
ERAD 940
ERAD 950
ERAD 960
ERAD 970
ERAD 980
ERAD 990
ERAD 1000
ERAD 1010
ERAD 1020
ERAD 1030
ERAD 1040
ERAD 1050
ERAD 1060
ERAD 1070
ERAD 1080

```

EFAD1090
 EFAD1100
 EFAD1110
 EFAD1120
 EFAD1130
 EFAD1140
 EFAD1150
 EFAD1160
 EFAD1170
 EFAD1180
 EFAD1190
 EFAD1200
 EFAD1210
 EFAD1220
 EFAD1230
 EFAD1240
 EFAD1250
 EFAD1260
 EFAD1270
 EFAD1280
 EFAD1290
 EFAD1300
 EFAD1310
 EFAD1320
 EFAD1330
 EFAD1340
 EFAD1350
 EFAD1360
 EFAD1370
 EFAD1380
 EFAD1390
 EFAD1400
 EFAD1410
 EFAD1420
 EFAD1430
 EFAD1440

```

1  EA(NB) = STR(NEX)*2.804/ENDLAP(NB)**2*XYM
   *EXP(1.43679*WAVE(JF)/300.0)/XYMH
   FCHER(NB) = 1.58065E-20/ENDLAP(NB)
   GFAC(NB) = 3.72E-24*STR(NEX)*ENDLAP(NB)**2
1  *EXP(1.43679*WAVE(JF)/300.0)
   GUP(NB) = GFAC(NB)*TFLUX(NB)*6.2432
   IF (NLTE .LT. 3) GUP(NB) = 0.0
   GDOWN(NB) = GFAC(NB)*SFLUX(NB)
   IF (NLTE .LT. 4) GDOWN(NB) = 0.0
   KEN = KE(NB)
   KLN = KL(NB)
   BNDL = ENCLAM(NB)
   FAC = PCHER(NB)*EA(NB)
   SNB = STR(NEX)*EXP(1.43679*WAVE(JF)/300.0)
   DO 180 NZ = 1, NALT
     ENDTAU(NZ,NB) = TAUMAX(KLN,KEN,RA,EV,TEMP(NZ),ENDL,SNB,
1  VIBPOP(NZ,NF))
     IF (NLTE .GT. 1) GO TO 180
     BNDRAD(NZ,NB) = VIBPOP(NZ,NI)*FAC
180  CONTINUE
     IF (NLTE .EQ. 1) GO TO 900
     ITER = 0
181  ITER = ITER + 1
     NMIN = NALT - 1
     DO 190 NE = 1, NBAND
       TTDOWN(NALT,NB) = 0.0
       TTUF(1,NB) = 0.0
       TTH(1,NE) = SIMP(BNCTAU(1,NB),1.0E+5,1)
     DO 182 NZ = 2, NMIN
       TTH(NZ,NB) = SIMP(ENDTAU(NZ - 1,NB),1.0E+5,2)
182  CONTINUE
     CC 184 NZ = 2, NMIN
     NY = NALT - NZ + 1
     TTUP(NZ,NB) = TTUP(NZ - 1,NB) + TTH(NZ - 1,NB)
     TTDOWN(NY,NB) = TTDOWN(NY + 1,NB) + TTH(NY,NB)
184  CONTINUE
  
```

ERAD1450
ERAD1460
ERAD1470
ERAD1480
ERAD1490
ERAD1500
ERAD1510
ERAD1520
ERAD1530
ERAD1540
ERAD1550
ERAD1560
ERAD1570
ERAD1580
ERAD1590
ERAD1600
ERAD1610
ERAD1620
ERAD1630
ERAD1640
ERAD1650
ERAD1660
ERAD1670
ERAD1680
ERAD1690
ERAD1700
ERAD1710
ERAD1720
ERAD1730
ERAD1740
ERAD1750
ERAD1760
ERAD1770
ERAD1780
ERAD1790
ERAD1800

```

      TTDCN(1,NE) = VTDOWN(2,NE) + TTF(1,NE)
      TTUP(NALT,AB) = TTUP(NMIN,NE) + TTF(NMIN,NE)
      IF (ALTE.GT. 3) GO TO 190
      CC 186 NZ = 1, NALT
      VTDOWN(NZ,AB) = 0.0
186  CONTINUE
      IF (ALTE.GT. 2) GO TO 190
      DO 188 NZ = 1, NALT
      TTUP(NZ,NE) = 0.0
188  CONTINUE
190  CONTINUE
200  CONTINUE
      GC TC (210,220,230,240,250,230,230), I
210  CCNTINUE
      DC 215 NZ = 1, NALT
      VIBPOP(NZ,3) = GNDSTE(NZ)*(GUP(2)*SLM1(KB(2),TTUP(NZ,2))
1  + GDOWN(2)*SM1(KB(2),TTDCN(NZ,2)))/(EA(2) + EA(4))
      VIBPOP(NZ,2) = (EA(4)*VIBFCF(NZ,3) + GNDSTE(NZ)
1  *(GUP(1)*SLM1(KB(1),TTUP(NZ,1))
2  + GDOWN(1)*SM1(KB(1),TTDCN(NZ,1))
3  + VVSIG(NZ) + 0.75*VVNSIG(NZ)))
4  /(EA(1) + VVSIG(NZ) + 0.75*VVNSIG(NZ))
      VIBPOP(NZ,4) = GNDSTE(NZ)*(0.25*VVNSIG(NZ)
1  + GUP(3)*SLM1(KB(3),TTUP(NZ,3))
2  + GDOWN(3)*SM1(KB(3),TTDCN(NZ,3))
3  /(EA(3) + 0.25*VVRSIG(NZ))
215  CCNTINUE
      GC TO 350
220  CONTINUE
      NZ = 0
221  NZ = NZ + 1
      VIBPOP(NZ,8) = GNDSTE(NZ)*(VVNSIG(NZ) + GDOWN(9)*SM1(KB(9),
1  TDCN2(NZ)) + GUP(9)*SLM1(KB(9),TTUP2(NZ)))
2  /(VVRSIG(NZ) + EA(9) + VVDSIG(NZ))
      VVS = VVSIG(NZ)
      VTS = VTSIG(NZ)

```

```

FCP3 = VIEPOP(NZ,3)
POP4 = VIEPOP(NZ,4)
FCP7 = VIEPCP(NZ,7)
PCP2 = ((TVS + GUP(1)*SLM1(KE(1),TTUP(NZ,1))
8 + 6.2832*GFAC(1)*(XFER(VIEFCF(1,2),TTH(1,1),NZ,1,1,KE(1))
9 + XFER(VIBPOP(NZ,2),TTH(NZ,1),NALT,NZ,2,KE(1)))*EA(1)
1 + GDOWN(1)*SM1(KB(1),TTDOWN(NZ,1))*GNDSTE(NZ)
2 + FCP3*(EA(2) + VTS) + POP4*(EA(3) + VTS)
3 + POP7*(EA(7) + VTS))/(VTS + EA(1) + VTS)
VIBPOP(NZ,2) = POP2
POP8 = VIEPCP(NZ,8)
POP7 = ((TVS*0.25 + GUP(7)*SLM1(KE(7),TTUP(NZ,7))
1 + GDOWN(7)*SM1(KB(7),TTDOWN(NZ,7))*PCP2
2 + EA(8)*FCP8)/(EA(7) + VTS)
VIBPOP(NZ,7) = POP7
PCP5 = VIEPOP(NZ,5)
POP6 = VIEPOP(NZ,6)
PCP3 = (PCF2*(0.25*TVS + GUP(2)*SLM1(KE(2),TTUP(NZ,2))
1 + GDOWN(2)*SM1(KB(2),TTDOWN(NZ,2)) + POP5*(EA(4) + VTS))
2 /(EA(2) + VTS + VTS)
VIBPOP(NZ,3) = POP3
POP4 = (PCP2*(0.5*TVS + GUP(3)*SLM1(KB(3),TTUP(NZ,3))
1 + GDOWN(3)*SM1(KB(3),TTDOWN(NZ,3)) + POP5*EA(5)
2 + POP6*(EA(6) + VTS))/(EA(3) + VTS + VTS)
VIBFCF(NZ,4) = POP4
FCP5 = (PCP3*(TVS + GUP(4)*SLM1(KE(4),TTUP(NZ,4))
1 + GDOWN(4)*SM1(KB(4),TTDOWN(NZ,4))
2 + PCP4*(GUP(5)*SLM1(KB(5),TTUP(NZ,5))
3 + GDOWN(5)*SM1(KB(5),TTDOWN(NZ,5)) + POP8*VDSIG(NZ))
4 /(EA(4) + EA(5) + VTS + VVLSIG(NZ))
VIBPCF(NZ,5) = POP5
FCP6 = (FCP4*(0.5*TVS + GUP(6)*SLM1(KB(6),TTUP(NZ,6))
1 + GDOWN(6)*SM1(KB(6),TTDOWN(NZ,6))/(EA(6) + VTS)
VIBPOP(NZ,6) = POP6
GNDSTE(NZ) = GCNC(NZ) - (POP8 + POP6 + POP5 + FCP7 + FCP4
1 + POP3 + FCP2)

```

```

EFAD1819
EFAD1820
EFAD1830
EFAD1840
EFAD1854
EFAD1869
EFAD1870
EFAD1889
EFAD1899
EFAD1900
EFAD1910
EFAD1920
EFAD1930
EFAD1940
EFAD1950
EFAD1960
EFAD1970
EFAD1980
EFAD1990
EFAD2000
EFAD2010
EFAD2020
EFAD2030
EFAD2040
EFAD2050
EFAD2060
EFAD2070
EFAD2080
EFAD2090
EFAD2100
EFAD2110
EFAD2120
EFAD2130
EFAD2140
EFAD2150
EFAD2160

```

```

225 IF (NZ.LT.NALT) GO TO 221
    GC TC 350
230 CONTINUE
    IF (ITER.EC.2) GC TO 400
    EC 235 NZ = 1, NALT
    VIBPCF(NZ,2) = (TVSIG(NZ) + VVNSIG(NZ)
1 + GLF(1)*SLM1(KB(1),TTUF(NZ,1))
2 + 6.2832*GFAC(1)*(XFER(VIBPCF(1,2),TTH(1,1),NZ,1,1,KB(1))
3 + XFER(VIBPCF(NZ,2),TTH(NZ,1),NALT,NZ,2,KE(1))*EA(1)
4 + GDCWN(1)*SM1(KB(1),TTCCWN(NZ,1))*GNDSTE(NZ)
5 /(VTSIG(NZ) + EA(1) + VVRSIG(NZ))
235 CONTINUE
    GO TO 350
240 CONTINUE
    EC 245 NZ = 1, NALT
    VIBPCF(NZ,2) = ((TVSIG(NZ) + VVNSIG(NZ)
1 + GUP(1)*SLM1(KB(1),TTUF(NZ,1))
2 + 6.2832*GFAC(1)*(XFER(VIBPCF(1,2),TTH(1,1),NZ,1,1,KB(1))
3 + XFER(VIBPCF(NZ,2),TTH(NZ,1),NALT,NZ,2,KE(1))*EA(1)
4 + GDCWN(1)*SM1(KB(1),TTCCWN(NZ,1))*GNDSTE(NZ)
5 + (EA(3) + VTSIG(NZ))*VIBPCF(NZ,3) + AAA*CHEPL(NZ))
6 /(EA(1) + VTSIG(NZ) + VVRSIG(NZ) + TVSIG(NZ))
    VIBPCF(NZ,3) = ((TVSIG(NZ)
1 + GLF(3)*SLM1(KB(3),TTLP(NZ,3))
2 + GUP(2)*SLM1(KB(2),TTUF(NZ,2))
3 + GDCWN(2)*SLM1(KB(2),TTCCWN(NZ,2))
4 + GDCWN(3)*SLM1(KB(3),TTCCWN(NZ,3))*VIBPCF(NZ,2)
5 + AAA*CHEPL(NZ))/(EA(2) + EA(3) + TVSIG(NZ))
    GNDSTE(NZ) = CCNC(NZ) - VIBPCF(NZ,2) - VIBPCF(NZ,3)
245 CONTINUE
    GC TC 350
250 CONTINUE
    EC 255 NZ = 1, NALT
    VIBPCF(NZ,6) = GNDSTE(NZ)*(VVNSIG(NZ)
1 + GDCWN(3)*SM1(KB(3),TTCCWN(NZ,3))
2 + GUP(3)*SLM1(KB(3),TTUF(NZ,3))

```

