United States
Environmental Protection
Agency

Environmental Sciences Research
Laboratory
Research Triangle Park NC 27711

EPA-600/8-79-015b
June 1979

Research and Development

EPA

A Lagrangian Photochemical Air Quality Simulation Model

PROPERTY OF
DIVISION
OF
METEOROLOGY

Adaptation to the St. Louis—RAPS Data Base Volume II. User's Manual

RESEARCH REPORTING SERIES

Research reports of the Office of Research and Development, U.S. Environmental Protection Agency, have been grouped into nine series. These nine broad categories were established to facilitate further development and application of environmental technology. Elimination of traditional grouping was consciously planned to foster technology transfer and a maximum interface in related fields. The nine series are:

1. Environmental Health Effects Research
2. Environmental Protection Technology
3. Ecological Research
4. Environmental Monitoring
5. Socioeconomic Environmental Studies
6. Scientific and Technical Assessment Reports (STAR)
7. Interagency Energy-Environment Research and Development
8. "Special" Reports
9. Miscellaneous Reports

This report has been assigned to the SPECIAL REPORTS series. This series is reserved for reports which are intended to meet the technical information needs of specifically targeted user groups. Reports in this series include Problem Oriented Reports, Research Application Reports, and Executive Summary Documents. Typical of these reports include state-of-the-art analyses, technology assessments, reports on the results of major research and development efforts, design manuals, and user manuals.

This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

EPA-600/8-79-015b
June 1979

A LAGRANGIAN PHOTOCHEMICAL AIR QUALITY SIMULATION MODEL

Adaptation to the St. Louis - RAPS Data Base

Volume II. User's Manual

by

Fred Lurmann, Daniel Godden, Alan C. Lloyd, Richard A. Nordsieck
Environmental Research and Technology, Inc.
Environmental Analysis Division
2625 Townsgate Road
Westlake Village, California 91361

Contract No. 68-02-2765

Project Officer

Jack H. Shreffler
Meteorology and Assessment Division
Environmental Sciences Research Laboratory
Research Triangle Park, NC 27711

ENVIRONMENTAL SCIENCES RESEARCH LABORATORY
OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
RESEARCH TRIANGLE PARK, NC 27711

DISCLAIMER

This report has been reviewed by the Environmental Sciences Research Laboratory, U.S. Environmental Protection Agency, and approved for publication. Approval does not signify that the contents necessarily reflect the views and policies of the U.S. Environmental Protection Agency, nor does mention of trade names or commercial products constitute endorsement or recommendation for use.

AFFILIATION

Dr. Shreffler, the Project Officer, is on assignment to the Meteorology and Assessment Division, Environmental Sciences Research Laboratory, from the National Oceanic and Atmospheric Administration, U.S. Department of Commerce.

ABSTRACT

A set of instructions have been compiled for use of a Lagrangian photochemical air quality simulation model adapted to the St. Louis, Missouri/Illinois metropolitan region and the Regional Air Pollution Study (RAPS) data base. The computer model, developed by Environmental Research and Technology, Inc., consists of a set of computer programs for the simulation of atmospheric transport, turbulent diffusion, and chemical kenetics of photochemical pollutants. The model is used to predict atmospheric concentrations of ozone, nitrogen dioxide, carbon monoxide, sulfur dioxide, and sulfate within an air column moving at the mean wind speed.

Descriptions of the meteorological, source emissions, and air quality data requirements, as well as sample input and output files, are provided. The computational procedures for using the model and a listing of the computer code are included.

CONTENTS

ABSTRACT iii

LIST OF ILLUSTRATIONS viii

LIST OF TABLES x

1. INTRODUCTION 1-1

2. COMMON FEATURES OF THE CODE MODULES 2-1

2.1 Reading the Data Cards 2-1

2.2 Systems of Units and Time-Date Conventions 2-1

2.3 Specification of Locations 2-4

3. THE METEOROLOGICAL MODULE 3-1

3.1 Raw Data Requirements 3-1

3.2 Control Parameters and Input Data 3-2

3.3 Specification of the Station Measurement Data 3-17

3.4 Modeling Region and Emissions Grid 3-20

3.5 Selection of the Vertical Mesh Point Elevations 3-24

3.6 Description of the Output 3-24

3.7 Internally Specified Data 3-39

3.8 Surface Roughness Length 3-39

3.9 Printer-plot Boundaries 3-43

3.10 Computational Procedures 3-44

3.10.1 Calculation of Air Parcel Trajectories 3-44

3.10.2 Interpolation of Other Atmospheric Parameters Along the Trajectory 3-48

3.10.3 Calculation of Eddy Diffusivity Coefficients 3-48

3.11 Program Modifications 3-49

3.11.1 Changing the Number of Measurement Stations 3-49

3.11.2 Changing the Azimuth Exclusion Sector 3-50

4. THE EMISSIONS MODULE 4-1

4.1 Control Parameters and the Input Deck 4-1

4.1.1 General Inputs 4-11

4.1.2 Area Source Emissions Control Inputs 4-12

4.1.3 Point Source Emissions Control Inputs 4-12

4.2 Description of the RAPS Input Files 4-13

4.2.1 Emissions Grid File 4-13

4.2.2	Area Source Emissions File	4-14
4.2.3	Point Source Emissions File	4-15
4.3	Description of the Outputs	4-16
4.3.1	Area Source Emissions Outputs	4-16
4.3.2	Point Source Emissions Outputs	4-20
4.3.3	Punched Outputs	4-27
4.4	Computational Procedures	4-27
4.4.1	Area Source Emissions Tabulations	4-27
4.4.2	Point Source Emissions Tabulations	4-29
4.5	Internally Specified Data	4-36
4.6	Program Modifications	4-36
5.	THE CHEMICAL-DIFFUSION MODULE	5-1
5.1	Control Parameters and Input Data	5-1
5.1.1	Source Emissions Data	5-25
5.1.2	Control Parameters and the Vertical Mesh	5-26
5.1.3	Chemical Species List	5-28
5.1.4	Emission Species and Surface Deposition Species	5-28
5.1.5	Initial Concentrations	5-29
5.1.6	Chemical Reaction Mechanism Inputs	5-29
5.1.7	Photodissociation Rate Input Parameters	5-31
5.1.8	Sky Clearness, Temperature, and Diffusivity Schedules	5-32
5.2	ERT Photochemical Mechanism	5-33
5.3	Photodissociation Rates	5-33
5.4	Description of the Outputs	5-39
5.5	Computational Procedures	5-54
5.6	Program Modifications	5-56
5.6.1	Changing the Chemical Mechanism	5-56
5.6.2	Changing the Partitioning of Emissions Input Data	5-59
5.6.3	Increasing the Number of Species with Emission Inputs	5-60
5.6.4	Changing the Plot Species	5-60

CONTENTS (CONTINUED)

6. REFERENCES 6-1

APPENDIX A CODE COMMON BLOCK LOCATIONS

1. Meteorological Module Common Blocks
2. Emissions Module Common Blocks
3. Chemical-Diffusion Module Common Blocks

APPENDIX B CODE SUBROUTINE DESCRIPTIONS

1. Meteorological Module Subroutines and Their Use
2. Emissions Module Subroutines and Their Use
3. Chemical-Diffusion Module Subroutines and Their Use
4. Utility Library Subroutines and Their Use

APPENDIX C CODE FORTRAN SOURCE LISTINGS

1. Meteorological Module Listings
2. Emissions Module Listings
3. Chemical-Diffusion Module Listing
4. Utility Library Listing

APPENDIX D AN IMRPOVED PHOTOCHEMICAL MECHANISM AND RELATED MODIFICATIONS TO THE CHEMICAL-DIFFUSION MODULE

1. Description of Chemical Mechanism and Related Modifications
2. KEMOD Input Data Modifications
3. KEMOD FORTRAN Source Code Modifications

LIST OF ILLUSTRATIONS

Figure		Page
1-1	Physical Concept of the Trajectory Model	1-2
1-2	Data Processing Flow Chart	1-4
2-1	Process for Reading Data Cards	2-5
3-1	Meteorological Module Sample Problem Input Listing	3-8
3-2	METMOD Input Variable Sequence and Options	3-13
3-3	The IVG Array	3-23
3-4	Meteorological Module Sample Output	3-26
3-5	Meteorological Module Sample Output	3-27
3-6	Meteorological Module Sample Output	3-28
3-7	Meteorological Module Sample Output	3-29
3-8	Meteorological Module Sample Output	3-30
3-9	Meteorological Module Sample Output	3-31
3-10	Meteorological Module Sample Output	3-33
3-11	Meteorological Module Sample Output	3-34
3-12	Meteorological Module Sample Output	3-35
3-13	Meteorological Module Sample Output	3-36
3-14	Meteorological Module Sample Output	3-37
3-15	Meteorological Module Sample Output	3-38
3-16	Surface Roughness Length Profile	3-42
3-17	Reverse Trajectory Development	3-47
4-1	Emissions Module Sample Problem Input Listing	4-5
4-2	EMMOD Input Variable Sequence and Options	4-10
4-3	Emissions Module Sample Output	4-17
4-4	Emissions Module Sample Output	4-18
4-5	Emissions Module Sample Output	4-19

LIST OF ILLUSTRATIONS (CONTINUED)

Figure		Page
4-6	Emissions Module Sample Output	4-21
4-7	Emissions Module Sample Output	4-22
4-8	Emissions Module Sample Output	4-23
4-9	Emissions Module Sample Output	4-24
4-10	Emissions Module Sample Output	4-25
4-11	Emissions Module Sample Output	4-26
4-12	Emissions Module Sample Output	4-28
4-13	Plume Entrainment in a Lagrangian Air Parcel	4-31
4-14	Lateral Spreading of Plumes Into and Out of Moving Air Parcels	4-32
4-15	Vertical Distribution of Emission Below Inversion Base	4-33
5-1	Chemical-Diffusion Module Sample Problem Input Listing	5-7
5-2	KEMOD Input Variable Sequence and Options	5-24
5-3	Chemical-Diffusion Module Sample Output	5-42
5-4	Chemical-Diffusion Module Sample Output	5-43
5-5	Chemical-Diffusion Module Sample Output	5-44
5-6	Chemical-Diffusion Module Sample Output	5-46
5-7	Chemical-Diffusion Module Sample Output	5-47
5-8	Chemical-Diffusion Module Sample Output	5-48
5-9	Chemical-Diffusion Module Sample Output	5-49
5-10	Chemical-Diffusion Module Sample Output	5-50
5-11	Chemical-Diffusion Module Sample Output	5-52
5-12	Original and Revised Area Source Flux Schedules	5-53

LIST OF TABLES

Table		Page
2-1	Code Utility Subroutine Library	2-2
2-2	Modeling Units	2-2
2-3	Equivalent Clock Times	2-3
3-1	Meteorological Module Input Variable Description	3-3
3-2	The Digital Wind Azimuth System	3-18
3-3	The Station-measured Parameters in the St. Louis Input File	3-18
3-4	Variables Stored in the CON Array	3-21
3-5	Meteorological Module Internally Specified Data	3-40
4-1	Emissions Module Input Variable Description	4-2
4-2	Emissions Module Internally Specified Data	4-38
4-3	Plume Dispersion Coefficients For Non-Tall Stacks	4-40
4-4	Plume Dispersion Coefficients For Tall Stacks	4-41
5-1	Chemical-Diffusion Module Input Variable Description	5-2
5-2	User Specified and Internally Computed Initial Concentrations	5-30
5-3	The ERT Photochemical Reaction Mechanism	5-34
5-4	Chemical Species Symbol Designations	5-37
5-5	NO_2 Photodissociation Rates	5-40
5-6	HCHO Photodissociation Rates	5-41

1. INTRODUCTION

This manual contains instructions for using the Lagrangian trajectory model developed by Environmental Research and Technology, Inc. (ERT). The model consists of a set of computer programs for simulation of Lagrangian transport and atmospheric reactivity of photochemical pollutants. These instructions describe features of the code which a user must understand in order to successfully utilize the model. The manual describes the code as adapted to the St. Louis, Missouri region and the Regional Air Pollution Study (RAPS) data base. Suggestions are provided for adapting the model to regions with different data base structures.

The Lagrangian trajectory code is written in the FORTRAN IV computer language in the single precision mode. The code has 3 main programs, 82 subroutines, 11 functions and block data programs. The maximum storage requirement of any one program is 51,000 words for execution.

A diagram of the physical concept of the model is shown in Figure 1-1. An air parcel is advected along a path (trajectory) determined by the local prevailing wind speed and direction. The air parcel is divided into a mesh of variably spaced points in the vertical direction. As the air parcel moves along a trajectory, it receives primary pollutant emissions from surface and elevated sources. The pollutants within the air parcel undergo chemical reactions and vertical diffusion, with photolysis by ultraviolet radiation driving several of the chemical transformations that occur. At each mesh point in the parcel, concentrations of 30 chemical species are computed. These species include ozone, oxides of nitrogen and sulfur, hydrocarbons, carbon monoxide and radical intermediates of known importance in the formation of photochemical smog. The output of the model consists of these calculated concentrations as functions of time and height along the trajectory.

The model is designed with a modular structure corresponding to the three data processing tasks required to compute atmospheric pollutant concentrations. The first task consists of processing meteorological data to determine air trajectories and characterize atmospheric conditions along the trajectories. This task is performed by the meteorological module of the code, program METMOD. The second task involves processing area source and point source emissions data to assemble

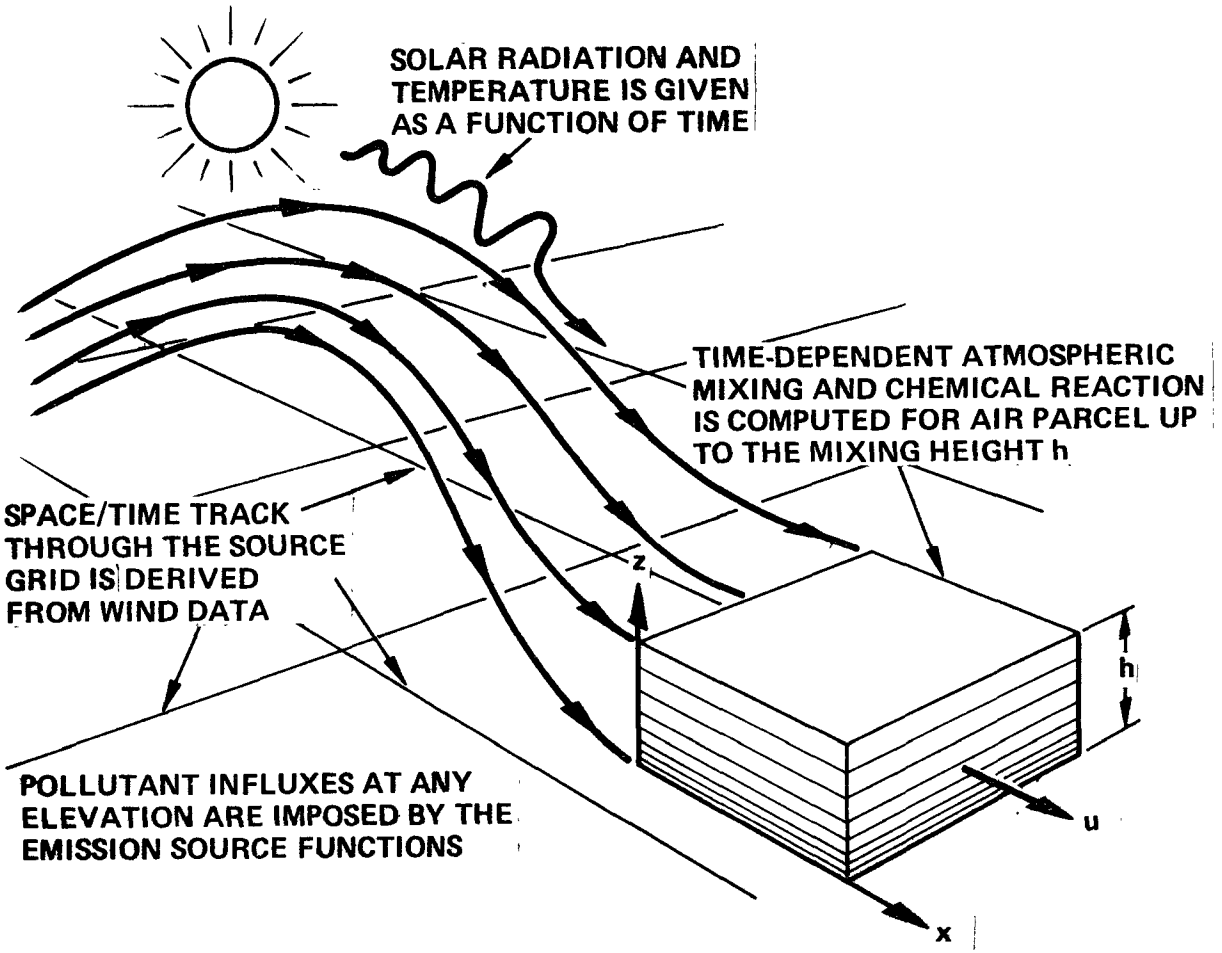


Figure 1-1 Physical Concept of the Trajectory Model

emission rate schedules for the various pollutants along the trajectories. This task is performed by the emission module, program EMMOD. The last task is the computation of the solution to the diffusion equation, with simultaneous chemistry, for the species concentrations along the trajectory. This task is performed by the chemical-diffusion module of the code, program KEMOD.

The data processing activities proceed in the sequence of the tasks described above, as schematically illustrated in Figure 1-2. The user assembles wind, temperature, and ultraviolet radiation data for a single day in the modeling region. A start location, vertical mesh point elevations, and several program control parameters are selected. This data is used by the meteorological module to determine temporal schedules of wind speed and direction, atmospheric mixing coefficients (K_z) and stability, surface temperature, and sky clearness. The trajectory start location, wind speed and direction, surface temperature, and atmospheric stability outputs are transferred to the emissions module input stream. This data is used in addition to the gridded area source emission rates, point source locations, emission rates, and stack parameters, and a schedule of mixing heights by EMMOD to generate the emission source schedule for the trajectory. This emission schedule is then transferred to the chemical-diffusion module input stream along with the temperature, sky clearness, and mixing coefficients output from METMOD. The chemical-diffusion module's input stream is complete upon user specification of initial pollutant concentrations, chemical reaction rates, and several control parameters. The data processing activities are concluded by running the chemical-diffusion module, which outputs the computed concentration profiles.

It should be noted that the meteorological and emissions modules have been equipped with input data retrieval software, which is specific to the RAPS data tape format. Much of the input data is retrieved from the RAPS tapes, not the card-image input stream. Modification of this software is required for application of the model in regions other than St. Louis. The chemical diffusion module input stream is read entirely from the card reader and thus does not bear this restriction. However, it has several subroutines which are specific to the ERT photochemical mechanism.