```

EFAD2170
EFAD2180
EFAD2190
EFAD2200
EFAD2210
EFAD2220
EFAD2230
EFAD2240
EFAD2250
EFAD2260
EFAD2270
EFAD2280
EFAD2290
EFAD2300
EFAD2310
EFAD2320
EFAD2330
EFAD2340
EFAD2350
EFAD2360
EFAD2370
EFAD2380
EFAD2390
EFAD2400
EFAD2410
EFAD2420
EFAD2430
EFAD2440
EFAD2450
EFAD2460
EFAD2470
EFAD2480
EFAD2490
EFAD2500
EFAD2510
EFAD2520

```

```

3 / (VVRSIG(NZ)*1.5 + EA(3))
  VIBFCF(NZ,5) = (VIBPOP(NZ,6)*VRSIG(NZ)*0.5
1 + TVSIG(NZ)*VIBPOP(NZ,4))/(VTISIG(NZ) + 5.0*EA(1))
  VIBFCF(NZ,4) = (VIBPOP(NZ,5)*(VTISIG(NZ) + 5.0*EA(1))
1 + TVSIG(NZ)*VIBPOP(NZ,3))
2 / (TVSIG(NZ) + VTISIG(NZ) + 4.0*EA(1))
  VIBPCF(NZ,3) = (VIBPOP(NZ,4)*(VTISIG(NZ) + 4.0*EA(1))
1 + TVSIG(NZ)*VIBPOP(NZ,2))/(TVSIG(NZ) + VTISIG(NZ) + 3.0*EA(1))
  VIBPOP(NZ,7) = (VVNSIG(NZ)*VIBFCF(NZ,3)
1 + GNDSTE(NZ)*(GUP(2)*SLM1(KB(2),TTUP(NZ,2))
2 + GDOWN(2)*SM1(KB(2),TTOCH(NZ,2)))/EA(2)
  VIBPOP(NZ,2) = (GNDSTE(NZ)*(TVSIG(NZ)
1 + GUP(1)*SLM1(KB(1),TTUP(NZ,1))
2 + GDOWN(1)*SM1(KB(1),TTOCH(NZ,1))
3 + VIBPOP(NZ,3)*(3.0*EA(1) + VTISIG(NZ)))
4 / (VTISIG(NZ) + TVSIG(NZ) + EA(1))
255 CONTINUE
350 CONTINUE
DO 360 NE = 1, NBAND
  FAC = POWER(NB)*EA(NB)
  NBX = (NB - 1)*25 + 1
  NEC = LEC(NBX)
  NI = MOD(NBC/100,100)
  NF = MOD(NBC,100)
  KLN = KL(NB)
  KBN = KB(NB)
  BNDL = BNCLAM(NB)
  JF = (NF - 1)*25 + 1
  SNB = STR(NBX)*EXP(1.43879*H*AVE(JF)/300.0)
DO 360 NZ = 1, NALT
  BNDRAD(NZ,NB) = VIBPOP(NZ,NI)*FAC
  IF (NB.EC. 1) GO TO 360
  IF (NB.EC. 9) GO TO 360
  BNDAU(NZ,NB) = TAUMAX(KLN,KBN,NP,EV,TEMP(NZ),BNDL,SNB,
1 VIBPOP(NZ,NF))
360 CONTINUE

```

```

ERAD2530
ERAD2540
ERAD2550
ERAD2560
ERAD2570
ERAD2580
ERAD2590
ERAD2600
ERAD2610
ERAD2620
ERAD2630
ERAD2640
ERAD2650
ERAD2660
ERAD2670
ERAD2680
ERAD2690
ERAD2700
ERAD2710
ERAD2720
ERAD2730
ERAD2740
ERAD2750
ERAD2760
ERAD2770
ERAD2780
ERAD2790
ERAD2800
ERAD2810
ERAD2820
ERAD2830
ERAD2840
ERAD2850
ERAD2860
ERAD2870
ERAD2880

```

```

ERAD2890
ERAD2900
ERAD2910
ERAD2920
ERAD2930
ERAD2940
ERAD2950
ERAD2960
ERAD2970
ERAD2980
ERAD2990
ERAD3000
ERAD3010
ERAD3020
ERAD3030
ERAD3040

```

```

430 IF (ITER .LT.3) GO TO 181
500 CONTINUE
    WRITE (NF,36)
    DO 850 IX = 1, NALT
        WRITE (NF,16) (VIBPGF(IX,JX), JX = 1, NLEVEL)
850 CONTINUE
    WRITE (NP,36)
    IF (NLTG .GT. 2)
        1 WRITE(NP,8) ((VIBPGF(NZ,NL), NL = 1, 2), NZ = 1, NALT)
    RETURN
    2 FCRMAT (8E10.4)
    15 FCRMAT (1X)
    16 FCRMAT (3X,10E12.4)
    26 FCRMAT (6X,718)
    36 FCRMAT (1F1)
    END
C
C

```

```

FUNCTION XFER (RAD,TTH,NTCP,NECT,K,KB)
DIMENSION RAD(161),TTH(161),PCLD(161)
NINT = NTCP - NBOT
NA = NINT/2
KK = MOD(NINT,2)
NPL = NINT + 1
IF (NINT.GT. 0) GO TO 100
XFER = 0.0
RETURN
100 CONTINUE
IF (K.GT. 1) GO TO 200
IF (NINT.GT. 1) GO TO 110
XFER = (0.707107*RAD(2) + SDF2(KE,TTH(1))*RAD(1))*0.5E+5
RETURN
110 CONTINUE
SUM = 0.0
CC 120 NZ = 1, NINT
NY = NINT - NZ + 2
HCLD(NY) = SUM
SUM = SUM + TTH(NY - 1)
120 CONTINUE
HCLD(1) = SUM
DO 130 NZ = 1, NPL
HOLD(NZ) = RAD(NZ)*SDF2(KE,PCLD(NZ))
130 CONTINUE
SUM = 0.0
DO 140 N = 1, NA
SUM = SUM + SIPP(HOLD( 2* N - 1),1.0E+5,3)
140 CONTINUE
IF (KK.EQ. 1) SUM = SUM + SIPP(HCLD(NINT - 1),1.0E+5,2)
XFER = SUM
RETURN
200 CONTINUE
IF (NINT.GT. 1) GO TO 210
XFER = (0.707107*RAD(1) + SDF2(KE,TTH(1))*RAD(2))*0.5E+5
RETURN

```

```

XFER 10
XFER 20
XFER 30
XFER 40
XFER 50
XFER 60
XFER 70
XFER 80
XFER 90
XFER 100
XFER 110
XFER 120
XFER 130
XFER 140
XFER 150
XFER 160
XFER 170
XFER 180
XFER 190
XFER 200
XFER 210
XFER 220
XFER 230
XFER 240
XFER 250
XFER 260
XFER 270
XFER 280
XFER 290
XFER 300
XFER 310
XFER 320
XFER 330
XFER 340
XFER 350
XFER 360

```


XFER 370
 XFER 380
 XFER 390
 XFER 400
 XFER 410
 XFER 420
 XFER 430
 XFER 440
 XFER 450
 XFER 460
 XFER 470
 XFER 480
 XFER 490
 XFER 500
 XFER 510
 XFER 520
 XFER 530

```

210  CCNTINUE
      SUM = 0.0
      DC 220 NZ = 1, NPL
      HCLD(NZ) = SUM
      SUM = SUM + TTH(NZ)
220  CCNTINUE
      CC 230 NZ = 1, NPL
      HCLD(NZ) = RAD(NZ)*SDP2(KE,HCLD(NZ))
230  CCNTINUE
      SUM = 0.0
      CC 240 N = 1, NN
      SUM = SUM + SIPP(HCLD( 2* N - 1),1.0E+5,3)
240  CCNTINUE
      IF (KK.EC. 1) SUM = SUM + SIPP(HCLD(NINT - 1),1.0E+5,2)
      XFER = SUM
      RETURN
      END
  
```

```

FUNCTION EVAL (XP,YP,X,N)
DIMENSION XP(N),YP(N)
H = XP(2) - XP(1)
IF (X .GE. XP(N - 1)) GO TO 120
I = 1
110 CONTINUE
I = I + 1
IF (X .GE. XP(I + 1)) GO TO 110
GO TO 130
120 CONTINUE
I = N - 1
130 CONTINUE
XH = (X - XP(I))/H
C = YP(I)
B = 0.5*(YP(I + 1) - YP(I - 1))
A = 0.5*(YP(I - 1) - C - C + YP(I + 1))
EVAL = (A*XH + B)*XH + C
RETURN
END

```

```

EVAL 10
EVAL 20
EVAL 30
EVAL 40
EVAL 50
EVAL 60
EVAL 70
EVAL 80
EVAL 90
EVAL 100
EVAL 110
EVAL 120
EVAL 130
EVAL 140
EVAL 150
EVAL 160
EVAL 170
EVAL 180
EVAL 190

```

1MAX	17
TPAX	20
1MAX	30
TPAX	40
TPAX	50

```

FUNCTION TAUMAX (KL,KE,RH,BV,T,FLAM,S,DEN)
SIG = 1.43879*EV/T
TAUMAX = 5.3613E-18*SQRT(RH/T)*FLAM*DEN*S*SIG*MAX(KL,KE,SIG)
RETURN
END

```

SJPA 10
 SJPA 20
 SJPA 30
 SJPA 40
 SJPA 50
 SJPA 60
 SJPA 70
 SJPA 80
 SJPA 90
 SJPA 100
 SJPA 110
 SJPA 120
 SJPA 130
 SJPA 140
 SJPA 150
 SJPA 160
 SJPA 170
 SJPA 180
 SJPA 190
 SJPA 200
 SJPA 210

```

C      FUNCTION SJMAX (KL,KB,SIG)
C      SJMAX COMPUTES THE RELATIVE STRENGTH OF THE STRONGEST LINE
C      IF KB = 1, SIGMA-SIGMA TRANSITION
C      IF KB = 2, SIGMA-PI TRANSITION
      GO TO (110,120,130), KB
110    CCNTINUE
      TJM = 0.5*(SQRT(0.25 + 2.0/SIG) - 1.5)
      A = 0.5*(TJM + TJM + 1.0)
      GO TO 200
120    CCNTINUE
      TJM = 0.5*(SQRT(2.0/SIG) - 1.0)
      A = TJM + 1.0
      GO TO 200
130    CCNTINUE
      SJMAX = 0.0
      RETURN
200    CCNTINUE
      SJM = A*EXP(-SIG*TJM*(TJM + 1.0))
      SJMAX = SJM/GJPART(KL,SIG)
      RETURN
      END

```

```

1.0
FUNCTION SINT (A,B,C,H,K)
IF (K.GT.1) GO TO 110
SINT = (8.0*B + 5.0*A - C)*H/12.0
RETURN
CONTINUE
SINT = (8.0*B + 5.0*C - A)*H/12.0
RETURN
END

```