ERT TRAJECTORY MODEL

DATA PROCESSING FLOW CHART

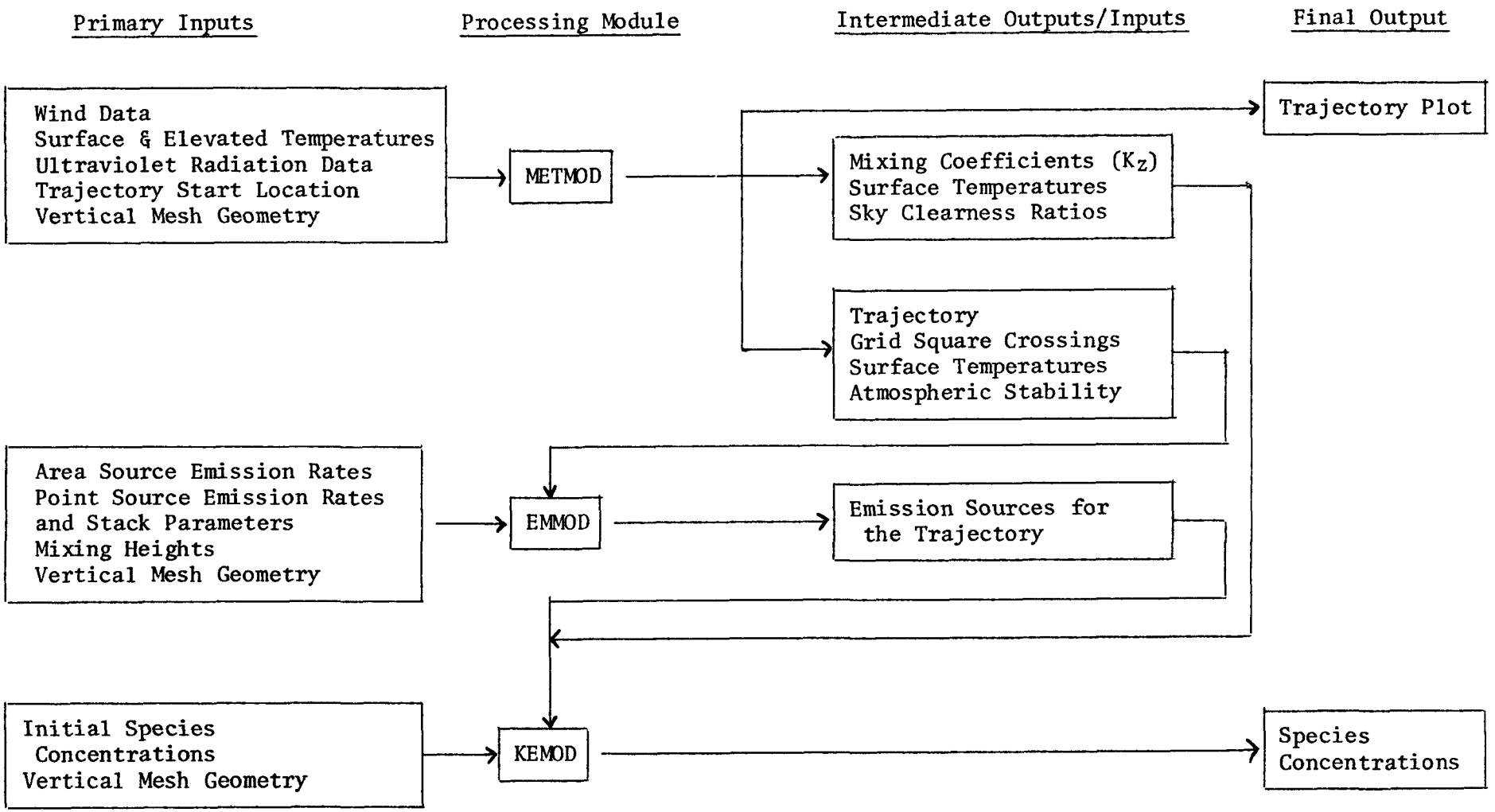


Figure 1-2

The sections of this manual that follow describe the operation of the three modules of the code. Appendix A lists the common block locations in the code. Appendix B includes a description of the programs, subroutines, and functions used by the code. Appendix C contains the FORTRAN source listing of the complete code as run on Control Data Corporation computers. Appendix D describes an alternate chemical mechanism and related modifications for its use.

2. COMMON FEATURES OF THE CODE MODULES

The three modules of the code have certain common features and use subroutines from a common utility program library. This section describes the common card reading process, system of units, time-date conventions, and coordinate systems used by the code. In addition, a list of utility library subroutines is provided in Table 2-1.

2.1 Reading the Data Cards

The process of reading the input cards is illustrated in Figure 2-1. The data cards are read by subroutine PREDAT from the card reader, which is referred to as logical unit 5 [e.g., READ (5,N) LIST]. PREDAT reads the data until it detects an end-of-file mark or a card with the word "MORE" in columns 1-4. It then produces a listing of the data deck on the line printer, which is designated as logical unit 6 [e.g., WRITE (6,N) LIST]. Examples of the output produced by PREDAT are contained in Sections 3, 4 and 5. The subroutine also writes the data on a scratch peripheral disk file designated as logical unit number 3. After writing the data on unit 3, the subroutine rewinds unit 3, and the data are then ready to be read by the program. Each module of the code calls PREDAT at the beginning of the main program.

2.2 Systems of Units and Time-Date Conventions

The model performs its calculations in metric units. Table 2-2 lists the units used by the code. Clock-time is expressed on two different clocks. The model uses the 0-2400 hours military clock and a 0-1440 minutes from midnight clock. Table 2-3 lists some equivalent times on the two clocks. Dates are generally expressed on a year-month-day basis (e.g., YRMODA = 760613 is June 13, 1976). In situations where dates are specified on a Julian basis, leap year corrections are included.

TABLE 2-1

CODE UTILITY SUBROUTINE LIBRARY

	Module Requirements		
Subroutine	METMOD	EMMOD	KEMOD
FMINF	X	X	
MCHAR	X		X
MDATE	X	X	X
NEWPAG	X	X	X
PREDAT	X	X	X
SECOND	X	X	X
SETPLT	X		X
SOLAR	X		X
XMIT	X	X	X

TABLE 2-2

MODELING UNITS

Quantity	Units
Length - vertical	meters
Length - horizontal	kilometers
Time	minutes and hours
Mass	kilograms
Concentration	parts per million
Temperature	degrees Celsius
Luminous intensity	Langleys per minute

TABLE 2-3

EQUIVALENT CLOCK TIMES

Time	0-2400 Hour Clock	0-1440 Minutes Clock
6 AM	600.	360.
7 AM	700.	420.
8 AM	800.	480.
9 AM	900.	540.
10 AM	1000.	600.
11 AM	1100.	660.
12 PM	1200.	720.
1 PM	1300.	780.
2 PM	1400.	840.
3 PM	1500.	900.
4 PM	1600.	960.
5 PM	1700.	1020.
6 PM	1800.	1080.
6:59 PM	1859.	1139.

2.3 Specifications of Locations

The program uses both a global and local coordinate system. Since the convention in air pollution modeling is to use the Universal Transverse Mercator (UTM) coordinate system, the program uses the label UTM for all global coordinate system data. All user specified location coordinates are input in the global coordinate system. The user also specifies an origin for a local coordinate system used internally by the programs.

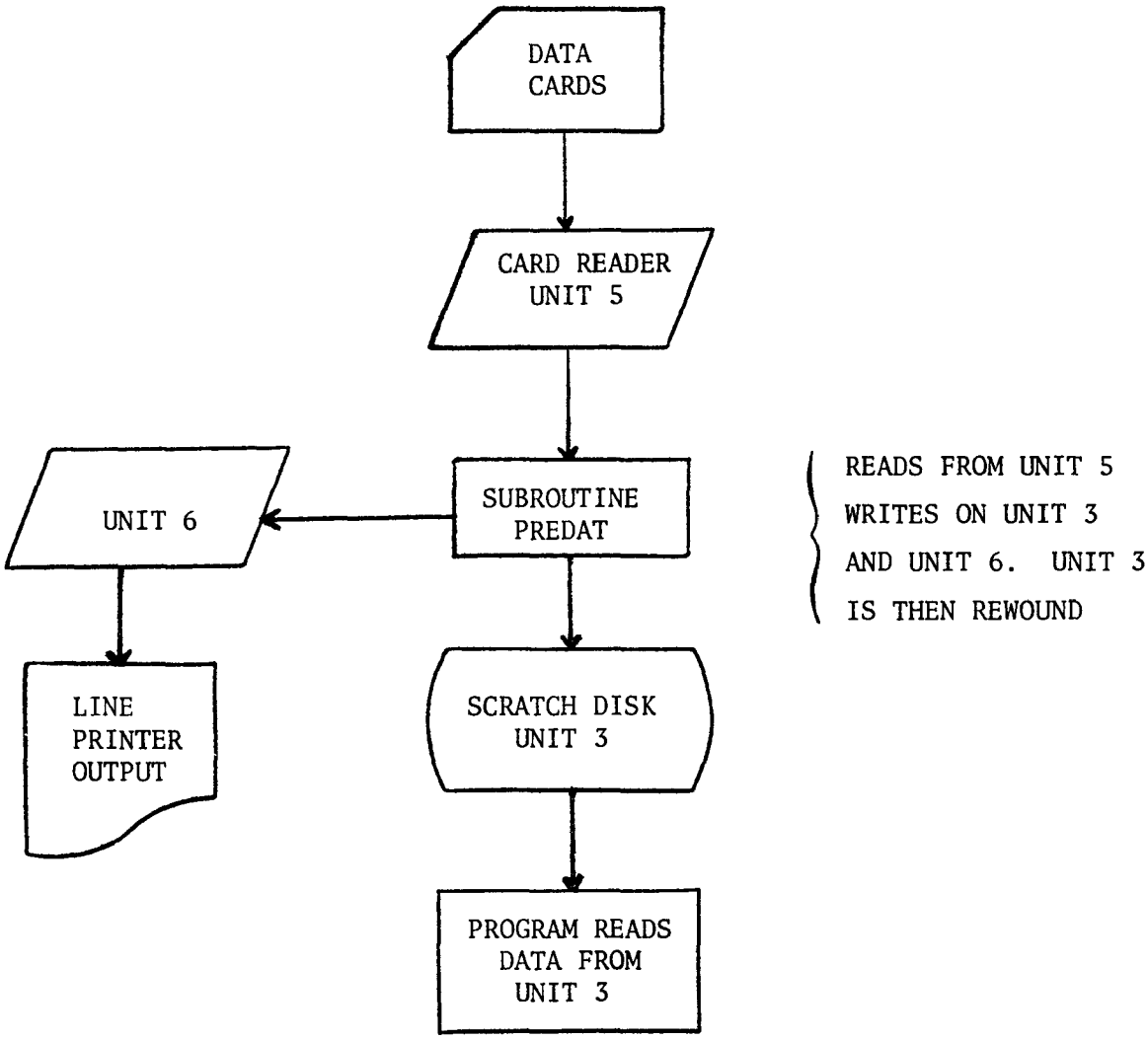


Figure 2-1 Process for Reading Data Cards

3. THE METEOROLOGICAL MODULE

This section describes procedures for using the program METMOD. METMOD generates air parcel trajectories and characterizes atmospheric conditions along the trajectories from basic meteorological data. Raw data requirements, input control parameters, input structures, optional modes of operation, output, and the basic computational procedures are described. Listings of the input and output for a sample problem are included. Since this program requires internal specification of data, which is characteristic of a particular modeling region, the user is encouraged to examine the FORTRAN source listing in Appendix C if he/she is adapting the model for applications to a region other than St. Louis.

3.1 Raw Data Requirements

The meteorological module requires a minimum input data base consisting of the following information:

- Hourly wind speeds and directions (from a network of monitoring stations)
- Hourly surface temperatures (regionally representative)
- Early morning and afternoon vertical temperature profiles (regionally representative).

The fidelity of the modeling is enhanced by using a data base with maximum spatial and temporal resolution. The program has the capability to utilize a data base, which includes the following data from a network of aerometric measurement stations:

- Hourly surface winds
- Hourly surface temperatures
- Vertical temperature profiles (three per day)
- Ultraviolet radiation data
- Ground-level pollutant concentrations

The optional ultraviolet radiation data are important to the modeling when the user intends to simulate days with unclear sky conditions. The pollutant concentrations, on the other hand, are input to the program more as a convenience to the user. They are interpolated along the trajectory in a manner similar to the wind data to approximate the measured concentrations in the Lagrangian reference frame. This computation is, of course, optional but quite useful when the user selects initial pollutant concentrations for the chemical-diffusion module.

3.2 Control Parameters and Input Data

The input parameters of the program will be discussed in the general order in which they are read by the code. The reader should refer to Figure 3-1, which contains a copy of the listing produced by PREDAT, to obtain a better picture of the actual structure of the data deck. Table 3-1 shows a list of the program inputs, along with a description of the function of each input variable, the name of the variable used in the code, and the format. In addition, Figure 3-2 contains a flow chart illustrating the sequence in which the variables are read under the different modes of operation.

It is important for the user to recognize that the selection of trajectories is often an iterative process. The user will usually desire the trajectories to start or end at a particular station or subregion of the modeling region and to have as long a duration as possible. Satisfying these requirements usually requires generation of several trajectories for different start times and/or start locations. The METMOD program is designed with a multiple-case feature, which allows generation of multiple trajectories in a single computer run.

Several modes of operation of the code are available to the user. The variables, which control the selection of these operational modes, are read immediately following the output page label (TITLE) in the input deck. When the user wishes the trajectory to be generated from wind data, he/she sets the input parameter TRAJEX equal to NO. This is the code's most common mode of operation. Alternatively, the user has the option to input a trajectory which has been externally computed. In this case, the user sets TRAJEX = YES and inputs a trajectory start location (input 23) and a schedule of times, wind speeds, and directions (input 24). In either of these cases, the user can request or suppress

TABLE 3-1

METEOROLOGICAL MODULE INPUT VARIABLE DESCRIPTION

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
1	TITLE	Label for each page of output.	(20A4)	May contain any user comments in columns 1-80.
2	TRAJEX	Wind trajectory is input externally (YES) or computed internally (NO).	(40X,A4)	This is a Hollerith variable which equals YES or NO. In either case the word must start in column 41.
3	KPWIND	Output for other code modules punched (YES) or not (NO).	(40X,A4)	See comments for input number 2.
4	NEEDAQ	Interpolation of extra atmospheric parameters is desired (YES) or not (NO).	(40X,A4)	See comments for input number 2.
5	NEEDKZ	Calculation of eddy diffusivity coefficients is desired (YES) or not (NO).	(40X,A4)	See comments for input number 2 and 26-33.
6	LTAPE	Logical unit number of meteorological data file.	(40X,I2)	Equals 3 for cards.
7	KPSTAT	Output a list of the measurement station names and coordinates (YES or NO).	(40X,A4)	See input number 2. Required first case only.
8	KXTRA	Print the extra trajectory output data (YES) or not (NO).	(40X,A4)	See input number 2. Required first case only. (See Section 3.7)
9	CONVRT	Multiplier needed to convert the wind speeds to km/hr.	(40X,F10.0)	Required first case only.

TABLE 3-1 (Continued)

Input Number	Variable Names Used in Code	Variable Function	FORTRAN Format	Remarks
10	UTMXOR, UTMYOR	Global x and y coordinates of the emissions grid origin (km).	(40X,2F10.0)	Required first case only.
11	DELX, DELY	Minimum grid square size (km) in the x and y directions.	(40X,2F10.0)	Required first case only.
12	NX, NY	Number of grid squares in the x and y directions.	(40X,2I10)	Required first case only.
13	RMIN, RMAX	Minimum and Maximum Distances used in the Interpolation Scheme (km).	(40X,2F10.0)	Required first case only.
14	IDT, XLCL	Data and time at the start of the trajectory	(40X,I6,4X, F4.0)	Date on the YRMODA basis and time on the 0-2400 clock. Only the date is read if TRAJEX = YES.
15	STIME	Is the start time STANDARD and DAYLIGHT time?	(40X,A4)	This is a Hollerith variable equal to STAN or DAYL.
16	ID, XS, YS	Trajectory start point location which can either be a station name or the global x and y coordinates.	(40X,2A4,2X, 2F10.0)	Do not specify both a station name and coordinates.
17	TTOTAL	Desired trajectory duration (hours).	(40X,F10.0)	
18	DTSEG	Time interval between trajectory wind calculations or nodes (hours).		Must be an even multiple of the measurement data interval.
19	AZ, VEL	Azimuth (degrees CCW from east to vector parallel to wind) and velocity (km/hr) of initial wind vector.	(40X,2F10.0)	These optional inputs override the initial wind vector calculation. Blanks or zeros yields normal operations.

TABLE 3-1 (continued)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
20	ITF	Trajectory is to be generated forward (+1) or backward (-1) in time.	(40X,I2)	
21	IRF	Station measurement weighted as 1/R if (01) or $1/R^2$ if (02).	(40X,I2)	Required with TRAJEX YES or NO.
22	NC	Number of nearby measurement stations desired for interpolation of station data.	(40X,I2)	$1 \leq NC \leq 3$ required with TRAJEX = YES or NO.
23	P(1,1), P(2,1)	Global x and y coordinates of the trajectory start point for externally input trajectory (km).	(40X,2F10.0)	Required only with TRAJEX = YES. See Figure 3-2 for its input position.
24	T, V, TH	Arrays of the externally input trajectory times, wind velocity, and wind azimuth (degrees CCW from east parallel the wind vector).	(40X,3F10.0)	Required only with TRAJEX = YES. See Figure 3-2 for these input positions. Time is on the 0-2400 basis. The last card should be a negative entry in columns 41-50. A maximum of 100 cards may be input.
25	XR, XL, YB, YT	Left, right, bottom, and top boundaries of the printer-plot (km).	(30X,4F10.0)	In local coordinates as described in Section 3.10.
26	NKZDAT	Read a new set of input data for the eddy diffusivity calculation (YES) or (NO).	(40X,A4)	See input number 2. Must equal YES in first data set with K_z calculation desired.

TABLE 3-1 (CONTINUED)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
27	KZPRTX	Print detailed K_z vertical profiles for each hour as an extra output (YES) or not (NO).	(40X,A4)	See input number 2 and Section 3.7.
28	TIMESD	Time of the vertical temperature profile.	(40X,F10.0)	One card with time on the 0-2400 clock precedes each vertical temperature profile input.
29	ZELVSD, TEMPSD	Arrays of elevations (meters) and corresponding temperatures (degrees C).	(40X,2F10.0)	Each card contains one elevation and temperature. Each profile may contain up to 50 cards. Up to 3 profiles are input, with a negative entry in columns 41-50 after each profile, and an extra negative entry after the last profile.
30	RDTEMP	An externally specified schedule of surface temperature is to be read (YES) or not (NO).	(40X,A4)	See input number 2. If RDTEMP = NO, the temperatures must be previously interpolated from station data.
31	A, TEMPSF	A schedule of times and externally specified surface temperatures (degrees C).	(40X,2F10.0)	One time on the 0-2400 clock and temperature per card. The data must be specified for every hour following the first profile release time through the last trajectory node time. Last card has a negative entry in columns 41-50.

TABLE 3-1 (CONTINUED)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
32	NOHTS	Number of vertical mesh points.	(40X,I10)	The maximum and recommended number of mesh points is 5. See Section 3.5.
33	CELHTS	Elevations above ground of vertical mesh points (meters).	(40X,F10.0)	One elevation per card, input by ascending elevations. See Section 3.5.
34	TERM	Last card for a single data set which equals MORE or END.	(A4)	TERM should equal MORE for all but the last data set in multiple case runs.

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
   1  ERT METEORELOGICAL MODULE      --       ST. LOUIS       --      6/29/76
   2  IS TRAJECTORY INPUT EXTERNALLY           NO
   3  DO WE WANT THE OUTPUTS   PUNCHED         YES
   4  DO WE WANT THE AIR QUALITY DATA          YES
   5  DO WE WANT DIFFUSIVITIES                 YES
   6  LOGICAL UNIT OF MET-DATA PERIPHERRAL     21
   7  PRINT STATION COORDS                     YES
   8  DO WE WANT EXTRA TRAJECTORY OUTPUT       NO
   9  WIND SPEED CONVERSION FACTOR             3.6          (M/S TO KM/HR)
  10  UTM COORDINATES OF GRID ORIGIN    (KM)   680.00     4230.00

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  11  UNIFORM GRID SIZE (DX, DY IN KM)         1.0         1.0
  12  NUMBER OF UNIFORM GRID SQUARES (NX, NY)          100         100
  13  MINIMUM AND MAXIMUM INTERP RADIUS (KM)   1.0         75.0
  14  DATE AND TIME (YRMODA, 2400 HR CLOCK)    760629      600
  15  IS LOCAL TIME STANDARD OR DAYLIGHT       STANDARD
  16  START LOCATION (STATION ID OR UTM X,Y)               680.444    4224.050
  17  TRAJECTORY DURATION  (HOURS)             11.0
  18  TRAJECTORY SEGMENT LENGTH  (HOURS)       1.0
  19  STARTING AZIMUTH AND VELOCITY (OPTIONAL)
  20  DIRECTION FLAG (POS=FRWRD, NEG=BKWRD)    1

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  21  DISTANCE WEIGHTING FACTOR (1/R**  )      02
  22  NUMBER OF CLOSE STATIONS TO USE          03          (MAX = 3)
  23  PRINT - PLOT BOUNDARIES (KM)    -15.0     158.4      -16.0      115.0
  24  DO WE NEED VERT. TEMP PROFILES           YES
  25  DO WE WANT EXTRA KZ OUTPUT               YES
  26   RELEASE  TIME  (CST)          76-06-29   348.                  UAN 142
  27   VERTICAL TEMPERATURE PROFILE  76-06-29   179.0      18.9       UAN 142
  28                                 76-06-29   200.0      19.1       UAN 142
  29                                 76-06-29   215.6      19.1       UAN 142
  30                                 76-06-29   250.7      20.2       UAN 142

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  31                                 76-06-29   295.1      23.3       UAN 142
  32                                 76-06-29   300.0      23.4       UAN 142
  33                                 76-06-29   313.0      23.8       UAN 142
  34                                 76-06-29   400.0      23.2       UAN 142
  35                                 76-06-29   448.5      22.9       UAN 142
  36                                 76-06-29   485.0      23.8       UAN 142
  37                                 76-06-29   500.0      23.8       UAN 142
  38                                 76-06-29   549.4      23.8       UAN 142
  39                                 76-06-29   600.0      23.6       UAN 142
  40                                 76-06-29   670.1      23.3       UAN 142
```

Figure 3-1 Meteorological Module Sample Problem Input Listing

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
  41                                76-06-29   700.0      23.0       UAN 142
  42                                76-06-29   800.0      22.0       UAN 142
  43                                76-06-29   801.5      22.0       UAN 142
  44                                76-06-29   877.3      21.5       UAN 142
  45                                76-06-29   900.0      21.4       UAN 142
  46                                76-06-29   982.4      21.1       UAN 142
  47                                76-06-29  1000.0      20.9       UAN 142
  48                                76-06-29  1078.8      20.0       UAN 142
  49                                76-06-29  1100.0      19.8       UAN 142
  50                                76-06-29  1200.0      18.9       UAN 142

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
  51                                76-06-29  1300.0      18.0       UAN 142
  52                                76-06-29  1362.6      17.5       UAN 142
  53                                76-06-29  1400.0      17.4       UAN 142
  54                                76-06-29  1500.0      17.3       UAN 142
  55                                76-06-29  1522.3      17.3       UAN 142
  56                                76-06-29  1582.8      16.9       UAN 142
  57                                76-06-29  1600.0      16.7       UAN 142
  58                                76-06-29  1700.0      16.0       UAN 142
  59                                76-06-29  1766.1      15.4       UAN 142
  60                                76-06-29  1800.0      15.2       UAN 142

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
  61                                76-06-29  1900.0      14.4       UAN 142
  62                                76-06-29  1973.7      13.9       UAN 142
  63                                76-06-29  2000.0      13.7       UAN 142
  64                                76-06-29  2078.9      13.1       UAN 142
  65                                76-06-29  2100.0      12.9       UAN 142
  66                                76-06-29  2200.0      12.1       UAN 142
  67                                76-06-29  2300.0      11.3       UAN 142
  68                                76-06-29  2400.0      10.5       UAN 142
  69                                76-06-29  2423.0      10.3       UAN 142
  70                                76-06-29  2500.0       9.6       UAN 142

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
  71                                76-06-29  2600.0       8.7       UAN 142
  72                                76-06-29  2632.6       8.5       UAN 142
  73                                76-06-29  2688.4       8.1       UAN 142
  74                                76-06-29  2700.0       8.0       UAN 142
  75                                76-06-29  2800.0       7.3       UAN 142
  76                                76-06-29  2846.2       7.0       UAN 142
  77                                76-06-29  2900.0       6.6       UAN 142
  78                                76-06-29  3000.0       5.8       UAN 142
  79                                76-06-29  3064.2       5.3       UAN 142
  80                                76-06-29  3100.0       5.1       UAN 142
```

Figure 3-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  81                                 76-06-29  3145.6      4.8       UAN 142
  82                                           - 10.0
  83   RELEASE  TIME  (CST)          76-06-29   945.                 UAN 141
  84   VERTICAL TEMPERATURE PROFILE  76-06-29   149.0     27.6       UAN 141
  85                                 76-06-29   161.1     26.7       UAN 141
  86                                 76-06-29   200.0     26.5       UAN 141
  87                                 76-06-29   288.7     26.0       UAN 141
  88                                 76-06-29   300.0     25.8       UAN 141
  89                                 76-06-29   351.9     25.3       UAN 141
  90                                 76-06-29   379.1     23.9       UAN 141

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  91                                 76-06-29   400.0     23.6       UAN 141
  92                                 76-06-29   451.6     23.0       UAN 141
  93                                 76-06-29   500.0     22.5       UAN 141
  94                                 76-06-29   600.0     21.6       UAN 141
  95                                 76-06-29   616.1     21.4       UAN 141
  96                                 76-06-29   700.0     21.5       UAN 141
  97                                 76-06-29   783.3     21.7       UAN 141
  98                                 76-06-29   800.0     21.7       UAN 141
  99                                 76-06-29   900.0     21.7       UAN 141
 100                                 76-06-29  1000.0     21.7       UAN 141

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 101                                 76-06-29  1001.6     21.7       UAN 141
 102                                 76-06-29  1100.0     20.9       UAN 141
 103                                 76-06-29  1195.5     20.1       UAN 141
 104                                 76-06-29  1200.0     20.2       UAN 141
 105                                 76-06-29  1283.9     20.4       UAN 141
 106                                 76-06-29  1300.0     20.3       UAN 141
 107                                 76-06-29  1400.0     19.8       UAN 141
 108                                 76-06-29  1453.1     19.5       UAN 141
 109                                 76-06-29  1500.0     19.1       UAN 141
 110                                 76-06-29  1523.7     18.9       UAN 141

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 111                                 76-06-29  1600.0     18.2       UAN 141
 112                                 76-06-29  1700.0     17.3       UAN 141
 113                                 76-06-29  1799.7     16.4       UAN 141
 114                                 76-06-29  1800.0     16.4       UAN 141
 115                                 76-06-29  1900.0     15.8       UAN 141
 116                                 76-06-29  2000.0     15.2       UAN 141
 117                                 76-06-29  2100.0     14.6       UAN 141
 118                                 76-06-29  2185.7     14.5       UAN 141
 119                                 76-06-29  2200.0     13.7       UAN 141
 120                                 76-06-29  2300.0     12.7       UAN 141
```

Figure 3-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 121                                 76-06-29  2395.8      11.8        UAN 141
 122                                 76-06-29  2400.0      11.8        UAN 141
 123                                 76-06-29  2500.0      11.4        UAN 141
 124                                 76-06-29  2527.7      11.2        UAN 141
 125                                 76-06-29  2600.0      10.7        UAN 141
 126                                 76-06-29  2700.0       9.9        UAN 141
 127                                 76-06-29  2800.0       9.1        UAN 141
 128                                 76-06-29  2876.3       8.5        UAN 141
 129                                 76-06-29  2900.0       8.2        UAN 141
 130                                 76-06-29  2968.1       7.3        UAN 141

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 131                                 76-06-29  3000.0       7.1        UAN 141
 132                                 76-06-29  3100.0       6.7        UAN 141
 133                                 76-06-29  3153.9       6.4        UAN 141
 134                                           - 10.0
 135  RELEASE  TIME  (CST)           76-06-29  1553.                   UAN 141
 136  VERTICAL TEMPERATURE PROFILE   76-06-29   149.0      29.5        UAN 141
 137                                 76-06-29   200.0      29.8        UAN 141
 138                                 76-06-29   278.1      30.3        UAN 141
 139                                 76-06-29   300.0      29.8        UAN 141
 140                                 76-06-29   379.1      27.8        UAN 141

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 141                                 76-06-29   400.0      27.4        UAN 141
 142                                 76-06-29   480.4      25.7        UAN 141
 143                                 76-06-29   500.0      25.6        UAN 141
 144                                 76-06-29   554.5      25.2        UAN 141
 145                                 76-06-29   600.0      24.5        UAN 141
 146                                 76-06-29   610.3      24.3        UAN 141
 147                                 76-06-29   700.0      23.4        UAN 141
 148                                 76-06-29   703.9      23.4        UAN 141
 149                                 76-06-29   800.0      22.2        UAN 141
 150                                 76-06-29   807.6      22.1        UAN 141

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 151                                 76-06-29   854.9      21.4        UAN 141
 152                                 76-06-29   900.0      21.2        UAN 141
 153                                 76-06-29   912.1      21.2        UAN 141
 154                                 76-06-29  1000.0      20.7        UAN 141
 155                                 76-06-29  1017.6      20.6        UAN 141
 156                                 76-06-29  1065.9      20.1        UAN 141
 157                                 76-06-29  1100.0      19.8        UAN 141
 158                                 76-06-29  1200.0      19.0        UAN 141
 159                                 76-06-29  1300.0      18.1        UAN 141
 160                                 76-06-29  1310.5      18.0        UAN 141
```

Figure 3-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 161                                76-06-29  1400.0      17.4        UAN 141
 162                                76-06-29  1499.9      16.6        UAN 141
 163                                76-06-29  1500.0      16.6        UAN 141
 164                                76-06-29  1560.4      16.4        UAN 141
 165                                76-06-29  1600.0      16.2        UAN 141
 166                                76-06-29  1700.0      15.6        UAN 141
 167                                76-06-29  1800.0      14.9        UAN 141
 168                                76-06-29  1900.0      14.3        UAN 141
 169                                76-06-29  1972.9      13.9        UAN 141
 170                                76-06-29  2000.0      13.9        UAN 141

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 171                                76-06-29  2004.5      13.9        UAN 141
 172                                76-06-29  2100.0      13.2        UAN 141
 173                                76-06-29  2200.0      12.5        UAN 141
 174                                76-06-29  2300.0      11.8        UAN 141
 175                                76-06-29  2358.5      11.4        UAN 141
 176                                76-06-29  2400.0      11.2        UAN 141
 177                                76-06-29  2500.0      10.6        UAN 141
 178                                76-06-29  2567.7      10.1        UAN 141
 179                                76-06-29  2600.0      10.1        UAN 141
 180                                76-06-29  2700.0      10.0        UAN 141

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 181                                76-06-29  2713.5      10.0        UAN 141
 182                                76-06-29  2781.6       9.6        UAN 141
 183                                76-06-29  2800.0       9.5        UAN 141
 184                                76-06-29  2872.9       8.9        UAN 141
 185                                76-06-29  2895.9       9.2        UAN 141
 186                                76-06-29  2900.0       9.3        UAN 141
 187                                76-06-29  2930.5       9.4        UAN 141
 188                                76-06-29  3000.0       9.1        UAN 141
 189                                76-06-29  3000.2       9.1        UAN 141
 190                                76-06-29  3100.0       8.2        UAN 141

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 191                                76-06-29  3129.0       8.0        UAN 141
 192  LAST DUMMY NEGATIVE                     - 10.0
 193  SECOND LAST DUMMY NEGATIVE              - 10.0
 194  DO WE WANT EXTERNALLY INPUT SURF. TEMP. NO
 195  NUMBER OF KEMOD VERTICAL MESH POINTS    05
 196  MESH POINT ELEVATIONS (ABOVE SURFACE)    0.0        (METERS)
 197  2                                       120.0
 198  3                                       300.0
 199  4                                       600.0
 200  5                                      1200.0