```

SINT
SINT
SINT
SINT
SINT
SINT
SINT
SINT
10
20
30
40
50
60
70
80

```

```

C      FUNCTION GJPART (K,SIG)
C      GJPART COMPUTES THE FOTATICAL PARTITION FUNCTION FOR A
C      RIGID ROTATOR, SIG = LE. 0.2.
C      K = 1, ALL LEVELS POPULATED
C      K = 2, EVEN LEVELS POPULATED
C      K = 3, ODD LEVELS POPULATED
C      IF (SIG .GT. 0.2) STOP
C      X = SIG
C      IF (K - 2) 110,120,130
C      COEFFICIENTS ARE FROM RUN DALLIE-EV
C      110 CONTINUE
C      GJPART = (((0.00386500232857*SIG + 0.012571835471)*X
C      + 0.0666756164530)*X + 0.332333133379)*X + 1.00000000017)/SIG
C      RETURN
C      120 CONTINUE
C      GJPART = (((((62.5879214103*SIG - 29.5697177693)*SIG
C      + 5.49692670096)*SIG - 0.502572235608)*SIG
C      + 0.0257378193546)*SIG + 0.0328350903506)*SIG
C      + 0.166670220770)*SIG + 0.4599999999456)/SIG
C      RETURN
C      130 CONTINUE
C      GJPART = (((((29.5703117876 - 62.58/4596502*SIG)*SIG
C      - 5.49575373669)*SIG + 0.5025750942701)*SIG
C      - 0.0170297551676)*SIG + 0.0328315889069)*SIG
C      + 3.166663112387)*SIG + 5.00000000544)/SIG
C      RETURN
C      END

```

SIMP 10
 SIMP 20
 SIMP 30
 SIMP 40
 SIMP 50
 SIMP 60
 SIMP 70
 SIMP 80
 SIMP 90
 SIMP 100
 SIMP 110
 SIMP 120
 SIMP 130

```

FUNCTION SIMP (A,H,K)
  DIMENSION A(3)
  IF (K - 2) 100,110,120
100 CONTINUE
  SIMP = (8.0*A(2) + 5.0*A(1) - A(3))*H/12.0
  RETURN
110 CONTINUE
  SIMP = (8.0*A(2) + 5.0*A(3) - A(1))*H/12.0
  RETURN
120 CONTINUE
  SIMP = (4.0*A(2) + A(1) + A(3))*H/3.0
  RETURN
END
  
```

10	SP1
20	SP1
30	SP1
40	SP1
50	SP1
60	SP1
70	SP1
80	SP1
90	SP1
100	SP1
110	SP1
120	SP1
130	SP1
140	SP1
150	SP1
160	SP1
170	SP1
180	SP1
190	SP1
200	SP1
210	SP1
220	SP1
230	SP1
240	SP1
250	SP1
260	SP1
270	SP1
280	SP1
290	SP1
300	SP1
310	SP1
320	SP1
330	SP1
340	SP1
350	SP1
360	SP1

```

FUNCTION SM1 (K,TAU)
T = ABS(TAU)
IF (T.EQ. 0.0) GO TO 300
IF (K.GE. 2) GO TO 200
IF (T.GT. 4.0) GO TO 110
X = (T - 2.0)*0.5
SM1 = (((12.3549E-03)*X - 8.90535E-03)*X + 2.56223E-02)*X
1 - 7.07877E-02)*X + 1.63735E-01)*X - 3.01749E-01)*X + 0.426845
RETURN
110 CONTINUE
IF (T.GT. 10.0) GO TO 120
X = (T - 7.0)/3.0
SM1 = (((15.2297E-04)*X - 1.79043E-03)*X + 4.05105E-03)*X
1 - 1.04745E-02)*X + 2.59109E-02)*X - 6.01708E-02)*X + 0.134123
RETURN
120 CONTINUE
IF (T.GT. 100.0) GO TO 130
Y = ALOG(T)
X = 9.210340371976/Y - 3.0
SM1 = (((((-6.24516E-04)*X - 6.79378E-04)*X + 2.98590E-03)*X
1 -1.28150E-03)*X - 6.31784E-03)*X + 3.78904E-02)*X
2 + 8.90441E-01)/T
RETURN
130 CONTINUE
Y = ALOG(T)
X = 9.210340371976/Y - 1.0
SM1 = (((16.12738E-04)*X - 2.93434E-04)*X - 4.36163E-04)*X
1 + 1.30027E-02)*X - 3.80088E-04)*X + 3.23441E-02)*X
2 + 8.17407E-01)/T
RETURN
200 CONTINUE
IF (T.GT. 4.0) GO TO 210
X = (T - 2.0)*0.5
SM1 = (((11.35071E-03)*X - 5.29904E-03)*X + 1.56245E-02)*X
1 - 4.74039E-02)*X + 1.26175E-01)*X - 2.86333E-01)*X + 0.517573
RETURN

```


210	CONTINUE	SP1	370
	IF (T.GT. 10.0) GO TO 220	SP1	380
	X = (T - 7.0)/3.0	SP1	390
	SM1 = ((13.76420E-04)*X - 1.25384E-03)*X + 3.41168E-03)*X	SP1	400
1	.. 1.05360E-02)*X + 3.07476E-02)*X - 8.14705E-02)*X + 0.1541308	SP1	410
	RETURN	SP1	420
220	CONTINUE	SP1	430
	IF (T.GT. 100.0) GO TO 230	SP1	440
	Y = ALOG(T)	SP1	450
	X = 9.210340371976/Y - 3.0	SP1	460
	SM1 = ((-7.53415E-03)*X - 8.87124E-03)*X + 6.62199E-02)*X	SP1	470
1	+ 1.304245)/T	SP1	480
	RETURN	SP1	490
230	CONTINUE	SP1	500
	Y = ALOG(T)	SP1	510
	X = 9.210340371976/Y - 1.0	SP1	520
	SM1 = ((11.99067E-03 - 5.82519E-04)*X) *X + 1.59669E-03)*X	SP1	530
1	+ 1.21174E-03)*X + 4.76165E-02)*X + 1.18445)/T	SP1	540
	RETURN	SP1	550
240	CONTINUE	SM1	560
	SM1 = 1.0	SM1	570
	RETURN	SM1	580
	END	SM1	590

SLM1 10
SLM1 20
SLM1 30
SLM1 40
SLM1 50
SLM1 60
SLM1 70
SLM1 80
SLM1 90
SLM1 100
SLM1 110
SLM1 120
SLM1 130
SLM1 140
SLM1 150
SLM1 160
SLM1 170
SLM1 180
SLM1 190
SLM1 200
SLM1 210
SLM1 220
SLM1 230
SLM1 240
SLM1 250
SLM1 260
SLM1 270
SLM1 280
SLM1 290
SLM1 300
SLM1 310
SLM1 320
SLM1 330
SLM1 340
SLM1 350
SLM1 360

```

FUNCTION SLM1 (K,TAU)
  T = ABS(TAU)
  IF (T.EC. 0.0) GO TO 300
  IF (K.GT. 1) GO TO 200
  IF (T.GT. 1.0) GO TO 110
  X = (T - 0.5)*2.0
  SLM1 = (((((2.30089E-03)*X - 2.29897E-03)*X - 2.19020E-03)*X
    + 8.432313E-04)*X + 4.56944E-03)*X - 1.03180E-02)*X
  1 + 3.28304E-02)*X - 1.15722E-01)*X + 3.34592E-01
  2 RETURN
110 CONTINUE
  IF (T.GT. 4.0) GO TO 120
  X = (T - 2.5)/1.5
  SLM1 = (((((9.76236E-04 - 3.52846E-04)*X - 1.79436E-03)*X
    + 4.96516E-03)*X - 1.33950E-02)*X + 3.19020E-02)*X
  1 - 6.75970E-02)*X + 1.23984E-01
  2 RETURN
120 CONTINUE
  IF (T.GT. 10.0) GO TO 130
  X = (T - 7.0)/3.0
  SLM1 = (((((3.17194E-04)*X - 0.05675E-04)*X + 1.48227E-03)*X
    - 3.59295E-03)*X + 8.66454E-03)*X - 1.97823E-02)*X + 0.0439928
  1 RETURN
130 CONTINUE
  IF (T.GT. 100.0) GO TO 140
  Y = ALOG(T)
  X = 9.210340371976/Y - 3.0
  SLM1 = (((((3.77202E-04)*X - 6.03531E-04)*X - 1.17101E-03)*X
    + 1.10506E-02)*X + 0.293204)/Y
  1 RETURN
140 CONTINUE
  Y = ALOG(T)
  X = 9.210340371976/Y - 1.0
  SLM1 = (((((9.45179E-05)*X + 2.68993E-04)*X - 5.45078E-04)*X
    + 1.01193E-02)*X + 0.272108)/Y
  1 RETURN

```

```

200  CONTINUE
    IF (T.GT. 1.0) GO TO 210
    X = (T - 0.5)*2.0
    SLM1 = (((1.53668E-03)*X - 1.53539E-03)*X - 1.46271E-03)*X
    1 + 5.61827E-04)*X + 3.06453E-03)*X - 7.08252E-03)*X
    2 + 2.42680E-02)*X - 9.93037E-02)*X + 3.65233E-01
    RETURN
210  CCNTINUE
    IF (T.GT. 4.0) GO TO 220
    X = (T - 2.5)/1.5
    SLM1 = (((6.60062E-04)*X - 1.73522E-03)*X + 3.65978E-03)*X
    1 - 1.05347E-02)*X + 2.95639E-02)*X - 7.50205E-02)*X + 0.164106
    RETURN
220  CONTINUE
    IF (T.GT. 10.0) GO TO 230
    X = (T - 7.0)/3.0
    SLM1 = (((2.55944E-04)*X - 7.18512E-04)*X + 1.57111E-03)*X
    1 - 4.31880E-03)*X + 1.14462E-02)*X - 2.80911E-02)*X + 0.0643049
    RETURN
230  CONTINUE
    IF (T.GT. 100.0) GO TO 240
    Y = ALOG(T)
    X = 9.210340371076/Y - 3.0
    SLM1 = (((-1.51727E-03)*X - 1.40803E-03)*X + 1.89331E-02)*X
    1 + 0.428568)/T
    RETURN
240  CONTINUE
    Y = ALOG(T)
    X = 9.21034037976/Y - 1.0
    SLM1 = (((4.59913E-04)*X + 2.46757E-04)*X - 1.12575E-04)*X
    1 + 1.48752E-02)*X + 0.394276)/T
    RETURN
300  CONTINUE
    SLM1 = 0.5
    RETURN
    END

```

SLM1 370
 SLM1 380
 SLM1 390
 SLM1 400
 SLM1 410
 SLM1 420
 SLM1 430
 SLM1 440
 SLM1 450
 SLM1 460
 SLM1 470
 SLM1 480
 SLM1 490
 SLM1 500
 SLM1 510
 SLM1 520
 SLM1 530
 SLM1 540
 SLM1 550
 SLM1 560
 SLM1 570
 SLM1 580
 SLM1 590
 SLM1 600
 SLM1 610
 SLM1 620
 SLM1 630
 SLM1 640
 SLM1 650
 SLM1 660
 SLM1 670
 SLM1 680
 SLM1 690
 SLM1 700
 SLM1 710
 SLM1 720

```

FUNCTION SGM2 (K,TAU)
T = ABS(TAU)
IF (T.EQ. 0.0) GO TO 300
IF (K.GT. 1) GO TO 200
IF (T.GT. 1.0) GO TO 110
X = (T - 0.5)*2.0
SDM2=(((((1.96434E-01)*X-1.37518E-01)*X-5.44871E-01)*X
1 +3.48950E-01)*X+5.84914E-01)*X-3.35128E-01)*X-2.57212E-01)*X
2 +1.44634E-01)*X+7.67977E-02)*X-3.36629E-02)*X+3.63636E-03)*X
3 -6.16125E-02)*X-2.74611E-01)*X+5.26563E-01+T*ALOG(T)/SQRT(3.)
RETURN
110 CONTINUE
IF (T.GT. 4.0) GO TO 120
X = (T - 2.5)/1.5
SDM2=(((((6.71928E-04)*X-1.34115E-03)*X+1.38025E-03)*X
1 -3.42908E-03)*X+5.33267E-03)*X-1.95584E-02)*X+3.76565E-02)*X
2 -5.98769E-02)*X+6.35365E-02
RETURN
120 CONTINUE
IF (T.GT. 10.0) GO TO 130
X = (T - 7.0)/3.0
SDM2=(((((6.91884E-05)*X-1.72263E-04)*X+2.7078E-04)*X
1 -6.27607E-04)*X+1.45592E-03)*X-2.50094E-03)*X+5.21475E-03)*X
2 -8.13190E-03)*X+9.28992E-03
RETURN
130 CONTINUE
IF (T.GT. 100.0) GO TO 140
Y = ALOG(T)
X = 9.210340371976/Y - 3.0
SDM2=((((-1.78966E-04)*X-1.12080E-03)*X+2.02868E-03)*X
1 +3.42340E-04)*X-5.16189E-03)*X+2.21993E-02)*X+4.28605E-01
SGM2 = SGM2/(T*T)
RETURN
140 CONTINUE
Y = ALOG(T)
X = 9.210340371976/Y - 1.0

```

```

SDM2=((3.3204E-04)*X+6.11261E-04)*X+5.85934E-04)*X+1.71696E-02)*X
2 +3.85175E-01
SCM2 = SDM2/(1+T)
RETURN
200 CONTINUE
IF (T.GT. 1.0) GO TO 210
X = (T - 0.5)/2.0
SCM2=((((((( (-5.54556E-01*X+3.82858E-01)*X+1.68014E+00)*X
1 -1.06923E+00)*X-2.00213E+00)*X+1.15274E+00)*X+1.17482E+00)*X
2 -5.89857E-01)*X-3.55692E-01)*X+1.51264E-01)*X+4.23668E-02)*X
3 +1.06679E-03)*X-7.54472E-02)*X-2.65402E-01)*X+5.61024E-01
4+1*ALOG(1)/SQRT(3.)
RETURN
210 CCNTINUE
IF (T.GT. 4.0) GO TO 220
X = (T - 2.5)/1.5
SDM2=(((((5.78273E-04*X-1.15507E-03)*X+1.19596E-03)*X
1 -3.00744E-03)*X+8.15919E-03)*X-1.84944E-02)*X+3.50840E-02)*X
2 -7.15164E-02)*X+9.08052E-02
RETURN
220 CCNTINUE
IF (T.GT. 10.0) GO TO 230
X = (T - 7.0)/3.0
SDM2=((((( (-1.58499E-04*X+3.58493E-04)*X-6.8423E-04)*X
1 +1.65527E-03)*X-3.91034E-03)*X+7.99235E-03)*X-1.38493E-02)*X
2 +1.70007E-02
RETURN
230 CONTINUE
IF (T.GT. 100.0) GO TO 240
Y = ALOG(T)
X = 5.210340371976/Y - 3.0
SDM2=(((((2.7684E-04*X-2.05836E-04)*X-8.55043E-03)*X-1.07621E-02)*X
1 +5.16025E-02)*X+2.12072E-01
2 +SDM2/(1+T)
RETURN
240 CCNTINUE

```

```

Y = ALOG(T)
X = 9.21034037976/Y - 1.0
SDM2=((((1.19945E-03*X-3.60978E-04)*X-1.48588E-05)*X
1+1.8104E-03)*X+3.40447E-03)*X+.26935E-02)*X+7.19045E-01
SDM2 = SDM2/(T*T)
RETURN
300 CONTINUE
SDM2=1./SQRT(2.)
RETURN
END

```

```

SLM2 730
SLM2 740
SLM2 750
SLM2 760
SLM2 770
SLM2 780
SLM2 790
SLM2 800
SLM2 810
SLM2 820

```

```

PROGRAM SFCTRA (XDATA,OLTPUT,TAPE1=XDATA,TAPE2=OLTPUT,TAPE4)
DIMENSION LSC(10),LCC(10),RCP(10),EV(10),LEC(10),LEL(10),STR(10)
DIMENSION P(100),Q(100),R(100),FS(100),QS(100),RS(100)
DIMENSION FHCLAM(450),HMCSTR(450),MHCLSE(450),HRAC(600)
DIMENSION ROTLAM(250),RCTEA(250),ROTLSE(250),ROTST(250)
DIMENSION AVLAM(201)
DIMENSION CCN(161,10),TEMP(161),FRCC(161)
DIMENSION AVTEMP(12,10),BVTEMP(12,10)
DIMENSION CCLCAT(12,10),FCRCNT(12,10)
DIMENSION GR(40)
DIMENSION GRH(40)
DIMENSION ENU(250),ENL(250),ENERGY(250)
DIMENSION WVLTH(450),EA(250)
DIMENSION TAU(250),RAD(250)
DIMENSION RADA( 50),RADE( 50),RACC( 50)
DIMENSION EALC(250),EALR(250)
DIMENSION SPECA(201,12),SPECR(201,12),SPECG(201,12)
DIMENSION EAMOL(10)
EQUIVALENCE (P(1),HEAD(1)),(Q(1),HRAC(101)),(R(1),HRAC(201)),
1 (PS(1),HRA(301)),(QS(1),HRAC(401)),(RS(1),HRA(501))
COMMON SPECA,SPECR,SPEC
COMMON /HPCOM/ HHOLAM,HMCSTR,MHCLSE,HRAC
COMMON /RCTCOM/ ROTLAM,ROTEA,ROTLSE,ROTST
REWIND 4
NCUT = 0
NLN = 100
NALT = 61
CALL ROTATE
DO 500 NCASES = 1, 2
WRITE (2,16)
BLAM = 5.0
CO 105 I = 1, 201
AVLAM(I) = BLAM
BLAM = BLAM + 0.1
CO 105 J = 1, 12
SPECA(I,J) = 0.0

```

```

SFTR 10
SFTR 20
SFTR 30
SFTR 40
SFTR 50
SFTR 60
SFTR 70
SFTR 80
SFTR 90
SFTR 100
SFTR 110
SFTR 120
SFTR 130
SFTR 140
SFTR 150
SFTR 160
SFTR 170
SFTR 180
SFTR 190
SFTR 200
SFTR 210
SFTR 220
SFTR 230
SFTR 240
SFTR 250
SFTR 260
SFTR 270
SFTR 280
SFTR 290
SFTR 300
SFTR 310
SFTR 320
SFTR 330
SFTR 340
SFTR 350
SFTR 360

```

```

SPECB(I,J) = 0.0
SPECC(I,J) = 0.0
105 CCNTINUE
DC 110 I = 1, NALT
IF (I .LT. 21) TEMP(I) = 255.0 - (I - 1)*3.7
IF (I .GT. 20) TEMP(I) = 181.0
IF (I .GT. 31) TEMP(I) = 121.0 + (I - 31)*2.5
IF (I .GT. 41) TEMP(I) = 210.0 + (I - 41)*4.7
IF (I .GT. 51) TEMP(I) = 257.0 + (I - 51)*9.2
110 CONTINUE
WRITE (2,8) (TEMP(I), I = 1, NALT)
WRITE (2,14)
READ (1,6) NSPTOT
DO 400 NSP = 1, NSPTOT
  READ (1,6) ISP
  READ (1,41) NLEVEL,NBAND,NM
  CO 210 I = 1, NLEVEL
  READ (1,41) LSC(I),LCC(I),RCM(I),EV(I)
210 CONTINUE
CO 220 I = 1, NEAND
  READ (1,41) LBC(I),LBU(I),STR(I)
220 CONTINUE
  READ (1,6) NALT,NRLEV
  REAC (1,42) ((CON(NZ,NL), NL = 1, 8), NZ = 1, NALT)
  C COMPUTE MEAN TEMPERATURES AND COLUMN COUNTS
  K = 12
DC 129 NL = 1, NLEVEL
DG 112 I = 1, 61
  PROD(I) = CON(I,NL)*TEMP(I)
112 CCNTINUE
  TF = 0.0
  TC = 0.0
  NA = 61
DO 120 I = 1, K
  TF = TF + SIMP(PROD(NA - 2),1.0E+5,2)
  TC = TC + SIMP(PROD(NA - 3),1.0E+5,3) + SIMP(PROD(NA - 5),1.0E+5,3)
  NA = NA + 1
120 CONTINUE

```

SFTR 370
 SFTR 380
 SFTR 390
 SFTR 400
 SFTR 410
 SFTR 420
 SFTR 430
 SFTR 440
 SFTR 450
 SFTR 460
 SFTR 470
 SFTR 480
 SFTR 490
 SFTR 500
 SFTR 510
 SFTR 520
 SFTR 530
 SFTR 540
 SFTR 550
 SFTR 560
 SFTR 570
 SFTR 580
 SFTR 590
 SFTR 600
 SFTR 610
 SFTR 620
 SFTR 630
 SFTR 640
 SFTR 650
 SFTR 660
 SFTR 670
 SFTR 680
 SFTR 690
 SFTR 700
 SFTR 710
 SFTR 720


```

1 TC = TC + SIMP(CCN(NA - 2,NL),1.0E+5,2)
  + SIMP(CCN(NA - 3,NL),1.0E+5,3) + SIMP(CCN(NA - 5,NL),1.0E+5,3)
  COLCNT(K - I + 1,NL) = TC
  AVTEMP(K - I + 1,NL) = TP/TC
  NA = NA - 5
  HCRCNT(I,NL) = 0.0
  CCNTINUE
120 WRITE (2,6) (COLCNT(I,NL), I = 1, K)
    WRITE (2,14)
    WRITE (2,6) (AVTEMP(I,NL), I = 1, K)
    WRITE (2,14)
    REASE = 6.431E+8
    DO 128 I = 1, K
      TPH = 0.0
      TCH = 0.0
      NA = 61
      RE2 = REASE*REASE
      RTOP = 6.431E+8 + NALT*1.0E+5
      SVH = SQRT(RTOP*RTOP - RE2)
      KK = K - I + 1
      DO 125 J = 1, KK
        RTOP = RTCP - 1.0E+5
        STH = SQRT(RTOP*RTOP - RE2)
        H5 = SVH - STH
        SVH = STH
        RTOP = RTCP - 1.0E+5
        STH = SQRT(RTOP*RTOP - RE2)
        H4 = SVH - STH
        SVH = STH
        RTOP = RTCP - 1.0E+5
        STH = SQRT(RTOP*RTOP - RE2)
        H3 = SVH - STH
        SVH = STH
        RTOP = RTCP - 1.0E+5
        STH = SQRT(RTOP*RTOP - RE2)
        H2 = SVH - STH

```

```

SFTR 730
SFTR 740
SFTR 750
SFTR 760
SFTR 770
SFTR 780
SFTR 790
SFTR 800
SFTR 810
SFTR 820
SFTR 830
SFTR 840
SFTR 850
SFTR 860
SFTR 870
SFTR 880
SFTR 890
SFTR 900
SFTR 910
SFTR 920
SFTR 930
SFTR 940
SFTR 950
SFTR 960
SFTR 970
SFTR 980
SFTR 990
SFTR1000
SFTR1010
SFTR1020
SFTR1030
SFTR1040
SFTR1050
SFTR1060
SFTR1070
SFTR1080

```

```

SVH = STH
RTCP = RTCF " 1.0E+5
STH = SCR1(ABS(RTOP*RTOP - RE2))
H1 = SVH - STH
SVH = STH
TPH = TPH + QINT(PROD(NA - 2),P4,H5,2)
1  + GINT(PROD(NA - 3),H3,P4,3) + GINT(PROD(NA - 5),P1,H2,3)
TCH = TCH + GINT(CON(NA - 2,NL),P4,H5,2)
1  + GINT(CON(NA - 3,NL),H3,H4,3) + GINT(CON(NA - 5,NL),H1,P2,3)
NA = NA - 5
125 CONTINUE
HORCNT(I,NL) = TCH
EVTEMP(I,NL) = TPH/TCH
RBASE = REASE + 5.0E+5
128 CCNTINUE
WRITE (2,8) (HORCNT(I,NL), I = 1, K)
WRITE (2,14)
WRITE (2,8) (EVTEMP(I,NL), I = 1, K)
WRITE (2,14)
129 CCNTINUE
DO 400 NE = 1, ABAND
DO 230 I = 1, NRLEV
READ (1,43) HT,RADC(I),RADA(I),RACH(I)
230 CONTINUE
NEC = LEC(NB)
NI = MOC(NBC/100,100)
NF = MOC(NEC,100)
CENTER = RCM(NI) - RCM(NF)
SNB = STR(NE)*EXP(1.43879*RCR(NF)/300.0)
KL = MOC(LBU(NE)/10,10)
KB = MOC(LBU(NE),10)
IF (ISP.EC. 3) GO TO 260
IF (ISP.EC. 6) GO TO 260
IF (ISP.EC. 7) GO TO 260
CALL RCTPCS (CENTER,BV(NI),BV(NF),F,O,R,1)
CC 250 I = 1, 12

```

SFTR1090
SFTR1100
SFTR1110
SFTR1120
SFTR1130
SFTR1140
SFTR1150
SFTR1160
SFTR1170
SFTR1180
SFTR1190
SFTR1200
SFTR1210
SFTR1220
SFTR1230
SFTR1240
SFTR1250
SFTR1260
SFTR1270
SFTR1280
SFTR1290
SFTR1300
SFTR1310
SFTR1320
SFTR1330
SFTR1340
SFTR1350
SFTR1360
SFTR1370
SFTR1380
SFTR1390
SFTR1400
SFTR1410
SFTR1420
SFTR1430
SFTR1440