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 201  END
```

Figure 3-1 (Continued)

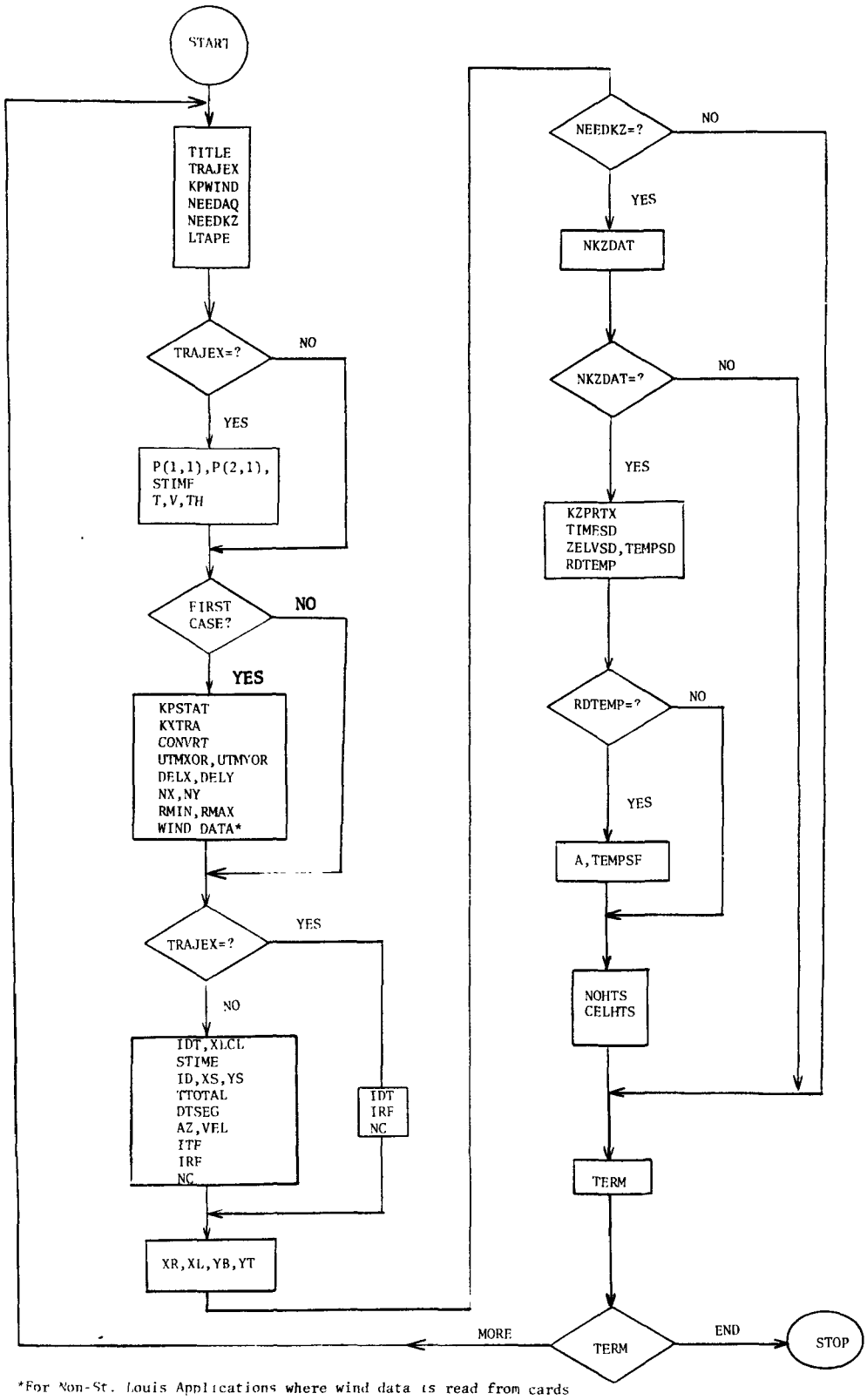


Figure 3-2 METMOD Input Variable Sequence and Options

the generation of eddy-diffusivity (K_z) coefficients by specifying NEEDKZ equal to YES or NO, respectively. If the user has supplied the surface temperature and air quality data by measurement station, the user sets NEEDAQ equal to YES for the program to interpolate the data along the trajectory. In addition, the user has the option to have the program produce the punched cards containing the METMOD outputs which are used as inputs to other modules of the code. This option is in effect when the user sets KPWIND equal to YES in the data deck. The use of this option is strongly recommended for error free data transfer.

The user next specifies the logical unit number of the peripheral device (LTAPE) where the wind data resides. If the wind data is read from cards, the user specifies LTAPE = 3. For St. Louis applications where the wind, temperature, radiation, and air quality data are retrieved from a tape, the user specifies LTAPE equal to the unit number assigned to the tape.

The program next reads a sequence of variables related to the code's initialization process. These variables are read only for the first case of a run with multiple case inputs. The first variable read in this series of cards is KPSTAT, which is set to YES when the user desires a printed listing of the measurement station names (stored internally). See Section 3.6 for a description of this output.

The program next reads an "extra" output parameter, KXTRA. When KXTRA equals YES, the output includes a listing of which station data are used in the interpolation of wind, temperature, and air quality data. A description of this extra output is included in Section 3.6.

The program next reads the wind speed conversion factor, CONVRT. The user must specify this factor, which is used to convert the user's wind data to the units of kilometers/hour used by the code. In the sample problem, this parameter has a value of 3.6 for conversion of the St. Louis wind data from meters/second to kilometers/hour. If the program reads a negative or zero value for CONVRT, it assumes a value of one.

The next three cards in the first-case-only series contain the grid origin coordinates (UTMXOR, UTMYOR), the minimum grid square sizes (DX, DY), and the number of minimum size grid squares (NX, NY) in the x and y directions. The reader is referred to Section 3.4 for an explanation of these parameters.

The next card read contains the minimum and maximum distances (RMIN and RMAX) used in the interpolation algorithm to screen data from stations very near to and very remote from a trajectory node. See Section 3.10 for a description of these parameters.

At this point the program completes the first-case-only initialization by either reading the wind data from cards or retrieving the winds, surface temperatures, radiation, and pollutant concentrations from tape. See Section 3.3 for a description of these inputs and the user's options.

The program next reads a series of cards with information concerning a specific trajectory. The first card in the series contains the date (IDT) and the start time (XLCL) of the trajectory. The date is specified using the year-month-day (YRMODA) convention. This card is followed by one indicating whether the start time is specified in daylight or standard time. Next, the program reads the start location. The user may specify either a station name (ID) or global x and y coordinates (XS, YS) to define this location. This card is followed by cards specifying the trajectory duration (TTOTAL) and the time interval between trajectory nodes (DTSEG) in hours. Next, the program reads a variable (ITF), which indicates whether the trajectory is to be generated in the forward or backward direction from the start location. This card is followed by one specifying whether the station measurements are to be weighted by reciprocal distances or by the squares of the reciprocal distances in the interpolation formula. The program next reads the number of close stations (up to 3) to be used in the interpolation of station data. The last card in this section specifies the printer-plot boundaries in local coordinates to be used in the trajectory plot. The reader is referred to Section 3.9 for instructions on proper selection of these boundary values.

When these cards have been read and the station measurements retrieved (from cards or tape), the program computes and plots the trajectory.

Following the trajectory generation and interpolation computations, the program calls the eddy diffusivity submodule (when NEEDKZ = YES). The submodule reads the remainder of the input data. The first card read here indicates whether or not the eddy diffusivity input cards should be read. This variable (NKZDAT) must equal YES in a first case data set and may equal NO in subsequent data sets where the same vertical

temperature profiles and mesh point elevations are to be used. When the user specifies NKZDAT equal to YES, the program reads a flag for optional K_z output which determines whether or not the detailed (internal) eddy diffusivity profiles are to be printed. See Section 3.6 for a description of this output.

The eddy diffusivity submodule next reads up to three radiosonde release times and vertical temperature profiles. The input data is structured with one release time card followed by a set of cards with the elevations and temperatures from the sounding. These cards are ordered by ascending elevations until a negative elevation is read. One or two subsequent vertical temperature profiles may be similarly input in chronological order. A negative release time is specified after the final profile to terminate this operation. The user may specify the elevations as either height-above-ground or height-above-sea-level since the program subtracts the first elevation from all elevations prior to using the data.

Next, the program reads a variable (RDTEMP) that determines whether the hourly surface temperatures are to be read. This variable is set equal to NO when the user has supplied temperature data for the measurement stations, and the program has interpolated this data along the trajectory. When the network data has not been interpolated, or when the user wishes to override the interpolated temperatures, this flag is set to YES, and temperatures are read in as follows. The program reads a set of cards, each of which contains a time on the 0-2400 clock and the corresponding surface temperature. These cards are input in chronological order followed by a negative input for the time.

The last set of cards in the input deck specifies the number of vertical mesh points (NOHTS) and their elevations above ground (CELHTS). The program reads NOHTS elevation cards in order by ascending elevation. These vertical mesh parameters are used by all three modules of the code and require careful selection. Instructions are provided in Section 3.5 for their specification.

Finally, the program reads a variable (TERM) equal to MORE or END. When the user has stacked multiple data sets together for multiple trajectory generations, the user specifies MORE as the last card in each data set, except the last set where END is specified.

3.3 Specification of the Station Measurement Data

The manner in which the station measurement data is read by the code depends on the types of information available. In applications where the user has only wind data, the code reads the data from the card reader. In applications for locations such as St. Louis, where each station measures many useful parameters, the program reads the data from a tape. The data format differs significantly for the two cases.

When the wind data is read from the card reader, the code reads a pair of cards for each combination of station and date. The first card in the pair contains the date, station name, and the wind direction data. Columns 1-6 contain the date on the year-month-day basis. Columns 8-15 contain the station name. Columns 29-30 contain the number of compass sectors ($4 \leq n \leq 90$) utilized in the meteorological wind data reporting system under which the program interprets the azimuth data on the remainder of the card. Table 3-2 shows examples of the correspondence between these digitized wind azimuths and their polar angles. The twenty-four wind azimuths are specified in columns 33-80. The second card of the pair contains the digitized wind speed data in columns 33-80. The program assumes the wind data are hourly averages, with the first entry being for the 0000-0100 time period, the next for 0100-0200, etc. Blanks or negative values in the azimuth and speed fields are interpreted as being missing data. The program reads and stores the wind data for the stations for up to ten different dates. The reading of the data is terminated by reading one card with 999999 in columns 1-6.

In cases where a more extensive and disaggregated meteorological data base exists, different retrieval software is used to read, edit, and store the data. Since these types of data bases are rarely structured similarly, the user is encouraged to modify subroutine METIN to suit his/her application, using the St. Louis version as an example.

In the St. Louis application, the station data input file contains measured data for the parameters listed in Table 3-3. As indicated in the table, the program utilizes the wind, temperature, temperature gradient, solar radiation, and pollutant concentration measurements.

The input file is structured with a header record at the beginning of each hour which contains the year, julian day, and hour (0-23). The program converts the user-specified date in the input data deck to the

TABLE 3-2

THE DIGITAL WIND AZIMUTH SYSTEM

	North	East	South	West
Compass Heading of Wind Vector	0°	90°	180°	270°
Point No. (16 pt. system)	8	12	16	4
Point No. (90 pt. system)	45	67	90	22

TABLE 3-3

THE STATION-MEASURED PARAMETERS IN THE ST. LOUIS INPUT FILE

No.		Units	Utilized
1	Wind Speed	m/sec	YES
2	Wind Direction	degrees	YES
3	Temperature	degrees C	YES
4	Dew Point	degrees C	NO
5	Barometric Pressure	mb	NO
6	Temperature Gradient (30m-5m)	ΔT/25m	YES
7	Solar Radiation, All(1)	Langleys/min	YES
8	Solar Radiation, λ>3950A	Langleys/min	YES
9	Solar Radiation, λ>6950A	Langleys/min	NO
10	Solar Radiation, All(2)	Langleys/min	NO
11	Ozone	ppm	YES
12	Carbon Monoxide	ppm	YES
13	Methane	ppm	YES
14	Total Hydrocarbons	ppmc	YES
15	Nitric Oxide	ppm	YES
16	Nitrogen Dioxide	ppm	YES
17	Nitrogen Oxides	ppm	YES
18	Total Sulfur	ppm	YES
19	Hydrogen Sulfide	ppm	NO
20	Sulfur Dioxide	ppm	YES
21	Light Scattering, BSCAT (1)		NO
22	Light Scattering, BSCAT (2)		NO

julian basis and proceeds if there is a match between it and the julian day read from tape. Each hour's data are structured with all the station values for the first parameters, followed by all the station values for the second parameters, etc., with the parameters ordered as shown in Table 3-2. The program reads individual station values for all 22 parameters for the first hour, then proceeds to edit and store the data it needs, then repeats this cycle until all the required data are stored. It is important to note that the program assumes that the station values are input in the order in which the station names are internally stored in subroutine SETIN.

The wind data are stored in the two-dimensional array AWDATA as described below.

AWDATA (1, Station Index)	=	Date (YRMODA)
AWDATA (2-3, Station Index)	=	Station Names
AWDATA (4, Station Index)	=	Number of wind sectors used in azimuth reporting system.
AWDATA (5-28, Station Index)	=	24 hours of wind azimuths
AWDATA (29-52, Station Index)	=	24 hours of wind speeds

These data are edited by placing a -1 in the array where missing data are indicated by very large values in the St. Louis input file. Also, the program makes a special adjustment to the St. Louis wind speed data from nine stations with short measurement towers. The program uses the regional average temperature gradient to infer atmospheric stability and adjusts the ten meter wind speeds to their thirty meter equivalents, using a stability dependent power-law vertical wind profile.

The program obtains the ultraviolet radiation by subtracting the solar radiation for wavelengths greater than 3950A from the total solar radiation data. In this application, the regional hourly average ultraviolet radiation is computed and stored in the array UV. The ultraviolet radiation data are not interpolated along the trajectory since only one-quarter of the stations measure this data.

The surface temperatures and pollutant concentrations are edited for missing data and stored in the array CON. This array is three-dimensional with its indices described as follows:

CON (I, J, K)

I = 1 thru 24 corresponding to hours 0 thru 23

J = station index corresponding to the order in which the stations are specified in subroutine SETIN and read in

K = parameter index

For the St. Louis application, the parameter index corresponds to one of the ten types of data listed in Table 3-4. The user specifies the number of parameters (NOSPEC) stored in CON and an array of the parameter names (CONAM) in data statements described in Section 3.7.

The user can store any data in this array which he/she wants interpolated along the trajectory. The most important data to store in the array are the surface temperature data, which should always be the first parameter (K=1).

3.4 Modeling Region and Emissions Grid

The user determines the geographic extent of the region being modeled. The region should be chosen large enough to allow for 8 to 12 hour trajectories for moderate wind speeds within its boundaries. It must extend far enough to include the major emission sources which affect air quality along the modeled trajectories.

Area source emissions data are compiled and allocated to a rectilinear grid, as described in Section 4. The meteorological module of the code determines a schedule of grid square crossing times and grid indices from the air parcel trajectories, so that the emission module can generate a schedule of emission fluxes along the trajectory. In order to do this, the program must be supplied with the x and y coordinates of the grid origin (UTMXOR, UTMYOR), the number (NX, NY) of fixed-size grid squares in the x and y directions, and the grid square size (DX, DY). In the sample problem, the origin is specified as follows:

UTMXOR =	680.(km)	UTMYOR =	4230.(km)
DX =	1.(km)	DY =	1.(km)
NX =	100	NY =	100

TABLE 3-4

VARIABLES STORED IN THE CON ARRAY

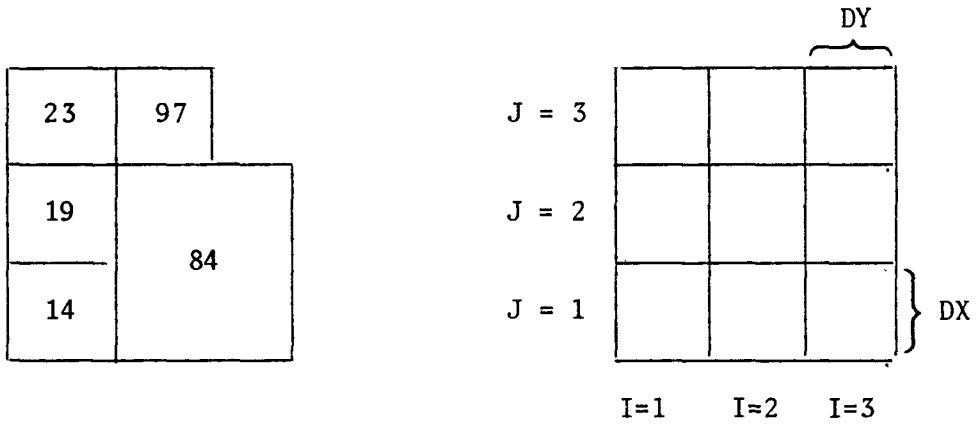
K Index	Parameters	Names
1	Temperature	(TEMP)
2	Ozone	(O3)
3	Carbon Monoxide	(CO)
4	Methane	(CH4)
5	Total Hydrocarbons	(THC)
6	Nitric Oxide	(NO)
7	Nitric Dioxide	(NO2)
8	Nitrogen Oxides	(NOX)
9	Total Sulfur	(TS)
10	Sulfur Dioxide	(SO2)

Thus, the area source emissions grid is a 100 x 100 kilometer region with its southwest corner located at UTM coordinates 680., 4230.

The area source emissions grid usually extends over the entire modeling region, but the program allows it to be a subregion within the modeling region. In other words, if the geographic distribution of significant area source emissions is of limited extent, the user is not constrained to specify an unnecessarily large grid. The user can internally specify (in subroutine EDGE) the width of a buffer zone which is added to all sides of the emission grid. Thus, the modeling region is defined by the area within the boundaries of the emissions grid plus the buffer zone. The program allows trajectories, monitoring stations, and emission point sources to be located anywhere within the modeling region. Nevertheless, the program uses the origin of the emissions grid as the origin of its local coordinate system.

The program has an additional feature that provides the capability of using area source emissions data allocated to variable size grid squares. In this situation, each grid square has a single identifier instead of the usual I and J indices on a fixed grid-square-size grid. This feature of the program is utilized in the St. Louis application where the area source emissions data has been allocated to grid squares ranging in size from 1 x 1 to 10 x 10 kilometers.

When this option is required, the user sets the variable ITGRID equal to YES in a data statement in the block data program. Then, the program calculates the schedule of grid square crossing times and single identifiers using its schedule of crossing times and (I, J) indices plus the IVG array. The IVG array contains the variable-size grid square identifiers for each (I, J) grid square in the emissions grid. As a preprocessing activity, the user must allocate the variable-size grid identifier to a grid with the minimum size squares. This information is then stored in a block data program for use in the code. Figure 3-3 shows an example of the correspondence between the variable-size grid square identifiers and the elements of the IVG array. Note that when there is no grid square identifier (i.e., no area source data) for a particular grid square, the corresponding element of the IVG array is set equal to zero.



IVG(I,J) = identifier of grid square I,J

IVG(1,3) = 23	IVG(2,3) = 97	IVG(3,3) = 0
IVG(1,2) = 19	IVG(2,2) = 84	IVG(3,2) = 84
IVG(1,1) = 14	IVG(2,1) = 84	IVG(3,1) = 84

Figure 3-3 The IVG Array

3.5 Selection of the Vertical Mesh Point Elevations

The vertical mesh point elevations (above ground) are selected on the basis of the meteorology of a particular day being modeled. The user must obtain or estimate the maximum mixing height along the trajectory. If this data is not explicitly reported by the meteorological monitoring network, the user should examine the afternoon vertical temperature profile(s) to find the elevation at which the potential temperature equals the surface potential temperature, and use this elevation as the maximum mixing height (Z_{max}). Typically, the maximum mixing height ranges from 500 to 2000 meters.

The recommended formulae for calcualting the mesh point elevations are provided below.

$$Z_1 = 0.0$$

$$Z_2 = \text{minimum of} \begin{cases} 60 \text{ meters} \\ .10 * Z_{max} \end{cases}$$

$$Z_3 = \text{minimum of} \begin{cases} 200 \text{ meters} \\ .25 * Z_{max} \end{cases}$$

$$Z_4 = \text{minimum of} \begin{cases} 700 \text{ meters} \\ .50 * Z_{max} \end{cases}$$

$$Z_5 = \text{minimum of} \begin{cases} 2000 \text{ meters} \\ Z_{max} \end{cases}$$

Using these formulae will ensure maximum resolution in the lower portion of the air parcel where vertical concentration gradients are most important.

It should be point out that the vertical mesh point elevations used in the sample problem for the second and third mesh points are higher than those calculated from the formulas. This occurrence was an oversight on the part of the modulers and should not be construed as the recommended procedures.

3.6 Description of the Output

This section describes the normal and optional outputs from the meteorological module. In addition to the listing of the input deck produced by PREDAT, the METMOD program generates the normal outputs shown in Figures 3-4 through 3-11, and the optional outputs shown in Figures 3-12 through 3-15.

Figure 3-4 shows the listing for verification of air trajectory calculation inputs. These include the date, start time and location, the trajectory direction, duration, time increment, and the interpolation parameters discussed in Section 3.2. On the same page, the program prints the area source grid description and start point in both global (UTM) and local coordinates. It should be pointed out that the LOCATION OF THE INITIAL POINT will differ from that listed as the START POINT when the program is used to generate backward trajectories. This occurs because backward trajectory nodes are computed and put in chronological order between the two printings.

The trajectory data output by the program is shown in Figure 3-5. Each line of the output describes a trajectory node's time, (I, J) indices on the grid, local coordinates, and the wind velocity and direction used to reach the next trajectory node. The wind velocity is listed in the code's units of kilometers per hour. The direction is indicated by the polar angle (THETA) of the wind vector, measured in degrees counterclockwise from the x-axis. It should be pointed out that this convention for wind vector direction must be used when the user inputs an externally computed trajectory. Each trajectory node in Figure 3-5 has an alphabetic symbol associated with it. These symbols are used in the printer-plot of the trajectory, as shown in Figure 3-6. The nodes are plotted in the local coordinate system with the chronology of the nodes indicated by the alphabetic sequence of the symbols.

Next, the program lists (for several pages) the schedule of grid square crossing times, locations, and (I, J) indices on the fixed-size-grid-square grid. A sample of this output is shown in Figure 3-7. When the variable-size grid square option is used, a schedule of crossing times, locations, and single grid square identifiers is printed, as shown in Figure 3-8. It should be noted that the crossing time is the time at which the air parcel enters the grid square listed. Also, it is worth noting that the sample problem's trajectory is one which starts and ends outside of the emissions grid.

The next page of output contains regional averages of several meteorological parameters calculated for the St. Louis application of the model. Figure 3-9 shows the listing which includes the averaging times, the ultraviolet radiation data, temperature, temperature gradient in the

ERT METEORELOGICAL MODULE -- ST. LOUIS -- 6/29/76

AIR TRAJECTORY CALCULATION DATA

START DATE AND LOCAL TIME AT INITIAL POINT	760629	600 HRS (STANDARD TIME)
LOCATION OF INITIAL POINT	.44	-5.95 (NOT AT A MEASUREMENT STATION)
PRESCRIBED STARTING AZIMUTH AND VELOCITY (OPTIONAL)		
DIRECTION OF TRAJECTORY DEVELOPMENT	FORWARD	
DESIRED TRAJECTORY DURATION	11.0 HRS	
TRAJECTORY SEGMENT LENGTH	1.0 HRS	
MEASUREMENT INTERPOLATION SCHEME	3 CLOSEST STATIONS WITH 1/R**2 WEIGHTING	

GRID REFERENCE DATA

(KILOMETERS)	UTM COORDINATES X	UTM COORDINATES Y	LOCAL COORDINATES X	LOCAL COORDINATES Y
NORTH WEST CORNER	680.00	4330.00	0.00	100.00
NORTH EAST CORNER	780.00	4330.00	100.00	100.00
SOUTH EAST CORNER	780.00	4230.00	100.00	0.00
SOUTH WEST CORNER	680.00	4230.00	0.00	0.00
START POINT	680.44	4224.05	.44	-5.95

Figure 3-4

ERT METEORELOGICAL MODULE -- ST. LOUIS -- 6/29/76

TRAJECTORY DATA

SYMBOL	T(MIN)	I	J	X(KM)	Y(KM)	V(KM/HR)	THETA
A	360.00	1	-4	.44	-5.95	11.09	66.27
B	420.00	5	5	4.91	4.21	8.42	62.38
C	480.00	9	12	8.81	11.67	16.93	45.98
D	540.00	21	24	20.58	23.84	17.18	21.53
E	600.00	37	31	36.56	30.15	15.40	29.04
F	660.00	51	38	50.03	37.63	15.38	25.01
G	720.00	64	45	63.97	44.13	21.63	42.26
H	780.00	80	59	79.97	58.67	18.18	46.72
I	840.00	93	72	92.43	71.91	19.68	30.47
J	900.00	110	82	109.40	81.89	21.53	28.28
K	960.00	129	93	128.35	92.09	24.82	15.54

Figure 3-5

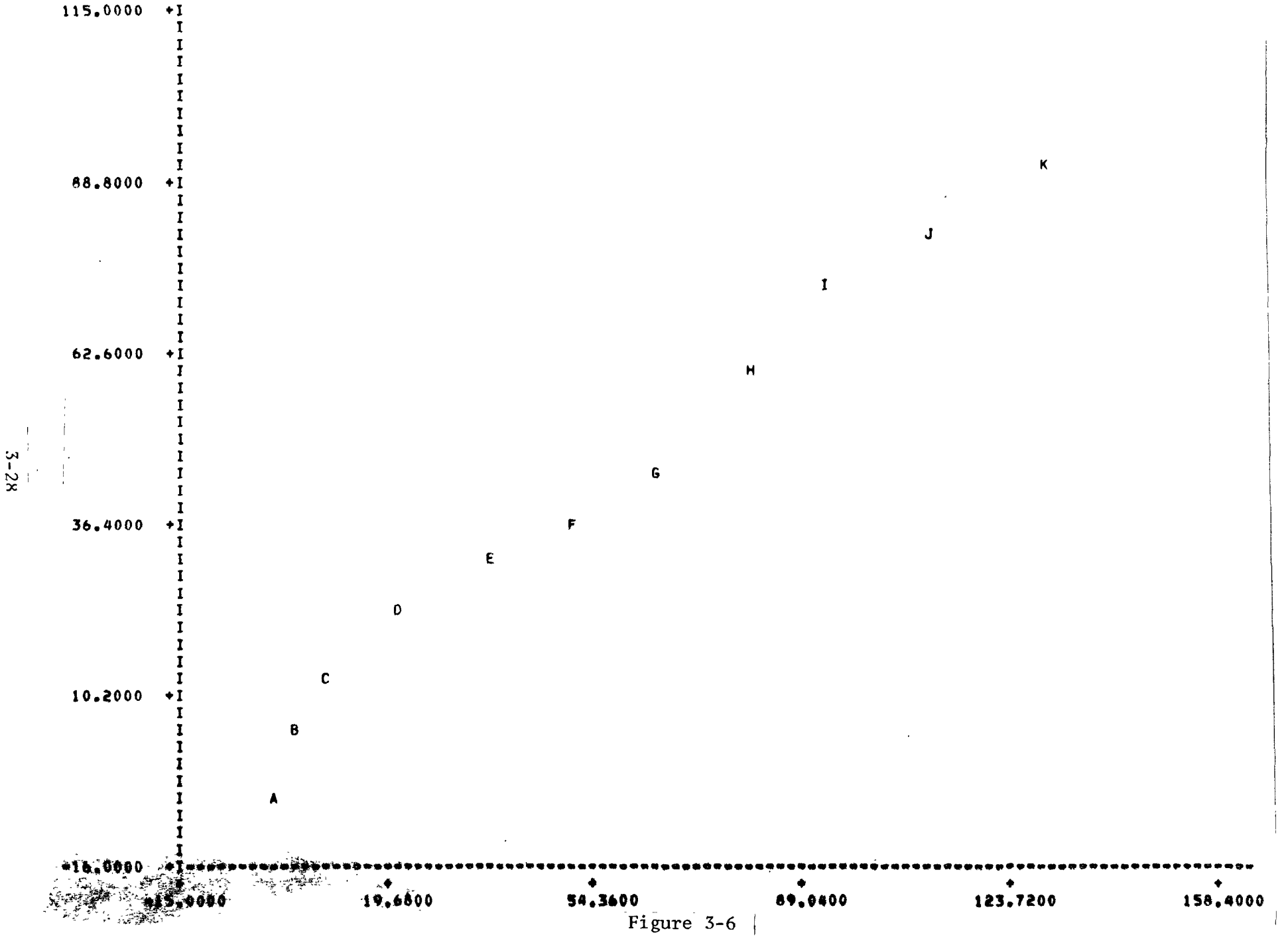

115.0000
88.8000
62.6000
36.4000
10.2000
-16.0000
-15.0000
19.6800
54.3600
89.0400
123.7200
158.4000
A
B
C
D
E
F
G
H
I
J
K

Figure 3-6

ERT METEOROLOGICAL MODULE -- ST. LOUIS --

TC(MIN)	UTM XC(KM)	UTM YC(KM)	IC	JC
360.00	680.44	4224.05	1	-4
367.47	681.00	4225.31	2	-3
371.52	681.30	4226.00	2	-2
380.91	682.00	4227.59	3	-1
383.34	682.18	4228.00	3	0
394.35	683.00	4229.86	4	1
395.15	683.06	4230.00	4	1
401.06	683.50	4231.00	4	2
406.97	683.94	4232.00	4	3
407.79	684.00	4232.14	5	3
412.88	684.38	4233.00	5	4
418.79	684.82	4234.00	5	5
421.40	685.00	4234.38	6	5
426.39	685.32	4235.00	6	6
434.42	685.85	4236.00	6	7
436.77	686.00	4236.29	7	7
442.46	686.37	4237.00	7	8
450.50	686.89	4238.00	7	9
452.13	687.00	4238.20	8	9
458.54	687.42	4239.00	8	10
466.58	687.94	4240.00	8	11
467.50	688.00	4240.11	9	11
474.61	688.46	4241.00	9	12
480.95	689.00	4241.86	10	12
481.63	689.13	4242.00	10	13

Figure 3-7

ERT METEORELOGICAL MODULE -- ST. LOUIS --

TIME (MIN)	UTM-X	UTM-Y	GRID SQUARE ID
360.00	680.44	4224.05	0
394.35	683.00	4229.86	64
421.40	685.00	4234.38	72
426.39	685.32	4235.00	65
486.05	690.00	4242.90	81
496.41	692.03	4245.00	82
511.56	695.00	4248.07	99
521.06	696.86	4250.00	2028
537.06	700.00	4253.25	129
551.01	703.51	4255.00	130
556.60	705.00	4255.59	2039
564.11	707.00	4256.38	2048
570.03	708.58	4257.00	2049
571.62	709.00	4257.17	2058
579.12	711.00	4257.96	2066
586.63	713.00	4258.74	2076
589.06	713.65	4259.00	2077
594.14	715.00	4259.53	2093
598.58	716.18	4260.00	2094
601.95	717.00	4260.39	2115
610.86	719.00	4261.50	2137
614.85	719.89	4262.00	2138
619.77	721.00	4262.61	2159
628.68	723.00	4263.72	2178
630.89	723.50	4264.00	2179

Figure 3-8

ERT METEORELOGICAL MODULE -- ST. LOUIS -- 6/29/76

ST. LOUIS YEAR = 1976 DAY = 181

REGIONAL HOURLY AVERAGES

HOUR	ULTRAVIOLET RADIATION (LANGLEYS/MIN)	TEMPERATURE (DEG C)	TEMPERATURE GRADIENT (DEG C/METER)	CLEARNESS RATIO
0 - 59	0.	21.35	.0099	1.000
100 - 159	0.	21.01	.0178	1.000
200 - 259	0.	20.40	.0125	1.000
300 - 359	0.	20.25	.0037	1.000
400 - 459	0.	20.31	-.0055	1.000
500 - 559	-2.994E-03	20.85	-.0099	1.000
600 - 659	1.095E-02	22.05	-.0203	.549
700 - 759	2.552E-02	23.46	-.0274	.703
800 - 859	4.018E-02	25.33	-.0364	.758
900 - 959	5.035E-02	26.72	-.0359	.725
1000 - 1059	5.786E-02	28.20	-.0366	.695
1100 - 1159	4.903E-02	28.69	-.0330	.545
1200 - 1259	5.129E-02	28.32	-.0339	.568
1300 - 1359	8.542E-02	28.85	-.0432	1.000
1400 - 1459	5.227E-02	28.76	-.0328	.727
1500 - 1559	5.408E-02	28.95	-.0332	.975
1600 - 1659	3.900E-02	29.46	-.0301	1.000
1700 - 1759	1.724E-02	29.24	-.0206	.772
1800 - 1859	1.912E-03	28.08	-.0114	.296
1900 - 1959	0.	25.63	-.0077	1.000
2000 - 2059	0.	23.90	-.0056	1.000
2100 - 2159	0.	22.60	-.0018	1.000
2200 - 2259	0.	21.45	.0015	1.000
2300 - 2359	0.	20.36	.0031	1.000

Figure 3-9

surface layer, and the sky clearness ratios. The clearness ratios are computed as the ratio of the measured ultraviolet radiation to the expected clear-sky ultraviolet radiation for that day and hour.

The next section of output lists the interpolated values of the parameters which have been interpolated along the trajectory. An example of this output for the sample problem is illustrated in Figure 3-10. This output format is repeated for several pages with data for all trajectory node times.

The last page of the normal output, as shown in Figure 3-11, is generated by the eddy diffusivity submodule. It lists the vertical mesh point elevations and the temporal schedules of surface temperature, atmospheric stability class, and average diffusion coefficients as a function of height. The program outputs the stability class data, using the values 1 through 6, which correspond to the Pasquill-Gifford classes A (unstable) through F (very stable).

If the user has specified KPSTAT equal to YES in the input data deck, the program lists the station names and coordinates as shown in Figure 3-12. The x and y coordinates are listed in the local coordinate system even though the user has internally specified them (in SETIN) in global coordinates.

When the user elects the option for extra trajectory output (KXTRA = YES), the program provides a listing of which the station's data are used in the interpolation of wind, temperature, and air quality data along the trajectory. Figure 3-13 shows the extra output from the wind interpolation section of the program. It lists the time, location, interpolated wind speed and azimuth, and the names of the stations from which data were used. When this option is selected, the normal output of temperature and air quality data is superceded by that shown in Figure 3-14. For each trajectory node, the listing includes the qualifying station names and corresponding distances to the stations from the node. Then, for each parameter, the station data and names (in parentheses) are listed, as well as the interpolated value of the parameter. Note that the program substitutes NONE for the station name when there are no stations with valid data.

If the user selects the option for extra output from the eddy diffusivity submodule (KZPRTX = YES), a listing similar to Figure 3-15

```
ERT METEORELOGICAL MODULE     --      ST. LOUIS      --

   TIME = 1000      DATE = 760629

   VARIABLE      INTERPOLATED VALUE

      TEMP             2.7672E+01

      03               4.3641E-02

      CO               2.1552E-01

      CH4              1.4504E+00

      THC              1.4733E+00

      NO               3.3445E-03

      NO2              1.1162E-02

      NOX              1.4244E-02

      TS               1.6972E-02

      SO2              2.5000E-03


   TIME = 1100      DATE = 760629

   VARIABLE      INTERPOLATED VALUE

      TEMP             2.6457E+01

      03               4.4532E-02

      CO               3.6233E-01

      CH4              1.3855E+00

      THC              1.4465E+00

      NO               2.8362E-03

      NO2              6.3903E-03

      NOX              8.1233E-03

      TS               2.9586E-03

      SO2              2.5000E-03
```

Figure 3-10

ERT METEORELOGICAL MODULE -- ST. LOUIS -- 6/29/76

AVERAGE DIFFUSION COEFFICIENTS FOR THE CHEMICAL/DIFFUSION MODULE

MESH POINT ELEVATIONS 0 120 300 600 1200 METERS

TIME	SURFACE TEMPERATURE	STABILITY CLASS	KZ1	KZ2	KZ3 (M**2/MIN)	KZ4	KZ5	KZ6
600	21.5	2	9	9	236	327	126	126
700	22.4	2	41	41	166	140	27	27
800	24.4	2	487	487	389	842	507	507
900	26.5	2	1486	1486	550	867	527	527
1000	27.7	2	1782	1782	6202	9845	1767	1767
1100	26.5	5	137	137	738	1495	363	363
1200	28.5	2	3099	3099	11193	18901	6756	6756
1300	29.6	2	2136	2136	7654	13583	10764	10764
1400	29.4	2	1701	1701	7383	13520	10833	10833
1500	29.3	3	1403	1403	5940	13019	10898	10898
1600	29.7	2	104	104	5353	7509	2520	2520

Figure 3-11

ERT METEORELOGICAL MODULE -- ST. LOUIS

STATIONS AND COORDINATES

RAMS 101	64.183	49.862
RAMS 102	62.518	56.045
RAMS 103	67.588	52.467
RAMS 104	67.312	47.304
RAMS 105	63.706	46.453
RAMS 106	58.660	47.566
RAMS 107	60.179	52.610
RAMS 108	68.407	61.102
RAMS 109	75.802	49.886
RAMS 110	67.209	42.826
RAMS 111	58.812	42.479
RAMS 112	53.938	50.913
RAMS 113	57.738	59.820
RAMS 114	64.320	67.456
RAMS 115	77.111	67.799
RAMS 116	82.777	60.083
RAMS 117	80.560	42.818
RAMS 118	63.065	33.256
RAMS 119	49.759	40.547
RAMS 120	43.079	55.909
RAMS 121	52.414	72.376
RAMS 122	61.631	99.223
RAMS 123	97.320	56.378
RAMS 124	69.275	6.537
RAMS 125	17.445	52.240

Figure 3-12

OLD TIME	DATE	X	Y	NEW TIME	DATE	VELOCITY	AZIMUTH	STATIONS		
600	760629	.44	-5.95	600	760629	11.09	66.27	125	119	124
700	760629	4.91	4.21	700	760629	8.42	62.38	125	119	120
800	760629	8.81	11.67	800	760629	16.93	45.98	125	119	120
900	760629	20.58	23.84	900	760629	17.18	21.53	125	119	120
1000	760629	36.56	30.15	1000	760629	15.40	29.04	119	111	120
1100	760629	50.03	37.63	1100	760629	15.38	25.01	119	111	106
1200	760629	63.97	44.13	1200	760629	21.63	42.26	105	110	104
1300	760629	79.97	58.67	1300	760629	18.18	46.72	116	115	109
1400	760629	92.43	71.91	1400	760629	19.68	30.47	116	115	123
1500	760629	109.40	81.89	1500	760629	21.53	28.28	123	116	115
1600	760629	128.35	92.09	1600	760629	24.82	15.54	123	116	115

Figure 3-13

```
ERT METEOROLOGICAL MODULE      --      ST. LOUIS      --      6/29/76

 TIME =  1000      STATION NUMBERS        119   111   120   118   112
                                          106   125   105   107   110
                                          101   104   113   102   103
                                          124   109   108   121   117
                                          114   116   115   123

 DATE = 760629     DISTANCES (KM)       16.80 25.44 26.57 26.68 27.08
                                        28.14 29.21 31.66 32.59 33.17
                                        33.93 35.21 36.45 36.66 38.22
                                        40.34 43.92 44.41 45.10 45.79
                                        46.50 55.06 55.33 66.18

 VARIABLES      X1     (STAT)      X2     (STAT)      X3     (STAT)  X-INTERPOLATED
  TEMP      2.70E+01 ( 119)  2.83E+01 ( 111)  2.86E+01 ( 120)      2.7672E+01
  O3        4.23E-02 ( 119)  4.29E-02 ( 111)  4.79E-02 ( 120)      4.3641E-02
  CO        2.29E-01 ( 119)  3.27E-01 ( 111)  5.89E-02 ( 120)      2.1552E-01
  CH4       1.38E+00 ( 119)  1.50E+00 ( 111)  1.57E+00 ( 120)      1.4504E+00
  THC       1.43E+00 ( 119)  1.46E+00 ( 111)  1.59E+00 ( 120)      1.4733E+00
  NO        3.88E-03 ( 119)  2.89E-03 ( 111)  2.50E-03 ( 120)      3.3445E-03
  NO2       1.14E-02 ( 119)  1.21E-02 ( 111)  9.50E-03 ( 120)      1.1162E-02
  NOX       1.49E-02 ( 119)  1.49E-02 ( 111)  1.19E-02 ( 120)      1.4244E-02
  TS        2.26E-02 ( 119)  1.73E-02 ( 111)  2.50E-03 ( 120)      1.6972E-02
  SO2       2.50E-03 ( 120)  2.50E-03 ( 108) -1.00E+00 (NONE)      2.5000E-03

 TIME =  1100      STATION NUMBERS        119   111   106   118   112
                                          105   110   107   101   120
                                          104   102   103   113   109
                                          108   117   114   121   125
                                          124   116   115   123

 DATE = 760629     DISTANCES (KM)        2.93 10.03 13.16 13.75 13.85
                                        16.28 17.95 18.10 18.71 19.56
                                        19.81 22.25 22.99 23.49 28.54
                                        29.81 30.97 33.08 34.83 35.71
                                        36.57 39.71 40.54 50.87