```

1 CALL ROTSTR (RADA(I),RM,EV(NF),AVTEMP(I,NF),CCLCNT(I,NF),
  SNE,PS,GS,RS,KE,1,P,Q,R)
  CALL FILTER (SPECAC(1,I),PS,P,NLN)
  CALL FILTER (SPECAC(1,I),CS,G,NLN)
  CALL FILTER (SPECAC(1,I),RS,R,NLN)
  CALL RCISTR (RACB(I),RM,EV(NF),BVTEMP(I,NF),HORCNT(I,NF),
1 SNE,PS,GS,RS,KE,1,P,Q,R)
  CALL FILTER (SPECUB(1,I),PS,P,NLN)
  CALL FILTER (SPECB(1,I),CS,G,NLN)
  CALL FILTER (SPECB(1,I),RS,R,NLN)
  CALL ROTSTR (RADC(I),RM,EV(NF),BVTEMP(I,NF),2.0*HORCNT(I,NF),
1 SNE,PS,GS,RS,KE,1,P,Q,R)
  CALL FILTER (SPECOC(1,I),PS,P,NLN)
  CALL FILTER (SPECOC(1,I),CS,G,NLN)
  CALL FILTER (SPECOC(1,I),RS,R,NLN)
  CONTINUE
250 GC TC 260
  CONTINUE
  DC 270 I = 1, 12
  IF (ISP,NE,3) GO TO 265
  CALL WATSTR (RADA(I),RM,AVTEMP(I,NF),CCLCNT(I,NF),1)
  CALL FILTER (SPECAC(1,I),HRAD,RCILAM,250)
  CALL WATSTR (RADA(I),RM,AVTEMP(I,NF),CCLCNT(I,NF),2)
  CALL FILTER (SPECAC(1,I),HRAD,HHCLAM,434)
  CALL WATSTR (RADB(I),RM,BVTEMP(I,NF),HORCNT(I,NF),1)
  CALL FILTER (SPECB(1,I),HRAD,RCILAM,250)
  CALL WATSTR (RADB(I),RM,BVTEMP(I,NF),HORCNT(I,NF),2)
  CALL FILTER (SPECB(1,I),HRAD,HHCLAM,434)
  CALL WATSTR (RADC(I),RM,BVTEMP(I,NF),2.0*HORCNT(I,NF),1)
  CALL FILTER (SPECOC(1,I),HRAD,RCILAM,250)
  CALL WATSTR (RADC(I),RM,BVTEMP(I,NF),2.0*HORCNT(I,NF),2)
  CALL FILTER (SPECOC(1,I),HRAD,HHCLAM,434)
  GO TO 270
  CONTINUE
265 CALL WATSTR (RADA(I),RM,AVTEMP(I,NF),CCLCNT(I,NF),3)
  CALL FILTER (SPECAC(1,I),HRAD,PS,32)

```

```

SFTR1450
SFTR1460
SFTR1470
SFTR1480
SFTR1490
SFTR1500
SFTR1510
SFTR1520
SFTR1530
SFTR1540
SFTR1550
SFTR1560
SFTR1570
SFTR1580
SFTR1590
SFTR1600
SFTR1610
SFTR1620
SFTR1630
SFTR1640
SFTR1650
SFTR1660
SFTR1670
SFTR1680
SFTR1690
SFTR1700
SFTR1710
SFTR1720
SFTR1730
SFTR1740
SFTR1750
SFTR1760
SFTR1770
SFTR1780
SFTR1790
SFTR1800

```

```

CALL MATSTR (RAD8(I),RM,BVTEMP(I,NF),HCRGNT(I,NF),3)
CALL FILTER (SPECB(1,I),HRAC,FS,32)
CALL MATSTR (RADC(I),RM,BVTEMP(I,NF),2.0*HCRGNT(1,NF),3)
CALL FILTER (SPEC(1,I),HRAC,FS,32)

```

270
280
400

CONTINUE

CONTINUE

CONTINUE

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

WRITE (2,18)

WRITE (2,16)

SFIR1810
SFIR1820
SFIR1830
SFIR1840
SFIR1850
SFIR1860
SFIR1870
SFIR1880
SFIR1890
SFIR1900
SFIR1910
SFIR1920
SFIR1930
SFIR1940
SFIR1950
SFIR1960
SFIR1970
SFIR1980
SFIR1990
SFIR2000
SFIR2010
SFIR2020
SFIR2030
SFIR2040
SFIR2050
SFIR2060
SFIR2070
SFIR2080
SFIR2090
SFIR2100
SFIR2110
SFIR2120
SFIR2130
SFIR2140
SFIR2150
SFIR2160

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 1, 51)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 51, 101)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 101, 151)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 151, 201)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 1, 51)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 51, 101)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 101, 151)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 151, 201)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 1, 51)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 51, 101)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 101, 151)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 151, 201)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 1, 51)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 51, 101)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 101, 151)

(AVLAM(J), (SPECB(J,I), I = 1,12), J = 151, 201)

500

CONTINUE

```

4      FCRMAT (214, ERROR IN AECVE CASE)
6      FCRMAT (616)
8      FCRMAT (6E12.4)
10     FCRMAT (613,F12.6,3F9.2,F5.1,E12.3,F12.3)
12     FCRMAT (4F12.4,E12.4,F12.6)
14     FCRMAT ( )
16     FCRMAT ( , 1)
18     FCRMAT (14,F8.2,12E10.3)
41     FCRMAT (216,5E12.4)
42     FCRMAT (8E10.4)
43     FCRMAT (F6.1,3E17.4)
      END

```

```

SFIR 10
SFIR 20
SFIR 30
SFIR 40
SFIR 50
SFIR 60
SFIR 70
SFIR 80
SFIR 90
SFIR 100
SFIR 110
SFIR 120
SFIR 130
SFIR 140

```

```

SUBROUTINE HAISTR (RAD,RM,T,CEN,N)
DIMENSION HHOLAM(450),HHCSTR(450),HHOLSE(450),HRAC(600)
DIMENSION ROTLAM(250),RCTEA(250),RCTLSE(250),RCTST(250)
DIMENSION STRO3(32),ENLC3(32),NLC3(32)
DATA STRO3 / 0.160,0.306,0.476,0.857,1.242,1.693,1.925,1.693,
1 1.564,1.244,1.822,2.057,1.911,1.133,0.0337,0.0067,0.0040,0.0044,
2 0.0047,0.0051,0.0042,0.0038,0.0036,0.0096,0.0066,0.0060,0.0023,
3 0.0068,0.0064,0.0042,0.0038,0.0046 /
DATA ENLC3 / 940.0,600.0,680.0,560.0,460.0,370.0,280.0,360.0,
1 440.0,280.0,66.0,140.0,250.0,420.0,710.0,570.0,470.0,400.0,
2 360.0,330.0,320.0,320.0,330.0,360.0,400.0,460.0,520.0,570.0,
3 590.0,570.0,490.0,380.0 /
DATA NLC3 / 44,43,43,46,54,57,71,130,167,76,75,97,110,133,233,
1 157,119,129,117,103,97,102,90,86,97,97,95,97,96,90,78,51 /
COMMON /HCCCH/ HHOLAM,HHOSTR,HHOLSE,HRAC
COMMON /RCTCOM/ ROTLAM,RCTEA,RCTLSE,ROIST
TCFAC = (273.15/T)**1.5
GC TO (100,200,300), N
CONTINUE
RADFAC = 1.58065E-20*DEN*TCFAC
TAUFAC = 4.8815E-18*SQRT(RM/T)*TCFAC*DEN
DO 110 I = 1, 250
TAU = TAUFAC*ROIST(I)*EXP(-1.43879*RCTLSE(I)/T)
1 *(1.0 - EXP(-14387.9/(ROTLAM(I)*T)))
HRAC(I) = RADFAC*ROTEA(I)*SFUN(TAU)
2 *EXP(-1.43879*(1.0E+4/ROTLAM(I) + ROTLSE(I))/T)
CONTINUE
RETURN
200
TAUFAC = 4.8815E-18*SQRT(RM/T)*DEN
TEFAC = (T - 273.15)/(T*273.15)*1.43879
SUM = 0.0
DO 210 I = 1, 434
HSTR = TCFAC*HHOSTR(I)*EXP(TEFAC*HHOLSE(I))
TAU = TAUFAC*HHOLAM(I)*HSTR
HRAC(I) = TAU*SFUN(TAU)*EXP(-14387.9/(HHCLAM(I)*T))

```

```

1 /HTCLAM(I)**4
  SUM = SUM + HRAD(I)
210 CONTINUE
  RADFAC = RAD/SUM
  DC 220 I = 1, 434
  HRAC(I) = HRAD(I)*RADFAC
220 CONTINUE
  RETURN
300 CONTINUE
  TGFAC = (238.0/T)**1.5
  TAU FAC = 4.8815E-18*SGRT(RH/T)*CEN*TGFAC
  TEFAC = (1 - 238.0)/(T*238.0)*1.43679
  WAVE = 597.5
  SUM = 0.0
  CC 310 I = 1, 14
  RLAM = 1.0E+4/WAVE
  TAU = TAU FAC*STRO3(I)*EXP(TEFAC*ENLO3(I))*RLAM
  HRAC(I) = TAU*SFUN(TAU)*NLC3(I)*EXP(-1.43879*WAVE/I)
1 * (WAVE/I.0E4)**4
  HRAC(I + 300) = RLAM
  SUM = SUM + HRAC(I)
  WAVE = WAVE + 5.0
310 CONTINUE
  TGFAC = (281.0/T)**1.5
  TAU FAC = 4.8815E-18*SGRT(RH/T)*CEN*TGFAC
  TEFAC = (1 - 281.0)/(T*281.0)*1.43679
  DC 320 I = 15, 32
  RLAM = 1.0E+4/WAVE
  TAU = TAU FAC*STRO3(I)*EXP(TEFAC*ENLO3(I))*RLAM
  HRAC(I) = TAU*SFUN(TAU)*NLC3(I)*EXP(-1.43879*WAVE/I)
1 * (WAVE/I.0E4)**4
  HRAC(I + 300) = RLAM
  SUM = SUM + HRAC(I)
  WAVE = WAVE + 5.0
320 CONTINUE
  RADFAC = RAD/SUM
  HRAC 370
  HRAC 380
  HRAC 390
  HRAC 400
  HRAC 410
  HRAC 420
  HRAC 430
  HRAC 440
  HRAC 450
  HRAC 460
  HRAC 470
  HRAC 480
  HRAC 490
  HRAC 500
  HRAC 510
  HRAC 520
  HRAC 530
  HRAC 540
  HRAC 550
  HRAC 560
  HRAC 570
  HRAC 580
  HRAC 590
  HRAC 600
  HRAC 610
  HRAC 620
  HRAC 630
  HRAC 640
  HRAC 650
  HRAC 660
  HRAC 670
  HRAC 680
  HRAC 690
  HRAC 700
  HRAC 710
  HRAC 720

```

DO 330 I = 1, 32
HRAO(I) = HRAO(I)*RAOFAC
330 CCNTINUE
RETURN
END

HSTR 730
HSTR 740
HSTR 750
HSTR 760
HSTR 770