 VARIABLES      X1     (STAT)      X2     (STAT)      X3     (STAT)  X-INTERPOLATED
  TEMP      2.63E+01 ( 119)  2.66E+01 ( 111)  2.85E+01 ( 106)      2.6457E+01
  O3        4.49E-02 ( 119)  3.88E-02 ( 111)  4.71E-02 ( 106)      4.4532E-02
  CO        3.65E-01 ( 119)  3.73E-01 ( 111)  2.92E-01 ( 106)      3.6233E-01
  CH4       1.37E+00 ( 119)  1.52E+00 ( 111)  1.55E+00 ( 106)      1.3855E+00
  THC       1.44E+00 ( 119)  1.44E+00 ( 111)  1.61E+00 ( 106)      1.4465E+00
  NO        2.88E-03 ( 119)  2.50E-03 ( 111)  2.50E-03 ( 106)      2.8362E-03
  NO2       5.64E-03 ( 119)  1.10E-02 ( 111)  1.37E-02 ( 106)      6.3903E-03
  NOX       7.27E-03 ( 119)  1.35E-02 ( 111)  1.62E-02 ( 106)      8.1233E-03
  TS        2.50E-03 ( 119)  8.57E-03 ( 111)  2.50E-03 ( 118)      2.9586E-03
  SO2       2.50E-03 ( 108) -1.00E+00 (NONE) -1.00E+00 (NONE)      2.5000E-03
```

Figure 3-14

ERT METEORELOGICAL MODULE -- ST. LOUIS -- 6/29/76

TIME 1300 LST

VERTICAL EDDY DIFFUSIVITY EVERY 20 METERS

HEIGHT	KZ	HEIGHT	KZ	HEIGHT	KZ	HEIGHT	KZ	HEIGHT	KZ
20	490.0	620	15794.4	1220	6.0	1820	6.0	2420	6.0
40	1151.7	640	15872.5	1240	6.0	1840	6.0	2440	6.0
60	1895.8	660	15921.7	1260	6.0	1860	6.0	2460	6.0
80	3151.0	680	15941.4	1280	6.0	1880	6.0	2480	6.0
100	3992.9	700	15931.0	1300	6.0	1900	6.0	2500	6.0
120	4804.4	720	15889.6	1320	6.0	1920	6.0	2520	6.0
140	5585.7	740	15816.4	1340	6.0	1940	6.0	2540	6.0
160	6337.2	760	15709.9	1360	6.0	1960	6.0	2560	6.0
180	7059.0	780	15568.7	1380	6.0	1980	6.0	2580	6.0
200	7751.3	800	15390.9	1400	6.0	2000	27.1	2600	6.0
220	8414.5	820	15174.1	1420	6.0	2020	17.1	2620	6.0
240	9048.7	840	14915.4	1440	6.0	2040	9.1	2640	6.0
260	9654.2	860	14611.0	1460	6.0	2060	6.0	2660	6.0
280	10231.1	880	14255.8	1480	6.0	2080	6.0	2680	6.0
300	10779.6	900	13843.4	1500	6.0	2100	6.0	2700	6.0
320	11300.0	920	13364.7	1520	6.0	2120	6.0	2720	6.0
340	11792.4	940	12806.9	1540	6.0	2140	6.0	2740	6.0
360	12256.9	960	12151.1	1560	6.0	2160	6.0	2760	6.0
380	12693.6	980	11367.1	1580	6.0	2180	6.0	2780	6.0
400	13102.8	1000	10402.2	1600	6.0	2200	6.0	2800	6.0
420	13484.5	1020	9148.5	1620	6.0	2220	6.0	2820	6.0
440	13838.8	1040	7307.2	1640	6.0	2240	6.0	2840	6.0
460	14165.8	1060	6.0	1660	6.0	2260	6.0	2860	6.0
480	14465.4	1080	6.0	1680	6.0	2280	6.0	2880	6.0
500	14737.8	1100	6.0	1700	6.0	2300	6.0	2900	6.0
520	14982.8	1120	6.0	1720	6.0	2320	6.0	2920	6.0
540	15200.5	1140	6.0	1740	6.0	2340	6.0	2940	6.0
560	15390.7	1160	6.0	1760	6.0	2360	6.0	2960	6.0
580	15553.2	1180	6.0	1780	6.0	2380	6.0	2980	6.0
600	15687.8	1200	6.0	1800	6.0	2400	6.0	3000	6.0

AVERAGE DIFFUSION COEFFICIENTS FOR THE CHEMICAL/DIFFUSION MODULE

TIME	SURFACE TEMPERATURE	STABILITY CLASS	KZ1	KZ2	KZ3 (M**2/MIN)	KZ4	KZ5	KZ6
1300	29.6	2	2136	2136	7654	13583	10764	10764

Figure 3-15

is generated for each hour. This figure shows the detailed (internal) eddy diffusivity coefficients as a function of elevation. Each hour's vertical profile is printed prior to the hour's average diffusion coefficients.

When the user selects the option for punched output (KPWIND = YES), the user obtains card images punched on logical unit number 1 with the following data:

1) the page title
2) trajectory start location in global coordinates
3) STANDARD or DAYLIGHT time card
4) trajectory node times, wind velocities and directions
5) trajectory grid square crossing times, coordinates, and either (I, J) indices or single identifiers
6) schedule of sky clearness ratios
7) schedule of surface temperatures
8) schedule of average diffusivity coefficients
9) schedule of atmospheric stability classes.

These card-images are punched in the order listed above with all references to time made on the 0-1440 minutes/day clock.

3.7 Internally Specified Data

The METMOD program requires internal specification of certain parameters specific to a particular modeling region. These parameters are designated using either FORTRAN DATA statements or FORTRAN replacement statements. Table 3-5 lists the code variable names, their functions, and the subroutines in which they are specified. In addition, the FORTRAN source listing contains notes to the user where internal specification of data is required.

3.8 Surface Roughness Lengths

The eddy diffusivity submodule requires the specification of surface roughness length parameters. Figure 3-16 illustrates the surface roughness profile used in the model. This profile has been designed for

TABLE 3-5

METEOROLOGICAL MODULE INTERNALLY SPECIFIED DATA

Variable Name	Function	Subroutine
NUMSTA	Number of measurement stations (wind, temperature, radiation, etc.)	SETIN
ISTANS	The measurement station names. A vector dimensioned 2 x NUMSTA which contains two 4-character Hollerith words for each station. The second word for each name must be unique.	SETIN
SSTAN	The measurement station locations. A vector dimensioned 2 x NUMSTA which contains the global (UTM) horizontal and vertical coordinates of the stations. (kilometers)	SETIN
RLAT	Latitude of the approximate center of the modeling region. Used in solar zenith angle computation. (degrees)	SETIN
RLONG	Longitude of approximate center of the modeling region. Used in solar zenith angle computation. (degrees)	SETIN
TMZONE	Time zone of the modeling region. Used in the solar zenith angle computation. Use 6. for St. Louis, which represents the difference between local time and Greenwich, England time.	SETIN
NUMBAS	Number of geographic terrain barriers specified. (max = 20)	BARIER
BAS	The global (UTM) end point coordinates of terrain barrier in the modeling region. A vector dimensioned 4 x NUMBAS which contains the x_1, y_1, x_2, y_2 coordinates of a line segment representing a physical barrier to the wind field. (kilometers)	BARIER
BUFZON	The width of a buffer zone outside of emissions grid. Backward trajectories are not permitted to back off the grid more than this distance. (kilometers)	EDGE
ZW	Elevation of wind measurement instrument above ground. (meters)	EDDY

TABLE 3-5 (CONTINUED)

Variable Name	Function	Subroutine
UTMXUR, UTMYUR	Coordinates (x,y) of the center of the urban roughness length region. (UTM = kilometers)	RUFNES
ZOURBN	The urban surface roughness length. (meters)	RUFNES
ZORURL	The rural surface roughness length. (meters)	RUFNES
RADURB	Radius of urban surface roughness length region. (kilometers)	RUFNES
RADSUB	Radius to rural surface roughness length region. (kilometers)	RUFNES
NSTMAX	The maximum number of measurement stations for which the program's arrays are dimensioned.	METIN
NOSPEC	Number of variables stored in the CON array for interpolation along the trajectory. (max = 10)	METIN
CONAM	A vector of 4-character Hollerith words which contains the names of the variables in the CON array.	METIN
ITGRID	A Hollerith variable equal to YES when the variable size grid square feature of the program is to be used.	BLOCK DATA
VGRID	A two-dimensional array which contains the variable size grid square identifiers corresponding to a grid with uniform size grid squares. (See Section 3.4)	BLOCK DATA

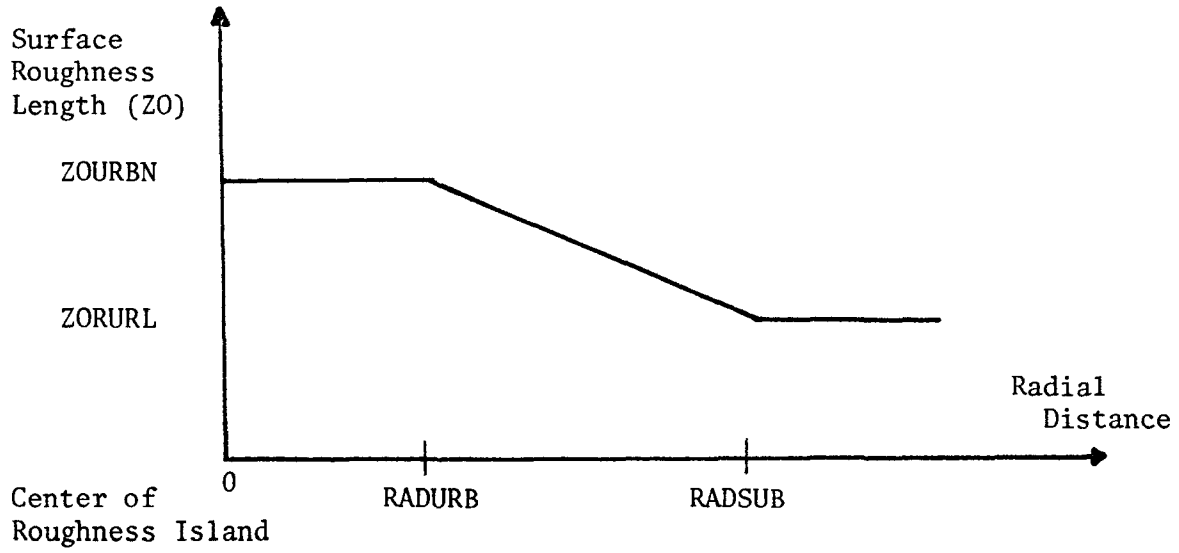


Figure 3-16 Surface Roughness Length Profile

modeling a region with a surface roughness island like St. Louis. The program assumes there is a circular subregion with a district surface roughness length ZOURBN and an outlying region with surface roughness length ZORURL. The user specifies these two roughness lengths in meters, the global x and y coordinates of the center of the subregion, and the radii of the inner (urban) and outer (suburban) regions, RADURB and RADSUB, respectively. The program assumes that the surface roughness varies linearly with radial distance in the region between the inner and outer radii. These parameters are specified in subroutine RUFNES, as described in Section 3.8.

For applications without a surface roughness island, the user can specify ZOURBN equal to ZORURL. Similarly, the user may have an application where an inverted profile (ZOURBN < ZORURL) is appropriate, such as a heavily forested outlying region. In any case, the user must choose RADURB not equal to RADSUB for proper operation of the code.

3.9 Printer-Plot Boundaries

The printer-plot boundaries specified in the input deck may focus on smaller areas than the complete modeling region. The parameters XL, XR, YB, and YT are, respectively, the local x-coordinates of the left and right (east) boundaries of the plotted areas, and the local y-coordinates of the bottom and top (north) boundaries of the plotted area. These parameters are input in kilometers.

Because of the constraints involved in using a line printer to produce a plot (fixed at 51 horizontal lines and 101 characters in this case), certain rules of thumb should be followed in selecting the plot boundaries in order to obtain undistorted representations. Since character spaces are 0.6 times as large as line spaces, true-to-scale plots can be obtained only by maintaining a ratio of .84:1 between the character position scale and the print line scale. Thus, the user is instructed to choose XR, XL, YB, and YT such that

$$\frac{YT - YB}{XR - XL} \simeq .84$$

to obtain undistorted plots of the trajectories.

3.10 Computational Procedures

3.10.1 Calculation of Air Parcel Trajectories

In its simplest form, the method, used by the program to obtain a wind vector at some given point and time, consists of constructing the weighted average of the wind speed and azimuth measurements recorded at up to three neighboring stations. However, the application of this method in practical situations requires introduction of some additional procedures to avoid various pitfalls. In particular, procedures are needed to deal with incomplete measurement records, stations supplying conflicting information from the same relative azimuth, stations screened from the trajectory node by geographic barriers, and stations very near to or very remote from a node. To clarify the ensuing discussion of forward and backward trajectory development, we list here the definitions of seven terms related to these supplementary procedures. The names of the subroutines performing the functions are also noted where appropriate.

1) DATA EXTRAPOLATION: The longest gaps in the data record that will be filled are those with two adjacent missing points embedded in or on the end of a one-day record. These gaps can only be filled (by linear interpolation or extrapolation) if the two adjacent points on one side or the other are non-zero. (Subroutine EXTRP)

2) DATA AVAILABILITY: Data is considered available only if it is recorded or can be extrapolated.

3) ANGLE TEST: Wind data will not be accepted from two stations lying within 2° azimuth of each other as seen from the trajectory node being considered. (Subroutine ANGTST)

4) BARRIER TEST: Wind data will only be accepted from stations which can be connected to the trajectory node by a straight line which does not intersect any of the input barrier line segments. (Subroutine BARIER)

5) VALID DATA: To be valid, a wind measurement must be available and satisfy the angle and barrier test.

6) CLOSE STATIONS: If a station is within a minimum distance (RMIN) from a particular trajectory node, the program uses only the data from this station. When there are no stations within the minimum distance, or when a station within the minimum distance has no data available, the program searches for data from up to three neighboring stations. It uses the measurements from the neighboring close stations, provided they are located within the maximum distance, RMAX, from the node.

7) EDGE TEST: Development of a backward air trajectory may be terminated early if its path takes it out side the modeling region. (Subroutine EDGE)

Having established this background of terms, we are ready to describe the calculation sequences for forward and reverse trajectory development. Forward trajectory development consists of cyclic execution of the following steps:

1) Search the wind data of the stations in order of increasing distance from the trajectory point until the desired number of close stations having valid data are found, or RMAX is reached. These operations are accomplished in subroutine GETAZV. Lacking any close stations, the trajectory calculation is terminated.

2) Combine the n wind speed/azimuth pairs found into an average speed and azimuth using the interpolation formula

$$\bar{S} = \frac{\sum_{i=1}^{n} \frac{1}{R_i^k} S_i}{\sum_{i=1}^{n} \frac{1}{R_i^k}} \qquad (3\text{-}1)$$

where i = station index

n = number of close stations to be used ($1 \leq n \leq 3$, as specified by the user)

k = 1 or 2 as specified by the user

R_i = distance from trajectory point to i^{th} station

S_i = speed or angle measurement from i^{th} station

$\overline{S}$ = weighted average speed or azimuth

This calculation is performed in subroutine AZVDIS.

3) Integrate the trajectory one time step using Euler's method (i.e., using the current position and velocity to extrapolate the trajectory to the next node in a single step).

4) Unless the trajectory is complete, and provided the barrier and edge tests permit, start again at step (1) using the time for the new node.

Reverse trajectory development is somewhat more complicated, but does utilize many of the calculations needed in the forward case. Reverse trajectories are calculated in the following manner:

1) Search the station measurement data using the current trajectory point, using the time that will be appropriate at the next node (that being one time step earlier).

2) Combine the measurements.

3) Extrapolate the trajectory back one time step, using the current position and the velocity calculated for one time step earlier, to obtain an estimate of the next earlier node position (see Figure 3-17a).

4) Using the estimated node position, search the station data with the same time for a new set of close station data.

5) Combine the measurements.

6) Starting from the reference node, reestimate the position of the next earlier node by extrapolating back one time step with the new wind vector (see Figure 3-17b).

7) Repeat steps 4), 5), and 6) as desired (program currently does three iterations, see Figure 3-17c).

8) Barrier and edge test permitting, establish the last estimate of the next earlier node as the new reference node.

9) Record the last calculated wind vector with the time appropriate to the new node.

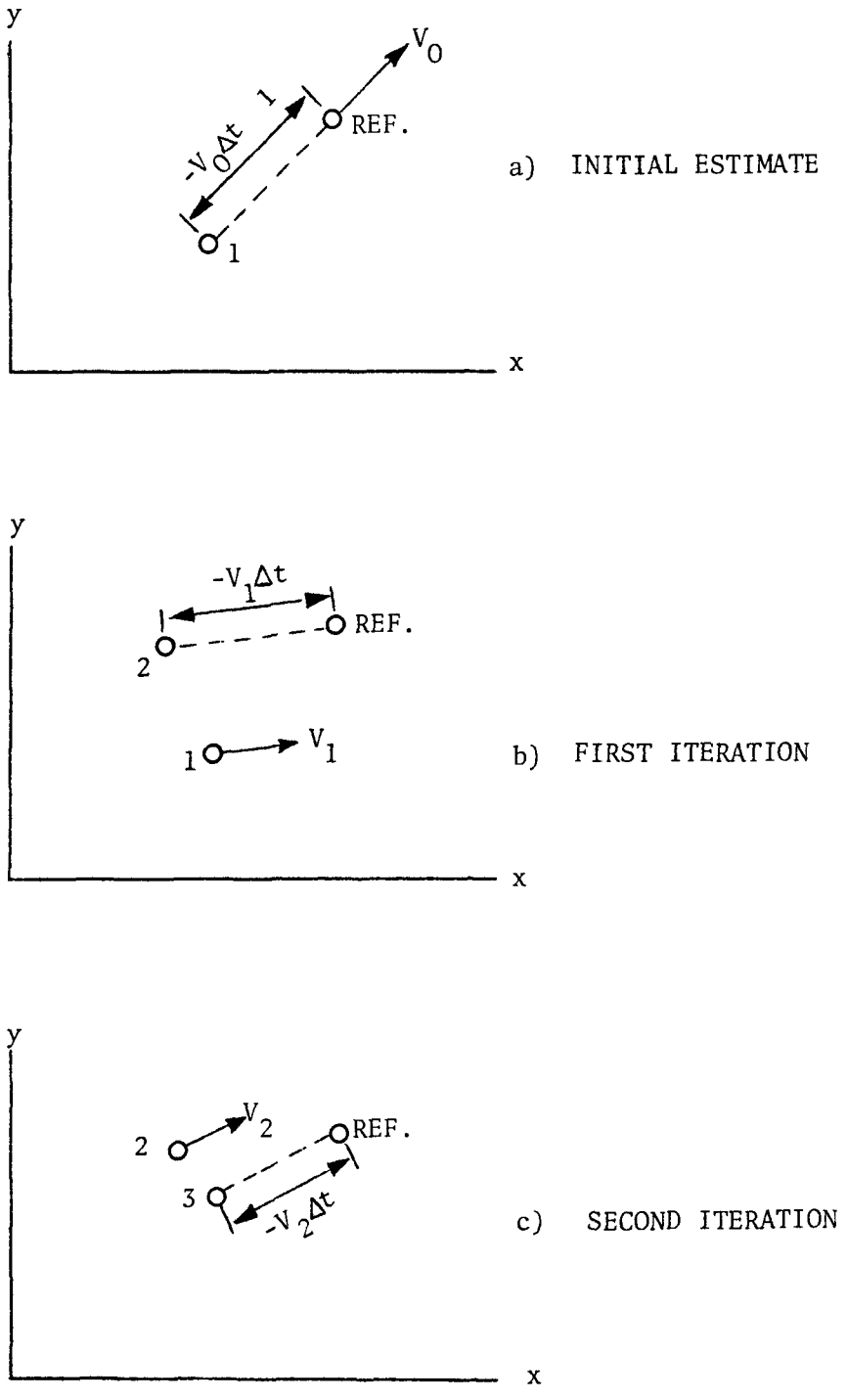


Figure 3-17 Reverse Trajectory Development

10) Return to step 1), unless the desired trajectory has been completed.

When a reverse trajectory has been completed, subroutine WINDY reorders the table of times and wind vectors, placing them in chronological order, and records the final calculated point as the trajectory start point.

3.10.2 Interpolation of Other Atmospheric Parameters Along the Trajectory

The code calculates the surface temperature and air quality data at trajectory nodes in a manner very similar to that used to obtain wind vectors. The program searches the measurement records of close stations for each parameter and interpolates the valid data (from up to three stations) using the Equation (3-1). The principal difference between the screening of these data and the wind data is that these data are not subjected to the angle test or barrier test. The rules for extrapolation of data and determination of close stations are identical to those described in the previous section. The calculations are made in subroutine GETCON.

3.10.3 Calculation of Eddy Diffusivity Coefficients

The computation of vertical eddy diffusivity coefficients (K_z) occurs after the generation of air trajectory and the interpolation of atmospheric parameters. The trajectory wind speeds, trajectory node locations, and the interpolated surface temperatures are passed to subroutine EDDY, which controls the generation of the K_z schedules. This subroutine calls KZDATA to read the vertical temperature data and the vertical mesh point elevations.

This subprogram's computation cycle employs an outer loop for each vertical temperature profile and a nested (inner) loop for calculation of hourly K_z profiles between sounding release times.

For each cycle through the outer loop, the program partitions the vertical temperature structure into (up to 30) layers with distinct lapse rates. Temperature data points which indicate very minor variations in the lapse rate are removed from the vertical profile by subroutine SMOOTH.

Given the edited vertical temperatures and other data mentioned above, the program proceeds to the inner loop. The first computation within the loop involves updating the vertical temperature profile if the soundings release time precedes the trajectory start time (or current time). The update is performed using the incremental difference between the current hour's surface temperature and that of the previous hour or the sounding, whichever is closer. The lapse rate used to update the sounding is determined from the following rules:

1) If the increment in surface temperature is positive, the program uses the greater of the lapse rates in the previous vertical sounding, the next vertical sounding, or the dry adiabetic lapse rate.

2) If the increment in surface temperature is negative, the program uses the lapse rate indicated by the surface temperature and first elevated temperature.

Once the sounding has been updated, the program calculates the surface roughness length for the particular trajectory location. Now the code has all the information required to calculate the current hour's K_z profile. The stability of each atmospheric layer with a distinct lapse rate is determined. In addition, the atmospheric stability class of the surface layer is computed for use in the emissions module. Within each layer, the K_z values are calculated at either 10 or 20 meter vertical increments, using different K_z formulations for within and above the surface layer. Once this detailed K_z profile is computed, the program calculates a set of average diffusion coefficients (averaged between the mesh points) for use in the chemical-diffusion model.

At this point, the program remains in the inner loop (i.e., updating the sounding and calculating K_z profiles) until a new sounding time is reached. Then, control is passed to the outer loop, where the entire process is repeated until the final trajectory time is reached.

3.11 Program Modifications

3.11.1 Changing the Number of Measurement Stations

The variable NUMSTA located in common block /WDATA/ is the number of stations recognized by the program. Many arrays in the program have

dimensions which must be equal or larger than NUMSTA. Currently, NUMSTA and these dimensions are set equal to 25 for the St. Louis application of the model. If the user desires to use N stations (N > 25), the following arrays should be redimensioned:

In all subroutines with common blocks:

```
COMMON/AIRQAL/ ....  CON(24,N,10) ....
COMMON/WDATA/  .... SSTAN(2,N), ISTANS(2,N) ...
COMMON/WINFLD/ .... AWDATA(52,N), DISAVE(N), IDSAVE(N)
```

In subroutines:

```
GETAZV   DIS(N), IDIS(N), DDIS(N)
GETCON   IWORK(N), WORK(N)
METIN    NWDATA(52,N), X(22,N), WORK(22,N)
```

In addition, NUMSTA and NSTMAX should be set equal to N in subroutines SETIN and METIN, respectively. See Appendix A for a list of subroutines with the common blocks mentioned above.

3.11.2 Changing the Station Azimuth Exclusion Sector

Currently, data will not be accepted from two stations lying within 2° azimuth of each other, as seen from the trajectory node being considered. The size of this exclusion angle is determined by the variable COSTST set in a DATA statement in subroutine ANGTST. COSTST should be set equal to the cosine of the desired exclusion angle. Note that since the exclusion angle is applied on both sides of the azimuth to a previously accepted station, the sector actually excluded is twice the specified angle.

4. THE EMISSIONS MODULE

This section provides instructions for proper use of the emissions module, program EMMOD. The function of the program is to process data from a region's emissions inventory to produce a tabulation of emissions along a trajectory for use in the chemical-diffusion module. The program requires regionally disaggregated emissions data for nitrogen oxides, sulfur oxides, carbon monoxide, and four classes of reactive hydrocarbon species: alkanes (paraffins), alkenes (olefins), aromatics, and aldehydes.

Program EMMOD is designed to utilize emission inventories which have been preprocessed into modeling format, a format in which the contributions of individual source types (mobile and stationary) have been summed into single hourly area source emission rates for each grid square. In addition, the program considers the major emission point sources on an individual basis. The inputs to the program include a small file of control parameters and trajectory data read from the card reader. Also included are three tape files, which contain the area source grid description; the hourly area source emission rates; and, the point source locations, hourly emission rates, and stack parameters.

Portions of the program are designed specifically for the format of the RAPS/St. Louis emissions tapes. The user should anticipate making significant modifications to the code for applications involving regions other than St. Louis. These modifications are necessary not only because of the different formats of emission data in different regions, but also in the interest of efficient processing of the data.

4.1 Control Parameters and the Input Deck

The input deck parameters will be discussed in the general order in which they are input to the code. The reader should refer to several figures and a table, as described below, to obtain a better understanding of the input data and format. Figure 4-1 illustrates the listing of the input deck produced by subroutine PREDAT for the sample problem. Table 4-1 lists the inputs by their code variable names, and describes their functions and formats. Figure 4-2 contains a flow chart of the input variable sequence and options. These aids will be helpful in formulating input decks for the program.

TABLE 4-1

EMISSIONS MODULE INPUT VARIABLE DESCRIPTION

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
1	TITLE	Label for each page of output.	(20A4)	May contain any user comments in columns 1-80.
2	NDAREA	Area source emission schedules are to be computed (YES or NO).	(40X,A4)	This is a 4-character Hollerith variable equal to YES or NO starting in column 41.
3	NDPONT	Point source emission schedules are to be computed (YES or NO).	(40X,A4)	
4	NDSUMS	Regional sums of emissions each hour are to be computed (YES or NO).	(40X,A4)	
5	IPUNCH	Outputs required as input for the chemical-diffusion module are to be punched (YES or NO).	(40X,A4)	
6	LUGRID, LUAREA, LUPONT	Logical unit number of area source grid file, area source emissions file, and point source emissions file.	(40X,I2)	(Three cards)
7	NFLXIN	Number of input emission rates.	(40X,I2)	
8	NFXOUT	Number of output emission rates.	(40X,I2)	
9	NAMIN, ADJUST	Hollerith names and conversion factors of input emission species.	(40X,A4,6X, F12.0)	(NFLXIN cards)

TABLE 4-1 (CONTINUED)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
10	NAMOUT	Hollerith names of output emission species	(40X,A4,6X, F10.0)	(NFXOUT cards)
11	PLENTH, PWIDTH	Length and width of the air parcel in kilometers.	(40X,F10.0)	Must correspond to the minimum grid square size. (Two cards)
12	EMABAS	Default area source emission rates for the region outside grid.	(40X,F10.0)	(NFXOUT cards)
13	TIMEHS, UTMAS, IDHS	Grid square crossing times, UTM coordinates, and corresponding grid square ID numbers.	(40X,3F10.0, I10)	(Output from METMOD)
14	UTMXOR, UTMYOR	Global x and y coordinates of local coordinate system origin in kilometers.	(40X,2F10.0)	
15	P(1,1), P(2,1)	Trajectory start point global x and y coordinates in kilometers.	(40X,2F10.0)	
16	T, V, TH	Time, wind velocity, and azimuth of trajectory nodes.	(40X,3F10.0)	Time in minutes, velocity in km/hr, and azimuth in degrees counterclockwise from east (output from METMOD).
17	NOSTAT	Number of vertical mesh points.	(40X,I10)	
18	ZEE	Elevations of vertical mesh points in air parcel in meters.	(40X,F10.0)	(NOSTAT cards)
19	TMTEMP, TEMPSF	Times and values of surface temperatures.	(40X,2F10.0)	Time in minutes and temperature in degrees centigrade.

TABLE 4-1 (CONTINUED)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
20	TMSTAB, KSTABL	Times and values of atmospheric stability class.	(40X,2F10.0)	Time in minutes and stability class A-F are specified as 1-6.
21	TMXHIT, HTMIXL	Schedule of times and corresponding mixing layer heights or inversion heights, whichever is greater.	(40X,2F10.0)	
22	TERM	Last card which has the value of MORE or END.		Note special JCL is needed for multiple runs.

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
   1  ERT EMISSIONS MODULE      --      ST. LOUIS      --      6/29/76   (181)
   2  DO WE NEED AREA SOURCE EMISSIONS          YES
   3  DO WE NEED POINT SOURCE EMISSIONS         YES
   4  DO WE NEED REGIONAL SUMS                  YES
   5  DO WE NEED PUNCHED OUTPUT                 YES
   6  LOGICAL UNIT OF GRID DESCRIPTION          10
   7  LOGICAL UNIT OF AREA SOURCE EMISSIONS     11
   8  LOGICAL UNIT OF POINT SOURCE EMISSIONS    12
   9  NUMBER OF INPUT VARIABLES                 11
  10  NUMBER OF OUTPUT VARIABLES                07

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  11  INPUT VARIABLE NAMES,ADJUSTMENT           PART      1.0
  12  2                                         SOX       15.625      FROM KG TO MOLES
  13  3                                         NOX       21.739      TO MOLES NO
  14  4                                         THC       25.0
  15  5                                         CO        35.714      TO MOLES CO
  16  6                                         DUST      1.0
  17  7                                         NRHC      1.0
  18  8                                         PARF      1.0
  19  9                                         OLEF      1.0
  20  10                                        AROM      1.0

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  21  11                                        ALDE      1.0
  22  OUTPUT VARIABLE NAMES,                    NOX
  23  2                                         PARF
  24  3                                         OLEF
  25  4                                         AROM
  26  5                                         ALDE
  27  6                                         CO
  28  7                                         SOX
  29  WIDTH OF THE AIR PARCEL     (KM)          1.0
  30  LENGTH OF THE AIR PARCEL    (KM)          1.0

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  31  DEFAULT AREA SOURCE RATES   NOX             0.0E 00     MOLES PER HOUR
  32  2                           PARF            0.0E 00
  33  3                           OLEF            0.0E 00
  34  4                           AROM            0.0E 00
  35  5                           ALDE            0.0E 00
  36  6                           CO              0.0E 00
  37  7                           SOX             0.0E 00
  38  TM,X,Y,ID   360.00    680.44    4224.05          0
  39  2           394.40    683.00    4229.87         64
  40  3           421.46    685.32    4234.39         72
```

Figure 4-1 Emissions Module Sample Problem Input Listing

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  41  4          426.39    685.32    4235.00         65
  42  5          486.07    690.00    4242.90         81
  43  6          496.41    692.03    4245.00         82
  44  7          511.58    695.00    4248.08         99
  45  8          521.06    696.86    4250.00       2028
  46  9          537.08    700.00    4253.25        129
  47  10         551.01    703.50    4255.00        130
  48  11         556.62    705.00    4255.59       2039
  49  12         564.13    707.00    4256.38       2048
  50  13         570.04    708.57    4257.00       2049

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  51  14         571.63    709.00    4257.17       2058
  52  15         579.14    711.00    4257.96       2066
  53  16         586.65    713.00    4258.75       2076
  54  17         589.06    713.64    4259.00       2077
  55  18         594.15    715.00    4259.53       2093
  56  19         598.58    716.18    4260.00       2094
  57  20         601.97    717.00    4260.39       2115
  58  21         610.88    719.00    4261.51       2137
  59  22         614.85    719.89    4262.00       2138
  60  23         619.79    721.00    4262.62       2159

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  61  24         628.70    723.00    4263.73       2178
  62  25         630.90    723.49    4264.00       2179
  63  26         637.61    725.00    4264.84       2198
  64  27         646.52    727.00    4265.95       2212
  65  28         646.94    727.09    4266.00       2213
  66  29         655.43    729.00    4267.06       2227
  67  30         663.44    730.82    4268.00       2228
  68  31         664.20    731.00    4268.08       2245
  69  32         672.81    733.00    4269.02       2255
  70  33         681.42    735.00    4269.95        706

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  71  34         681.90    735.11    4270.00        707
  72  35         685.72    736.00    4270.41        746
  73  36         690.03    737.00    4270.88        781
  74  37         691.13    737.36    4271.00        782
  75  38         694.33    738.00    4271.35        818
  76  39         698.64    739.00    4271.81        850
  77  40         700.36    739.40    4272.00        851
  78  41         702.94    740.00    4272.28        889
  79  42         707.25    741.00    4272.75        924
  80  43         709.59    741.54    4273.00        925
```

Figure 4-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  81  44          711.55    742.00    4273.21          955
  82  45          715.85    743.00    4273.68          998
  83  46          723.60    744.92    4275.00         1030
  84  47          723.89    745.00    4275.07         1066
  85  48          727.64    746.00    4275.98         1095
  86  49          727.72    746.02    4276.00         1096
  87  50          731.38    747.00    4276.89         1119
  88  51          731.85    747.12    4277.00         1120
  89  52          735.13    748.00    4277.80         1152
  90  53          735.97    748.22    4278.00         1153

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  91  54          738.88    749.00    4278.71         1182
  92  55          740.10    749.32    4279.00         1183
  93  56          742.63    750.00    4279.61         1219
  94  57          744.22    750.42    4280.00         1220
  95  58          746.38    751.00    4280.52         1254
  96  59          748.35    751.53    4281.00         1255
  97  60          750.13    752.00    4281.43         1283
  98  61          752.47    752.63    4282.00         1222
  99  62          753.87    753.00    4282.34         1317
 100  63          756.60    753.73    4283.00         1318

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 101  64          757.62    754.00    4283.25         1350
 102  65          760.72    754.83    4284.00         1351
 103  66          761.37    755.00    4284.16         2326
 104  67          764.85    755.93    4285.00         1391
 105  68          780.14    760.00    4288.70         1499
 106  69          786.01    761.22    4290.00         1500
 107  70          804.21    765.00    4294.01         1592
 108  71          808.68    765.93    4295.00         2374
 109  72          809.03    766.00    4295.08         2392
 110  73          817.74    767.81    4297.00         2393

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 111  74          818.65    768.00    4297.20         2404
 112  75          826.81    769.69    4299.00         2405
 113  76          828.28    770.00    4299.32         1626
 114  77          831.34    770.64    4300.00         1627
 115  78          849.08    775.00    4303.42         1642
 116  79          858.58    777.69    4305.00         1643
 117  80          866.76    780.00    4306.36            0
 118  LAST       -10.00
 119  UTM COORDINATES OF GRID ORIGIN                680.00     4230.00
 120  STARTING LOCATION UTM X & Y                   680.44     4224.05
```

Figure 4-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 121  TRAJECTORY DESCRIPTIONS                  360.0     11.09     66.27
 122  B  TIME(MIN), WIND V & D (KM/HR & DEG)   420.00     8.421    62.38
 123  C                                        480.00    16.93     45.98
 124  D                                        540.00    17.18     21.53
 125  E                                        600.00    15.40     29.04
 126  F                                        660.00    15.38     25.00
 127  G                                        720.00    21.63     42.26
 128  H                                        780.00    18.18     46.72
 129  I                                        840.00    19.68     30.47
 130  J                                        900.00    21.53     28.28

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 131  K                                        960.00    24.82     15.54
 132  L                                        1020.0    24.82     15.54
 133  M                                        1080.0    24.82     15.54
 134  LAST TRAJECTORY NODE (DUMMY NEGATIVE)    -100.0
 135  NUMBER OF KEMOD VERTICAL MESH POINTS     05
 136  MESH POINT ELEVATIONS (ABOVE SURFACE)     0.0          (METERS)
 137  2                                         120.0
 138  3                                         300.0
 139  4                                         600.0
 140  5                                         1200.0

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 141  SURFACE TEMPERATURES                       360.      21.50
 142                                             420.      22.41
 143                                             480.      24.39
 144                                             540.      26.46
 145                                             600.      27.67
 146                                             660.      26.46
 147                                             720.      28.51
 148                                             780.      29.61
 149                                             840.      29.43
 150                                             900.      29.27

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 151                                             960.      29.66
 152                                            1020.      29.64
 153  LAST DUMMY SURFACE TEMP (NEGATIVE)        -10.
 154  STABILITY CLASSES                         360.                2
 155    USE 1-6 FOR A-F                         660.                5
 156                                            720.                2
 157                                            900.                3
 158                                            960.                2
 159  LAST STABILITY (DUMMY NEGATIVE)           -10.
 160  MIXING LAYER DEPTHS (METERS)              360.       200.
```

Figure 4-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 161                                          420.      270.
 162                                          480.      340.
 163                                          540.      410.
 164                                          600.      490.
 165                                          660.      600.
 166                                          720.      710.
 167                                          780.      820.
 168                                          840.      930.
 169                                          900.     1040.
 170                                          960.     1080.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 171  LAST MIXING LAYER DEPTH (NEGATIVE)       -10.
 172  END
```

Figure 4-1 (Continued)

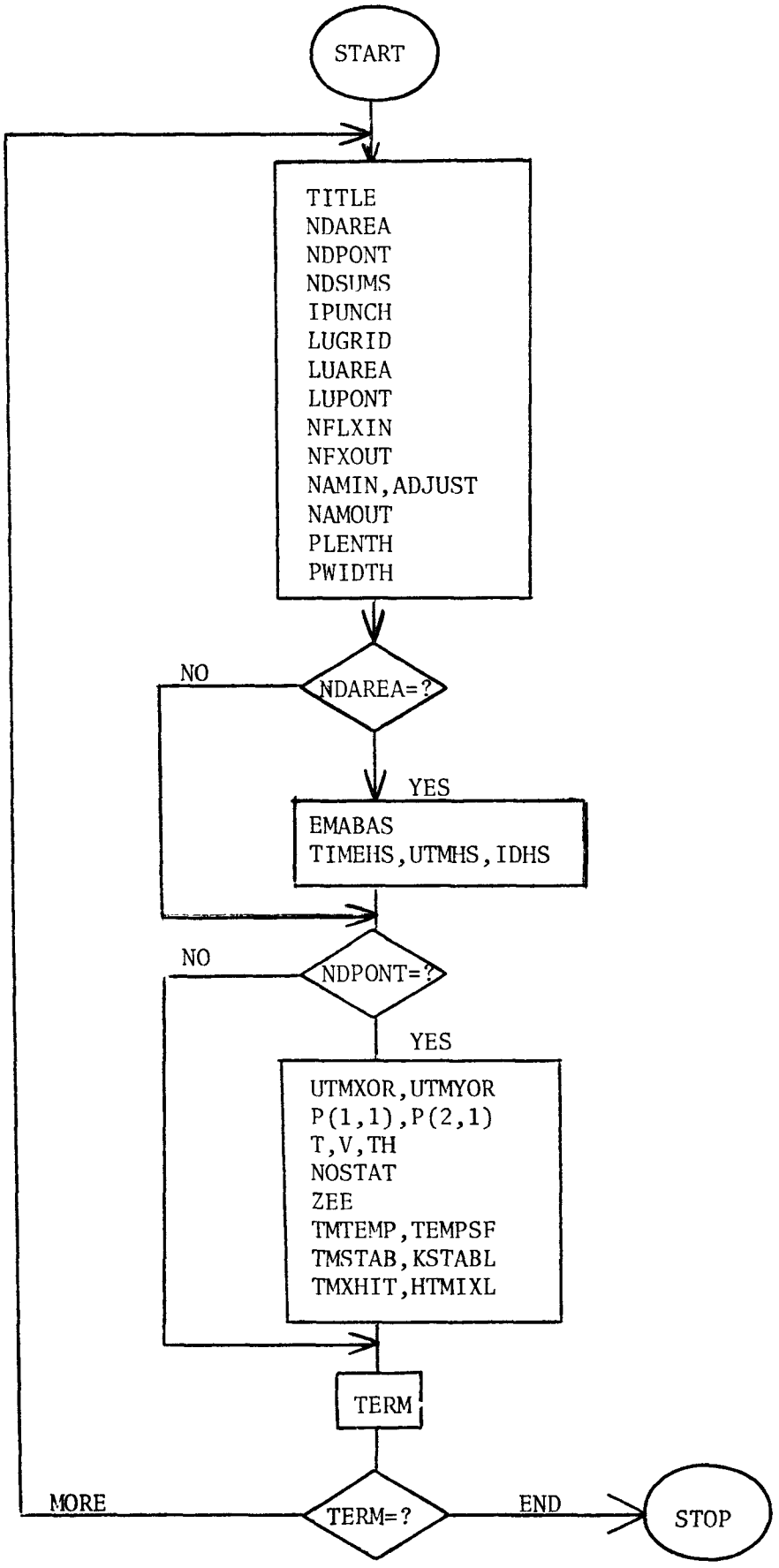


Figure 4-2 EMMOD Input Variable Sequence and Options

4.1.1 General Inputs

The input stream begins with the user-selected page label (TITLE), similar to the first card in the METMOD input deck. This card is followed by four option flags, which are specified by YES or NO inputs. The user first specifies whether (YES) or not (NO) the area source emissions schedule is to be generated in the run. The next option flag similarly indicates whether the point source schedule is desired. The third flag indicates whether or not the user wants the hourly sums of the regional inventories generated and printed. The fourth flag instructs the program to punch the area source and point source emission schedules in a format suitable for input to the chemical-diffusion module. Since the punched output of the program is typically 100 to 500 cards, the use of this option is strongly recommended for error-free data transfer.