```

SLURCLINE ROTATE
DIMENSION EALC(250),EALR(250)
DIMENSION WVLTH(250),EA(250)
DIMENSION ENL(250),ENL(250),ENERGY(250)
DIMENSION PCLAM(450),HMCSTR(450),PHOLSE(450),PRAG(600)
DIMENSION ROTLAM(250),RCTEA(250),RCTLSE(250),RCTST(250)
COMMON /HPCCCM/ HHOLAM,PHOSTR,PHCLSE,PRAC
COMMON /RCTCCM/ ROTLAM,RCTEA,RCTLSE,ROTSI
TEFAC = (273.15 - 287.7)/(273.15*287.7)*1.43875
TGFAO = (287.7/273.15)**1.5
WRITE (2,16)
DO 110 I = 1, 434
  READ (1,21) HHOLAM(I),HMCSTR(I),PHCLSE(I)
  WRITE (2,21) PHCLAM(I),PHCSTR(I),PHCLSE(I)
  HHOLAM(I) = 1.0E+4/HHOLAM(I)
  PHOSTR(I) = PHCSTR(I)*8.083E-4*TCFAC
  *EXP(TEFAC*HHCLSE(I))
110 CONTINUE
C
  READ IN MOLECULAR CONSTANTS,
  READ (1,12) WCLWT,RA,RE,RC,CIFCLE,GRFAC
  EAFAC = 64.0*3.1415927**4*DIPCLE*DIPOLE/(3.0*6.625EE-27)
  OR = GRFAC*SCRT((273.15/1.43875)**3*3.14159/(RA*RE*RC))
  READ IN DATA FOR LINES
  READ (1,6) NLINES
DO 130 I = 1, NLINES
  READ (1,10) JU,KAU,KCU,JL,KAL,KCL,STR,
1  ENU(I),ENL(I),ENERGY(I),SIWT
  WRITE (2,10) JL,KAU,KCU,JL,KAL,KCL,STR,
1  ENU(I),ENL(I),ENERGY(I),SIWT
  WVLTH(I) = 10000.0/ENERGY(I)
  EA(I) = EAFAC*STR*ENERGY(I)**3/(JL*(JU + 1))
  EALR(I) = EA(I)*SIWT*JU*(JU + 1)/WVLTH(I)
  EALC(I) = EALR(I)*WVLTH(I)**4
  IF (ABS(ENL(I) - ENL(I) - ENERGY(I)) .GT. 0.005) WRITE (2,4)
  ROTS(I) = EALC(I),OR
  ROTE(I) = EALR(I)/OR

```

FCTR 370
 FCTR 380
 FCTR 390
 FCTR 400
 FCTR 410
 FCTR 420
 FCTR 430
 FCTR 440
 FCTR 450
 FCTR 460
 FCTR 470
 FCTR 480
 FCTR 490
 FCTR 500
 FCTR 510

```

130  RCTLAP(I) = WVLTH(I)
      RCILSE(I) = ENL(I)
      CONTINUE
      RETURN (21H ERROR IN ABOVE CARD)
      4  FORMAT (6I6)
      6  FORMAT (6E12.4)
      8  FORMAT (5I3,F12.6,3F9.2,F5.1,E12.3,F12.3)
      10 FORMAT (4F12.4,E12.4,F12.6)
      12 FORMAT (1X)
      14 FORMAT (1F1)
      16 FORMAT (F6.2,F10.4,F8.2)
      21 FORMAT (F10.5,E12.4,F10.2)
      22 FORMAT (F10.5,E12.4,F10.2,E12.4)
      23  END
  
```

SIMP 10
 SIMP 20
 SIMP 30
 SIMP 40
 SIMP 50
 SIMP 60
 SIMP 70
 SIMP 80
 SIMP 90
 SIMP 100
 SIMP 110
 SIMP 120
 SIMP 130

```

100  FUNCTION SIMP (A,H,K)
      DIMENSION A(3)
      IF (K - 2) 100,110,120
      CONTINUE
      SIMP = (8.0*A(2) + 5.0*A(1) - A(3))*H/12.0
      RETURN
110  CONTINUE
      SIMP = (8.0*A(2) + 5.0*A(3) - A(1))*H/12.0
      RETURN
120  CONTINUE
      SIMP = (4.0*A(2) + A(1) + A(3))*H/3.0
      RETURN
      END
  
```

SFUN 10
 SFUN 20
 SFUN 30
 SFUN 40
 SFUN 50
 SFUN 60
 SFUN 70
 SFUN 80
 SFUN 90
 SFUN 100
 SFUN 110
 SFUN 120
 SFUN 130
 SFUN 140
 SFUN 150
 SFUN 160
 SFUN 170
 SFUN 180
 SFUN 190
 SFUN 200

```

FUNCTION SFUN (TAU)
  T = ABS(TAU)
  IF (T.GT. 4.0) GO TO 110
  X = (T - 2.0)*0.5
  SFUN = (((0.001061*X - 0.004288)*X + 0.013686)*X - 0.042491)
    1 *X + 0.114932)*X - 0.267057)*X + 0.556485
  RETURN
110 CONTINUE
  IF (T.GT. 10.0) GO TO 120
  X = (T - 7.0)/3.0
  SFUN = (((0.000347*X - 0.001167)*X + 0.03148)*X - 0.001545)
    1 *X + 0.028248)*X - 0.081972)*X + 0.247900
  RETURN
120 CONTINUE
  Y = ALOG(T)
  X = 1.0/Y
  SFUN = (((0.067876*X - 0.288698)*X + 0.264568)*X
    1 - 0.269167)*X + 0.325368)*X + 1.128379)*SQRT(Y)/T
  RETURN
END
  
```

```

FUNCTION GINT (A,H1,H2,K)
DIMENSION A(3)
HS = H1 + H2
IF (K - 2) 100,110,120
100 CONTINUE
GINT = ((A(2)*HS - A(1)*H2 - A(3)*H1)/(6.0*H2*HS)
+ (A(1) + A(2))*0.5)*H1
110 RETURN
GINT = ((A(2)*HS - A(1)*H2 - A(3)*H1)/(6.0*H1*HS)
+ (A(2) + A(3))*0.5)*H2
120 RETURN
GINT = (A(2)*HS - A(1)*H2 - A(3)*H1)*(H1**3 + H2**3)
/(6.0*HS*H1*H2) + (A(1)*H1 + A(2)*H2)*0.5
1 RETURN
END

```

```

GINT 10
GINT 20
GINT 30
GINT 40
GINT 50
GINT 60
GINT 70
GINT 80
GINT 90
GINT 100
GINT 110
GINT 120
GINT 130
GINT 140
GINT 150
GINT 160
GINT 170

```

FLTR 10
FLTR 20
FLTR 30
FLTR 40
FLTR 50
FLTR 60
FLTR 70
FLTR 80
FLTR 90
FLTR 100
FLTR 110
FLTR 120
FLTR 130
FLTR 140
FLTR 150
FLTR 160
FLTR 170

```

SUBROUTINE FILTER (SPEC,RAD,HVLT,N)
  DIMENSION SPEC(201),RAD(450),HVLT(450)
  DO 150 I = 1, N
    RT = HVLT(I)
    RADT = RAD(I)
    JS = (RT - 5.25)*10.0 + 2.0
    JF = (RT - 4.75)*10.0 + 1.0
    IF (JS .LT. 1) JS = 1
    IF (JF .GT. 201) JF = 201
    IF (JS .GT. JF) GO TO 150
    DO 120 J = JS,JF
      RLAM = (J - 1)*0.1 + 5.0
      SPEC(J) = SPEC(J) + (0.25 - ABS(RLAM - RT))*RADT*16.0
    CONTINUE
  CONTINUE
  RETURN
END
120
150

```

```

SUBROUTINE ROTSTR (RAC,RM,EVFF,T,CEN,SIR,FS,GS,RS,AF,NG,F,O,R)
DIMENSION P(100),Q(100),R(100),FS(100),QS(100),RS(100)
NC = EC. 1 FOR ALL LINES
ND = EC. 2 FOR ALTERNATE LINES
NP = EC. 1 FOR PARALLEL EANC
NPP = EC. 2 FOR PERPENDICULAR EANC
SIG = 1.43479*EVFF/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
CCONTINUE
GO TO 300
200 CCONTINUE
DG 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
CCONTINUE
300 CCONTINUE
TAUFAC = 4.8815E-18*SGRT(RM/T)*CEN*STR/SUM
SUM = 0.0
DO 310 I = 1, 100

```

```

RSTR 10
RSTR 20
RSTR 30
RSTR 40
RSTR 50
RSTR 60
RSTR 70
RSTR 80
RSTR 90
RSTR 100
RSTR 110
RSTR 120
RSTR 130
RSTR 140
RSTR 150
RSTR 160
RSTR 170
RSTR 180
RSTR 190
RSTR 200
RSTR 210
RSTR 220
RSTR 230
RSTR 240
RSTR 250
RSTR 260
RSTR 270
RSTR 280
RSTR 290
RSTR 300
RSTR 310
RSTR 320
RSTR 330
RSTR 340
RSTR 350
RSTR 360

```

FSIR 370
 FSIR 380
 FSIR 390
 FSIR 400
 FSIR 410
 FSIR 420
 FSIR 430
 FSIR 440
 FSIR 450
 FSIR 460
 FSIR 470
 FSIR 480
 FSIR 490
 FSIR 500
 FSIR 510
 FSIR 520

```

PTAL = FS(I)*P(I)*TAUFAC
CTAL = GS(I)*G(I)*TAUFAC
RTAU = FS(I)*R(I)*TAUFAC
PS(I) = PTAL*SFUN(PTAU)*EXP(-14387.9/(P(I)*T))/P(I)**4
GS(I) = CTAL*SFUN(CTAU)*EXP(-14387.9/(G(I)*T))/G(I)**4
RS(I) = RTAL*SFUN(RTAU)*EXP(-14387.9/(R(I)*T))/R(I)**4
SUM = SUM + PS(I) + GS(I) + RS(I)
310 CONTINUE
RADFAC = RAD/SUM
CO 320 I = 1, 100
PS(I) = PS(I)*RADFAC
GS(I) = GS(I)*RADFAC
RS(I) = RS(I)*RADFAC
320 CONTINUE
RETURN
END
  
```



```

PROGRAM TRYCOT (INPUT, OUTPUT, TAPF2, TAPF5=INPUT,
1 TAPF6=OUTPUT, TAPF30)
VERSION 2
COMMON AVLAM, SECTRA, X, Y
DIMENSION IDENT(3)
DIMENSION AVLAM(201), SECTRA(201,12)
DIMENSION X(201), Y(2412)
DIMENSION XRCN(7), YRCN(7)
DIMENSION JREAM(12)
DIMENSION TFMT(6)
DATA IDENT(2), IDENT(3) /104 NFGGFS, 104 7023 0-4/
DATA JREAM /16,15,14,13,16,15,14,13,16,15,14,13/
DATA TFMT /104(5X,5HSTA,10HPT,6X,3HEN,104N,7X,5HNOT,
1 10HAL,6X,5HTT,10HMF/1X,3F1,5H0.3/)/
6 FORMAT (1X,15)
8 FORMAT (1X,12,1X,7A10)
10 FORMAT (1X,F9.3,1X,F9.3)
16 FORMAT (1H1)
CALL SFCOND (SECTRA)
CALL DATE (IDENT(1))
NH = 6
REWIND 2
WRITE (6,16)
SX = 9.0
SY = 14.0
NH = 2
CALL CRTPLT (IDENT,1.0,17.0)
CALL PLOT (1.0,1.0,-3)
READ (5,6) NCASES
WRITE (6,6) NCASES
READ (5,8) NXL,XRCN
WRITE (6,8) NXL,XRCN
READ (5,8) NYL,YRCN
WRITE (6,8) NYL,YRCN
ON 400 NCS = 1, NCASES
READ (5,10) WVMIN,WVMAX
TCRT 10
TCRT 20
TCRT 30
TCRT 40
TCRT 50
TCRT 60
TCRT 70
TCRT 80
TCRT 90
TCRT 100
TCRT 110
TCRT 120
TCRT 130
TCRT 140
TCRT 150
TCRT 160
TCRT 170
TCRT 180
TCRT 190
TCRT 200
TCRT 210
TCRT 220
TCRT 230
TCRT 240
TCRT 250
TCRT 260
TCRT 270
TCRT 280
TCRT 290
TCRT 300
TCRT 310
TCRT 320
TCRT 330
TCRT 340
TCRT 350
TCRT 360

```

```

WRITE (5,10) WVMIN,WVMAX
TNS = 10.0*WVMIN - 49.0
TF (TNS .LT. 1) TNS = 1
TNF = 10.0*WVMAX - 49.0
TF (TNF .GT. 201) TNF = 201
NX = TNF - TNS + 1
NY = 12*NX
N0 300 NR = 1, 2
READ (2) AVLAM
N0 110 T = 1, NX
X(T) = AVLAM(TNS + T - 1)
110 CONTINUE
CALL LTNCSA (X,NX,SX,YMIN,NY,KONY)
CAP = SY/FLCAT(KONY/10)
GX = CAP*NY
NGAB = MOD(KONY,10)
N0 250 NSPC = 1, 2
READ (2) SORTQA
IJ = 1
N0 130 J = 1, 12
N0 120 T = 1, NX
Y(IJ) = SORTQA(TNS + T - 1,J)
IJ = IJ + 1
CONTINUE
120 CONTINUE
130 CONTINUE
CALL LOGSCA (Y,NY,SY,MTNY,MAXY,3)
CALL STREAM (16)
CALL LTNAXS (SY,0.0,YCDD,-NXL,SX,
1 90.0,XMIN,SX,CAP,NH,NGAB)
CALL LTNAXS (0.0,0.0,YCDD,NXL,SX,
1 90.0,XMTN,SX,CAP,NH,-NGAB)
CALL LOGAXS (SY,0.0,YCDD,NYL,SY,
1 180.0,MTNY,MAXY,1)
CALL LOGAXS (SY,SX,YCDD,-NYL,SY,
1 180.0,MTNY,MAXY,0)
CALL JOIN (FEX(CAP - 0.9),CAP,SY)

```

```

TCRT 370
TCRT 380
TCRT 390
TCRT 400
TCRT 410
TCRT 420
TCRT 430
TCRT 440
TCRT 450
TCRT 460
TCRT 470
TCRT 480
TCRT 490
TCRT 500
TCRT 510
TCRT 520
TCRT 530
TCRT 540
TCRT 550
TCRT 560
TCRT 570
TCRT 580
TCRT 590
TCRT 600
TCRT 610
TCRT 620
TCRT 630
TCRT 640
TCRT 650
TCRT 660
TCRT 670
TCRT 680
TCRT 690
TCRT 700
TCRT 710
TCRT 720

```

```

CALL GRID (MINY + 1, MAXY, SY/FLNAT(MAXY - MINY), SX)
DO 150 I = 1, NY
  Y(I) = SY - Y(I)
150 CONTINUE
DO 200 J = 1, 12
  CALL STRFAM (JPEAM(J))
  IJ = (J - 1)*NX + 1
  CALL NULINE (Y(IJ), X, NX, SY, SX)
200 CONTINUE
  CALL FRAME (1.5, 1.5)
250 CONTINUE
300 CONTINUE
  REMIND 2
400 CONTINUE
  CALL ENRPLT
  CALL SECOND (SECA)
  SECC = SECC - SECA
  WRITE (NH, TEND) SECA, SECC, SECC
  STOP
END

```

```

TCRT 730
TCRT 740
TCRT 750
TCRT 760
TCRT 770
TCRT 780
TCRT 790
TCRT 800
TCRT 810
TCRT 820
TCRT 830
TCRT 840
TCRT 850
TCRT 860
TCRT 870
TCRT 880
TCRT 890
TCRT 900
TCRT 910
TCRT 920

```

SUBROUTINE LNSC0 (X,N,S,XMIN,OX,KOM)

THE SCALING ALGORITHM GUARANTEES THAT THE PLOT WILL USE AT
LEAST SEVENTY PER CENT OF THE LENGTH OF A LINEAR AXIS.

DIMENSION A(14),KA(14),KR(14),Y(100)
DATA A /1.0,1.2,1.5,1.8,2.0,2.4,3.0,4.0,5.0,6.0,7.5,
1 9.0,10.0,12.0/
DATA KA / 10,12,15,9,20,12,15,10,10,12,15,9,10,12 /
DATA KR / 44,33,34,32,44,33,34,44,44,33,34,32,44,33 /

XX = X(1)
XMIN = XMX

ADX = 0.0

DO 110 I = 2, N

XX = AMAX1(XMX,Y(I))

XMIN = AMIN1(XMIN,Y(I))

110 CONTINUE

SX = XMX - XMIN

K = 0

120 CONTINUE

IF (SX .GT. 0.99999) GO TO 130

SX = 10.0*SX

K = K + 1

GO TO 120

130 CONTINUE

IF (SX .LT. 10.0) GO TO 140

SX = 0.1*SX

K = K + 1

GO TO 130

140 CONTINUE

DO 150 J = 1, 14

IF (SX .GT. 1.0005*A(J)) GO TO 150

ST = A(J)*10.0**K

OX = ST/FLOAT(KA(J))

XMIN = XMIN - AMOD(OX + AMOD(XMIN,OX),OX)

OX = ST/S

LNSC 10
LNSC 20
LNSC 30
LNSC 40
LNSC 50
LNSC 60
LNSC 70
LNSC 80
LNSC 90
LNSC 100
LNSC 110
LNSC 120
LNSC 130
LNSC 140
LNSC 150
LNSC 160
LNSC 170
LNSC 180
LNSC 190
LNSC 200
LNSC 210
LNSC 220
LNSC 230
LNSC 240
LNSC 250
LNSC 260
LNSC 270
LNSC 280
LNSC 290
LNSC 300
LNSC 310
LNSC 320
LNSC 330
LNSC 340
LNSC 350
LNSC 360

```

LNSC 370
LNSC 380
LNSC 390
LNSC 400
LNSC 410
LNSC 420
LNSC 430
LNSC 440
LNSC 450
LNSC 460

```

```

      KON = KQ(T)
      IF ((XMX - XMIN)/DX .LT. S*1.0005) GO TO 150
150  CONTINUE
160  CONTINUE
      NO 220 J = 1, N
      X(J) = (X(T) - XMIN)/DX
220  CONTINUE
300  CONTINUE
      RETURN
      END

```

```

SUBROUTINE LOGSCA (X,N,S,MIN,MAX,K)
  IF K IS POSITIVE, K LOG CYCLES WILL BE PRODUCED.
  IF K IS NEGATIVE, INPUT VALUE OF MTN WILL BE USED.
  DIMENSION X(100)
  MTNSAV = MTN
  XX = ALOG10(ABS(X(1))) + 1.0E-99)
  X(1) = XX
  XMAX = XX
  XMIN = XX
  DO 110 I = 2, N
    XX = ALOG10(ABS(X(I))) + 1.0E-99)
    X(I) = XX
    XMAX = AMAX1(XMAX,XX)
    XMIN = AMIN1(XMIN,XX)
  110 CONTINUE
  MAX = XMAX
  MTN = XMIN
  IF (FLOAT(MAX) .LT. XMAX) MAX = MAX + 1
  IF (FLOAT(MTN) .GT. XMTN) MTN = MTN - 1
  IF (K .LT. 0) MTN = MTNSAV
  IF (K .EQ. 0) GO TO 120
  MTN = MAX0(MIN, MAX - K)
  120 CONTINUE
  FAC = S/FLOAT(MAX - MIN)
  XMIN = MIN
  DO 130 I = 1, N
    X(I) = (X(I) - XMIN)*FAC
  130 CONTINUE
  IF (X(T) .LT. 0.0) X(T) = 0.0
  RETURN
END

```

LGSC 10
 LGSC 20
 LGSC 30
 LGSC 40
 LGSC 50
 LGSC 60
 LGSC 70
 LGSC 80
 LGSC 90
 LGSC 100
 LGSC 110
 LGSC 120
 LGSC 130
 LGSC 140
 LGSC 150
 LGSC 160
 LGSC 170
 LGSC 180
 LGSC 190
 LGSC 200
 LGSC 210
 LGSC 220
 LGSC 230
 LGSC 240
 LGSC 250
 LGSC 260
 LGSC 270
 LGSC 280
 LGSC 290
 LGSC 300
 LGSC 310

```

SUBROUTINE LTNAXS (X,Y,BDN,NC,S,THETA,XMIN,XY,GAP,NH,NC)
DIMENSION BCD(4)
DATA W /104 X 10 /
NJ = TARS(NC)
NSN = 1
IF (NC.LT. 0) NSN = -1
TH = THETA*0.017453294
SN = NSN
MC = NC*NSN
N = S/GAP + 0.5
S2 = 0.5*S
CTH = COS(TH)
STH = SIN(TH)
CFAC = CTH*GAP/FLOAT(NJ + 1)
SFAC = STH*GAP/FLOAT(NJ + 1)
NARS = IARS(NH)
ANJ = FLOAT(NARS)
TF (NARS.GT. 3) ADJ = 5.0
SARJ = ADJ*STH
CANJ = ADJ*CTH
XA = X + 0.0875*SN*SADJ
YA = Y - 0.0875*SN*CANJ
XN = X + 0.05*SN*ANJ
YN = Y - 0.05*SN*CANJ
HEIGHT = 0.05*ANJ
CALL PLOT (XA,YA,3)
XA = X
YA = Y
DO 120 I = 1, N
DO 110 J = 1, NJ
CALL PLOT (XR,YR,2)
XC = XR + CFAC
YC = YR + SFAC
CALL PLOT (XC,YC,2)
XD = XN + CFAC
YD = YN + SFAC

```

```

LNAX 10
LNAX 20
LNAX 30
LNAX 40
LNAX 50
LNAX 60
LNAX 70
LNAX 80
LNAX 90
LNAX 100
LNAX 110
LNAX 120
LNAX 130
LNAX 140
LNAX 150
LNAX 160
LNAX 170
LNAX 180
LNAX 190
LNAX 200
LNAX 210
LNAX 220
LNAX 230
LNAX 240
LNAX 250
LNAX 260
LNAX 270
LNAX 280
LNAX 290
LNAX 300
LNAX 310
LNAX 320
LNAX 330
LNAX 340
LNAX 350
LNAX 360

```

LNAX 370
LNAX 380
LNAX 390
LNAX 400
LNAX 410
LNAX 420
LNAX 430
LNAX 440
LNAX 450
LNAX 460
LNAX 470
LNAX 480
LNAX 490
LNAX 500
LNAX 510
LNAX 520
LNAX 530
LNAX 540
LNAX 550
LNAX 560
LNAX 570
LNAX 580
LNAX 590
LNAX 600
LNAX 610
LNAX 620
LNAX 630
LNAX 640
LNAX 650
LNAX 660
LNAX 670
LNAX 680
LNAX 690
LNAX 700
LNAX 710
LNAX 720

```

CALL PLOT (X0,Y0,Z)
XR = XC
YR = YC
110 CONTINUE
CALL PLOT (XR,YR,Z)
XC = XR + CFAC
YC = YR + CFAC
CALL PLOT (XC,YC,Z)
XA = XB + GAP*CFH
YA = YB + GAP*CFH
CALL PLOT (XA,YA,Z)
XD = XD + CFAC
YD = YD + CFAC
XB = XC
YB = YR
120 CONTINUE
IF (NG.LE. 0) RETURN
ARSV = XMIN + DX*FLNAT(N)
IF (NH.LT. 0) GO TO 160
ADX = ARS(DX)
E = 0.0
130 CONTINUE
IF (ADX.EQ. 0.0) GO TO 150
IF (ADX.LT. 1.0) GO TO 140
ADX = ADX/10.0
E = E + 1.0
ARSV = 0.1*ARSV
GO TO 130
140 CONTINUE
IF (ADX.GE. 0.1) GO TO 150
ARSV = ARSV*10.0
ADX = ADX*10.0
E = E + 1.0
GO TO 140
150 CONTINUE
IF (DX.GF. 0.0) GO TO 160

```


LNAX 730
LNAX 740
LNAX 750
LNAX 760
LNAX 770
LNAX 780
LNAX 790
LNAX 800
LNAX 810
LNAX 820
LNAX 830
LNAX 840
LNAX 850
LNAX 860
LNAX 870
LNAX 880
LNAX 890
LNAX 900
LNAX 910
LNAX 920
LNAX 930
LNAX 940
LNAX 950
LNAX 960
LNAX 970
LNAX 980
LNAX 990
LNAX1000
LNAX1010
LNAX1020
LNAX1030
LNAX1040
LNAX1050
LNAX1060
LNAX1070
LNAX1080

```

160 ANX = -ANX
    CONTINUE
    XA = XR - (0.1*SN - 0.025)*SADJ - 0.0964*SADJ
    YA = YR + (0.1*SN - 0.025)*SADJ - 0.0964*SADJ
170 CONTINUE
    N = N + 1
    CFAC = 0.0429*SADJ
    SFAC = 0.0420*SADJ
    DO 200 I = 1, N
        IF (NH.LT. 0) GO TO 180
        CALL NUMPR (XA,YA,HEIGHT,ARCV,THETA,3)
        ARSV = ARCV - ANX
        GO TO 190
180 CONTINUE
        TRD = 1
        IF (ARV(ARSV).GF. 10.0) IRD =
1          IFIX(ALOG10(ARV(ARSV)) + 1.0E-9) + 1
        XB = XA - CFAC*FLOAT(TRD - 1)
        YB = YA - SFAC*FLOAT(TRD - 1)
        CALL NUMPR (XB,YB,HEIGHT,ARCV,THETA,3)
        ARSV = ARSV - NY
190 CONTINUE
        XA = XA - GAP*CTH
        YA = YA - GAP*STH
200 CONTINUE
        IF (NH.GF. 0) GO TO 210
        ANC = MC
        XA = X + (S2 - 0.03*ANC*ADJ)*CTH - (0.18*SN - 0.035)*SADJ
        YA = Y + (S2 - 0.03*ANC*ADJ)*STH + (0.18*SN - 0.035)*SADJ
        HEIGHT = 0.07*ADJ
        CALL SYMPO (XA,YA,HEIGHT,ROD,THETA,MC)
        RETURN
210 CONTINUE
        TNC = MC + 7
        XA = X + (S2 - 0.03*TNC*ADJ)*CTH - (0.1875*SN - 0.0375)*SADJ
        YA = Y + (S2 - 0.03*TNC*ADJ)*STH + (0.1875*SN - 0.0375)*SADJ

```

```

HEIGHT = 0.07*ANJ
CALL SYM9L (XA,YA,HEIGHT,PCD,THETA,MC)
IF (F.EQ. 0.0) RETURN
XA = XA + ((TNC - 5.0)*0.06*ANJ)*CTH
YA = YA + ((TNC - 5.0)*0.06*ANJ)*STM
CALL SYM9L (XA,YA,HEIGHT,W,THETA,S)
IF (F - 1.0) 220,250,230
220 CONTINUE
XA = XA + 0.30*CANJ - 0.035*SANJ
YA = YA + 0.30*SANJ + 0.075*CANJ
GO TO 240
230 CONTINUE
XA = XA + 0.30*CANJ - 0.035*SANJ
YA = YA + 0.30*SANJ + 0.035*CANJ
240 CONTINUE
HEIGHT = 0.05*ANJ
CALL NUMBER (X0,YA,HEIGHT,F,THETA,-1)
250 CONTINUE
RETURN
END

```

```

LNAX1090
LNAX1100
LNAX1110
LNAX1120
LNAX1130
LNAX1140
LNAX1150
LNAX1160
LNAX1170
LNAX1180
LNAX1190
LNAX1200
LNAX1210
LNAX1220
LNAX1230
LNAX1240
LNAX1250
LNAX1260
LNAX1270
LNAX1280

```

SUBROUTINE LOGAXS (X,Y,PCN,NC,S,THETA,MIN,MAX,NTEST)
 DIMENSION PCD(4)

SN = 1.0

TF (NC .LT. 0) SN = -1.0

FAC = 1.0

TH = 0.017453294*THETA

CTH = COS(TH)

STH = SIN(TH)

CFAC = CTH*S/FLCAT(MAX - MIN)

SFAC = STH*S/FLCAT(MAX - MIN)

XL = 0.1*SN*STH*FAC

YL = 0.1*SN*CTH*FAC

XW = 1.75*XL

YW = 1.75*YL

CALL PLOT (X + XW,Y - YW,3)

XA = X

YA = Y

N = 0

CONTINUE

D = 2.0

Q = 1.0

XR = XA

YR = YA

CONTINUE

CALL PLOT (XB,YR,2)

PNLOG = ALOG10(P/Q)

XB = XB + CFAC*PNLOG

YB = YB + SFAC*PNLOG

CALL PLOT (XR,YR,2)

IF (P .EQ. 10.0) GO TO 160

P = P + 1.0

Q = Q + 1.0

CALL PLOT (XB + XL,YR - YL,2)

GO TO 150

CONTINUE

CALL PLOT (XB + YW,YR - YW,2)

LGAX 10
 LGAX 20
 LGAX 30
 LGAX 40
 LGAX 50
 LGAX 60
 LGAX 70
 LGAX 80
 LGAX 90
 LGAX 100
 LGAX 110
 LGAX 120
 LGAX 130
 LGAX 140
 LGAX 150
 LGAX 160
 LGAX 170
 LGAX 180
 LGAX 190
 LGAX 200
 LGAX 210
 LGAX 220
 LGAX 230
 LGAX 240
 LGAX 250
 LGAX 260
 LGAX 270
 LGAX 280
 LGAX 290
 LGAX 300
 LGAX 310
 LGAX 320
 LGAX 330
 LGAX 340
 LGAX 350
 LGAX 360

```

N = N + 1
XA = XA + CFAC
YA = YA + SEAC
IF (N - MAX + MIN .LT. 0) GO TO 140
IF (NTEST .EQ. 0) GO TO 300
FAC = 0.7*FAC
XP = MIN
XA = X
YA = Y
170 CONTINUE
XR = XA - ((0.325*SN - 0.1)*STH + 0.235675*CTH)*FAC
YR = YA + ((0.325*SN - 0.1)*CTH - 0.235675*STH)*FAC
RASE = 10.0
IF (XP .NE. 0.0) GO TO 180
XR = XR + 0.064175*CTH*FAC
YR = YR + 0.064175*STH*FAC
RASE = 1.0
180 CONTINUE
CALL NUMERO (XR, YR, 0.1*FAC, RASE, THETA, -1)
IF (XP .LT. 0.0) GO TO 200
IF (XP .EQ. 0.0) GO TO 236
IF (XP .EQ. 1.0) GO TO 230
IF (XP .LT. 10.0) GO TO 190
190 CONTINUE
XT = XR + (0.3143*CTH - 0.1*STH)*FAC
YT = YR + (0.3143*STH + 0.1*CTH)*FAC
GO TO 220
200 CONTINUE
IF (XP .GT. -10.0) GO TO 210
210 CONTINUE
XT = XR + (0.3713*CTH - 0.125*STH)*FAC
YT = YR + (0.3713*STH + 0.125*CTH)*FAC
220 CONTINUE
CALL NUMERO (XT, YT, 0.1*FAC, XP, THETA, -1)
230 CONTINUE
XA = XA + CFAC

```

```

LGAX 370
LGAX 380
LGAX 390
LGAX 400
LGAX 410
LGAX 420
LGAX 430
LGAX 440
LGAX 450
LGAX 460
LGAX 470
LGAX 480
LGAX 490
LGAX 500
LGAX 510
LGAX 520
LGAX 530
LGAX 540
LGAX 550
LGAX 560
LGAX 570
LGAX 580
LGAX 590
LGAX 600
LGAX 610
LGAX 620
LGAX 630
LGAX 640
LGAX 650
LGAX 660
LGAX 670
LGAX 680
LGAX 690
LGAX 700
LGAX 710
LGAX 720

```

LGAX 730
 LGAX 740
 LGAX 750
 LGAX 760
 LGAX 770
 LGAX 780
 LGAX 790
 LGAX 800
 LGAX 810
 LGAX 820
 LGAX 830
 LGAX 840
 LGAX 850
 LGAX 860

```

YA = YA + SFAC
XP = XP + 1.0
IF (XP - 1.0 - FLOAT(MAX) .LT. 0.0) GO TO 170
NC = FLOAT(NC)*SM
T = S/FACT
XR = X + ((T - NC*0.17155 - 0.05725)*0.5*CTH -
1 (0.6*SM - 0.1)*STH)*FAC
YR = Y + ((T - NC*0.17155 - 0.05725)*0.5*STH +
1 (0.6*SM - 0.1)*CTH)*FAC
NR = NC
CALL SYMBOL (XP,YR,0.2*FAC,RCN,THETA,NR)
300 CONTINUE
RETURN
END

```

```

SUBROUTINE GPTD (NL,SPACE,S)
CALL STREAM (12)
NI = TABS(NL)
AX = 0.0
AY = 0.0
IF (SPACE.EQ. 0.0) GO TO 200
IF (NL.EQ. 0) GO TO 200
IF (NL.GT. 0) GO TO 120
GX = 0.0
GY = S
NX = AMS(SPACE)
DY = 0.0
GO TO 130
120 CONTINUE
RX = S
RY = 0.0
GX = 0.0
DY = AMS(SPACE)
130 CONTINUE
IT = 0
DO 150 I = 1, NI
AX = AX + DX
AY = AY + DY
RX = RX + DX
RY = RY + DY
IT = 1 - IT
IF (IT.LT. 0) GO TO 200
IF (IT.GT. 0) GO TO 140
CALL PLOT (GX,GY,Z)
CALL PLOT (AX,AY,Z)
GO TO 150
140 CONTINUE
CALL PLOT (AX,AY,Z)
CALL PLOT (RX,RY,Z)
150 CONTINUE
200 CONTINUE
CALL STREAM (14)
RETURN
END

```

```

GRID 10
GRID 20
GRID 30
GRID 40
GRID 50
GRID 60
GRID 70
GRID 80
GRID 90
GRID 100
GRID 110
GRID 120
GRID 130
GRID 140
GRID 150
GRID 160
GRID 170
GRID 180
GRID 190
GRID 200
GRID 210
GRID 220
GRID 230
GRID 240
GRID 250
GRID 260
GRID 270
GRID 280
GRID 290
GRID 300
GRID 310
GRID 320
GRID 330
GRID 340
GRID 350
GRID 360
GRID 370
GRID 380
GRID 390

```

```

SURROUTNF MULTNF (X,Y,N,SX,SY)
DIMENSION X(100),Y(100)
NL = 0.1
SXA = SX + NL
SYA = SY + NL
XA = X(1)
YA = Y(1)
IPNA = 2
IF (XA .LT. -NL) TPNA = 3
IF (XA .GT. SXA) TPNA = 3
IF (YA .LT. -NL) TPNA = 3
IF (YA .GT. SYA) TPNA = 3
CALL PLOT (XA,YA,3)
DO 150 I = 2, N
  TPNA = IPNA
  XA = X(I)
  YA = Y(I)
  IPNA = 2
  IF (XA .LT. -NL) TPNA = 3
  IF (XA .GT. SXA) TPNA = 3
  IF (YA .LT. -NL) TPNA = 3
  IF (YA .GT. SYA) TPNA = 3
  IPN = 2
  IF (IPNA + TPNA .NE. 4) IPN = 3
  CALL PLOT (XA,YA,IPN)
CONTINUE
RETURN
END
150

```

```

NLTA 10
NLIN 20
NLIN 30
NLIN 40
NLIN 50
NLIN 60
NLIN 70
NLIN 80
NLIN 90
NLIN 100
NLIN 110
NLIN 120
NLIN 130
NLIN 140
NLIN 150
NLIN 160
NLIN 170
NLIN 180
NLIN 190
NLIN 200
NLIN 210
NLIN 220
NLIN 230
NLIN 240
NLIN 250
NLIN 260
NLIN 270
NLIN 280

```