Next, the user supplies three cards with the logical unit numbers of the peripheral devices assigned to the three input files (tapes). These integer constants refer to the emissions grid file, the area source file, and point source file, respectively.

The user next specifies the number of species with emissions data in the input files (NFLXIN) and in the output schedules (NFXOUT). The RAPS area source input file has data for eleven species and the point source file has data for ten species. The user should specify NFLXIN equal to eleven in the input deck since the program internally accounts for this discrepancy between the files. The program utilizes data for seven species, thus, the user should set NFXOUT equal to seven in the deck.

Next, the program reads NFLXIN cards, which contain 4-character names for the species in the input files. Adjacent to each species name is a conversion factor to adjust the units of the input data to the units required by the code. The non-unity conversion factors shown in Figure 4-1 are those used to convert the nitrogen oxides (NOX), sulfur oxides (SOX), total hydrocarbons (THC), and carbon monoxide (CO) emissions data from kilograms per hour to moles per hour. A value of unity is specified for the individual hydrocarbon class data since this data is provided in moles per hour. The program does not use the particulate (PART) and dust (DUST) emissions data, so their units are left in kilograms per hour by inputting a conversion factor of one.

The program proceeds to read NFXOUT cards, which contain the 4-character names of the species in the output schedules. These species name cards, as well as those described above, must appear in the order shown in Figure 4-1 for proper labeling of the output data.

The last two general inputs designate the air parcel length and width in kilometers. These are specified to correspond to the minimum grid square size to which the area source data has been allocated. For the St. Louis application, the air parcel is considered to be on kilometer square.

4.1.2 Area Source Emission Control Inputs

The user first specifies a set of NFXOUT cards which indicate the default area source emission rates in moles per hour for the region outside of the emissions grid. The order in which this data is input corresponds to the order of the output species names, as listed in Figure 4-1. For the St. Louis application, the region outside the emissions grid is rural, thus, the default emission rates have been set to zero in the sample problem input stream. There are, of course, cases where non-zero default emission rates would be appropriate. For example, if a user generates a trajectory which parallels a major vehicular artery prior to entering the emissions grid, the user should supply reasonable estimates of the emission rates based on the vehicle-miles-traveled per one-by-one kilometer grid square.

Next, the user supplies a set of cards which describe the trajectory grid square crossing schedule. Each card contains a time in minutes (0-1440 clock), global x and y coordinates of the crossing point, and the corresponding grid square identifier. These cards are punched by the meteorological module to facilitate data transfer. The last card in the set is identified by a negative entry in columns 11-20.

4.1.3 Point Source Emission Control Inputs

The portion of the input deck which applies to the generation of point source emission schedules begins with the specification of global coordinates for the origin of the local coordinate system. This card is followed by data pertaining to the trajectory. The card specifying the trajectory start point in x and y global coordinates is followed by a

set of cards with the trajectory node times (minutes), wind velocities (km/hr), and wind directions (polar angles). These cards are output by the meteorological module, and the final entry in the schedule is a negative number in columns 41-50 of the last card.

The user next inputs the number of vertical mesh points and the vertical mesh point elevations in the same format used for the meteorological module input deck. These cards are followed by schedules of surface temperatures, atmospheric stability categories, and mixing height elevations along the trajectory. The times in the schedules are specified on the 0-1440 minute clock, and each of the schedules is terminated by a negative entry in columns 41-50 of the last card. The temperature and stability class data cards are output by the meteorological module. The mixing height elevations are specified externally at hourly intervals in meters above the surface from available meteorological data.

4.2 Description of the RAPS Input Files

The format and contents of the RAPS/St. Louis input files are described in this section to help the user understand the program's present raw data requirements and retrieval software. A common feature of the files is that the data are blocked in groups of 510 binary words per physical record on the tapes. Each file, of course, has a different number of physical records, as described below. The data are read with BUFFER IN statements in the CDC version of the code, but the equivalent binary READ statements for other computers are included as comment cards in the source listings contained in Appendix C.

4.2.1 Emissions Grid File

The emissions grid file contains 32 physical records of data. The first word in the file is the total number of grid squares which is less than or equal to 2000 for the RAPS grid. This word is followed by the specification of eight parameters for each grid square as defined below:

1) Grid Square Identifier

2) Grid Square Area (km^2)

3) UTM Horizontal Coordinate of Southwest Corner (km)

4) UTM Vertical Coordinate of Southwest Corner (m)

5) State Code

6) County Code

7) Average Source Height

8) Surface Roughness Length (cm)

The EMMOD program reads all the data in the file, but stores only the total number of grid squares and two arrays with the grid square identifiers and corresponding areas. The importance of this data stems from the fact that the ordering of the data in the area source emissions file corresponds to that of the grid square identifiers in this file.

4.2.2 Area Source Emissions File

The area source emissions file contains 1056 physical records of emissions data for the single day. The file contains subsets of 44 records for each hour beginning at midnight. Each hourly subset lists the Julian data, number of grid squares, hour (0-23), and emission rates by grid squares for the eleven species listed below:

1) Particulates (kg/hr)

2) Sulfur Oxides (kg/hr weighed as SO_2)

3) Nitrogen Oxides (kg/hr weighed as NO_2)

4) Total Hydrocarbons (kg/hr)

5) Carbon Monoxide (kg/hr)

6) Dust (kg/hr)

7) Nonreactive Hydrocarbons (moles/hr)

8) Alkanes (moles/hr)

9) Alkenes (moles/hr)

10) Aromatics (moles/hr)

11) Aldehydes (moles/hr)

The program is designed to read as much of each hour's data as necessary to satisfy the list of grid squares traversed by the trajectory during a a particular hour. See Section 4.4 for a description of these procedures.

4.2.3 Point Source Emissions File

The point source emissions file may contain up to 408 physical records for one day. The file is structured with hourly subsets of data, and the number of point sources considered may vary from hour to hour. The file contains data for up to 505 point sources, which translates to a maximum of 17 records per hour. Each hourly subset contains the date and hour stored as a 7-character word (YRDAYHR = year, Julian day, hour) and the number of sources for that hour. These two words are followed by the seventeen parameters described below for each point source.

1) UTM Horizontal Coordinate (km)
2) UTM Vertical Coordinate (km)
3) Stack Height (m)
4) Stack Diameter (m)
5) Stack Effluent Temperature (^{o}C)
6) Stack Effluent Volumetric Flow Rate (M^3/min)
7) Particulates (kg/hr)
8) Sulfur Oxides (kg/hr weighed as SO_2)
9) Nitrogen Oxides (kg/hr weighed as NO_2)
10) Total Hydrocarbons (kr/hr)
11) Carbon Monoxide (kg/hr)
12) Nonreactive Hydrocarbons (moles/hr)
13) Alkanes (moles/hr)
14) Alkenes (moles/hr)
15) Aromatics (moles/hr)
16) Aldehydes (moles/hr)
17) Stack Identification Code

The program reads all the data for each hour of the trajectory and stores all the data except the stack diameters, particulates, total hydrocarbons, nonreactive hydrocarbons, and stack identification codes.

4.3 Description of the Outputs

This section describes the printed and punched output obtained from the EMMOD program. The output includes a listing of the input deck produced by PREDAT, verification of portions of the input data, and listings of the area and point source emissions data along the trajectory. The program also prints the hourly sums of the regional emission inventory when the user selects this option.

4.3.1 Area Source Emission Outputs

Following the PREDAT input deck listing shown in Figure 4-1, the program lists verification of the trajectory grid square crossing schedule. Figure 4-3 shows the first page of output for the sample problem schedule, containing the crossing times, global x and y coordinates of the crossing points, and the grid square identifiers. The reader may observe that additional entries have been placed in the schedule for the even hours.

Next, the program lists the hourly regional area source emission sums when this option is selected. Figure 4-4 shows the emission rate sums for the eleven species with input data. These regional emission rates are tabulated in moles per hour with the exception of particulates and dust, which are listed in kilogram per hour. The reader may note the message from subroutine AREAEM shown in the figure, which informs the reader that due to parity errors in the file, the regional sums could not be accurately tabulated for a particular hour. The program may use some data from previous hours when it encounters this problem, as discussed in Section 4.4.

The program then lists several pages of output with area source emission rates corresponding to the grid square crossing schedule. Figure 4-5 shows a sample of the schedule, listing the crossing time, grid square identifier, and emission rates in moles per hour. At the end of this listing, the program provides the total moles of each species entrained into the air parcel during the trajectory (not shown). This

ERT EMISSIONS MODULE -- ST. LOUIS -- 6/29/76 (181)

TRAJECTORY DATA BY GRID SQUARE

TIME	UTM-X	UTM-Y	ID
360.00	680.44	4224.05	0
394.40	683.00	4229.87	64
420.00	685.19	4234.15	64
421.46	685.32	4234.39	72
426.39	685.32	4235.00	65
480.00	689.52	4242.10	65
486.07	690.00	4242.90	81
496.41	692.03	4245.00	82
511.58	695.00	4248.08	99
521.06	696.86	4250.00	2028
537.08	700.00	4253.25	129
540.00	700.75	4253.62	129
551.01	703.50	4255.00	130
556.62	705.00	4255.59	2039
564.13	707.00	4256.38	2048
570.04	708.57	4257.00	2049
571.65	709.00	4257.17	2058
579.14	711.00	4257.96	2066
586.65	713.00	4258.75	2076
589.06	713.64	4259.00	2077
594.15	715.00	4259.53	2093
598.58	716.18	4260.00	2094
600.00	716.52	4260.16	2094
601.97	717.00	4260.39	2115

Figure 4-3

ERT EMISSIONS MODULE -- ST. LOUIS -- 6/29/76 (181)

REGIONAL AREA SOURCE EMISSION SUMS (MOLES/HOUR)

TIME	PART	SOX	NOX	THC	CO	DUST	NRHC	PARF	OLEF	AROM	ALDE
360.0	1.408E+05	1.734E+04	4.260E+05	6.660E+05	7.681E+06	1.130E+04	7.687E+04	1.536E+05	1.032E+05	4.128E+04	2.835E+04
420.0	1.304E+05	1.930E+04	4.473E+05	6.589E+05	7.570E+06	1.020E+04	7.665E+04	1.552E+05	1.006E+05	3.941E+04	2.665E+04
480.0	1.465E+05	1.867E+04	4.073E+05	6.982E+05	7.100E+06	8.481E+03	8.665E+04	1.600E+05	1.037E+05	4.473E+04	3.040E+04
540.0	1.485E+05	1.995E+04	4.183E+05	7.070E+05	7.378E+06	8.915E+03	8.762E+04	1.596E+05	1.055E+05	4.610E+04	3.136E+04
600.0	1.362E+05	1.956E+04	4.410E+05	7.472E+05	7.966E+06	9.831E+03	9.217E+04	1.679E+05	1.124E+05	4.895E+04	3.337E+04
660.0	1.287E+05	2.016E+04	4.642E+05	7.911E+05	8.573E+06	1.078E+04	9.717E+04	1.777E+05	1.199E+05	5.189E+04	3.545E+04
720.0	1.477E+05	1.981E+04	4.502E+05	7.678E+05	8.212E+06	1.021E+04	9.452E+04	1.727E+05	1.158E+05	5.022E+04	3.421E+04

SUBROUTINE AREAEM ENCOUNTERED PARITY ERRORS -- SOME DATA FROM PREVIOUS HOUR USED

SUBROUTINE AREAEM ENCOUNTERED PARITY ERRORS -- SOME DATA FROM PREVIOUS HOUR USED

Figure 4-4

ERT EMISSIONS MODULE -- ST. LOUIS -- 6/29/76 (181)

AREA SOURCE EMISSION RATES -- MOLES/HR

TIME	ID	NOX	PARF	OLEF	AROM	ALDE	CO	SOX
360.00	0	0.	0.	0.	0.	0.	0.	0.
394.40	64	1.5358E+00	1.1579E-01	7.5503E-02	5.2732E-02	3.7162E-02	9.7460E+00	8.3542E-02
420.00	64	1.5384E+00	1.1581E-01	7.5518E-02	5.2737E-02	3.7177E-02	9.7471E+00	8.3542E-02
421.46	72	2.6094E+00	7.7094E-01	5.1429E-01	2.2089E-01	1.4090E-01	3.9558E+01	1.1636E-01
426.39	65	3.4124E+00	9.3613E-01	6.9291E-01	3.3134E-01	2.0666E-01	5.7848E+01	1.4653E-01
480.00	65	3.1214E+00	8.2712E-01	6.0673E-01	2.9523E-01	1.8510E-01	5.0896E+01	1.3716E-01
486.07	81	2.8224E+00	1.1907E+00	6.6651E-01	2.5071E-01	1.5186E-01	4.1140E+01	1.2327E-01
496.41	82	1.7249E+00	5.4565E-01	2.3331E-01	8.3888E-02	4.8603E-02	1.1679E+01	8.8268E-02
511.58	99	1.2834E+00	1.5005E-01	8.3601E-02	6.9617E-02	3.9611E-02	1.0059E+01	6.8186E-02
521.06	2028	1.3106E+00	3.1026E-01	1.4598E-01	6.5374E-02	3.9885E-02	1.0229E+01	6.9496E-02
537.08	129	9.3589E-01	3.0628E-01	1.2538E-01	6.3952E-02	3.0194E-02	7.2528E+00	4.8335E-02
540.00	129	9.3973E-01	2.8878E-01	1.1898E-01	6.3958E-02	3.0214E-02	7.2544E+00	4.8335E-02
551.01	130	1.3400E+00	2.4266E+00	1.1248E+00	3.6323E-01	2.3463E-01	2.4126E+01	7.7250E-02
556.62	2039	2.7680E+00	1.6504E-01	8.8494E-02	7.3531E-02	5.0729E-02	1.0378E+01	1.5134E-01
564.13	2048	6.0738E-01	1.1611E-01	4.4769E-02	5.4346E-02	2.0984E-02	4.9002E+00	3.0428E-02
570.04	2049	6.0738E-01	1.1611E-01	4.4769E-02	5.4346E-02	2.0984E-02	4.9002E+00	3.0428E-02
571.63	2058	1.1478E+00	5.6061E-01	1.4833E-01	2.6501E-01	6.7663E-02	1.1965E+01	3.3576E-02
579.14	2066	2.4138E+00	2.8468E+00	8.6321E-01	7.0197E-01	1.6614E-01	2.6755E+01	4.6356E-02
586.65	2076	1.6766E+01	1.4414E+01	8.0023E+00	2.5849E+00	1.5873E+00	4.6777E+02	4.5901E-01
589.06	2077	3.6274E+00	8.2383E+00	3.1679E+00	6.0526E-01	2.3172E-01	5.3355E+01	9.3886E-02
594.15	2093	8.0745E+00	8.7550E+00	6.5108E+00	3.8904E+00	2.5968E+00	3.3657E+02	3.1749E-01
598.58	2094	1.5456E+01	3.0484E+01	1.7883E+01	7.4558E+00	5.1314E+00	6.6376E+02	6.3736E-01
600.00	2094	1.6795E+01	3.1038E+01	1.8348E+01	7.6546E+00	5.2679E+00	7.0543E+02	6.7479E-01
601.97	2115	1.6911E+01	1.0559E+01	7.3760E+00	3.4778E+00	2.1121E+00	5.9894E+02	5.4346E-01

Figure 4-5

portion of the output is followed by a listing of the same area source emission rate schedule in the code's units of mole fraction-meters per minute. This schedule is illustrated in Figure 4-6, which shows the global coordinates of the grid square crossing points, in addition to the crossing times and emission rates. This schedule concludes the printed output from the area source portion of the program.

4.3.2 Point Source Emission Outputs

The output from the point source portion of the program begins with a verification of the trajectory description (see Figure 4-7). This page of output is followed by verification of the temporal schedules for atmospheric stability class, the surface temperature, and mixing height. Examples of these verification outputs are shown in Figure 4-8.

The program prints the regional sums of the point source emissions for each hour of the trajectory when this option is designated in the input deck. Figure 4-9 is an example of this printout, with the regional emissions rates of all the input species listed in units of moles per hour, except particulates, which remain in kilograms per hour.

The program lists data concerning the point sources with emissions entrained into the air parcel. As shown in Figure 4-10, the program prints the effective time at which a source is passed, the fraction of its emissions which will be entrained into the air parcel moving along the trajectory, the perpendicular distance from the trajectory centerline to the source, and the emission rates for the output species in moles per hour. The treatment of point source emissions in the Lagrangian reference frame is quite complex; therefore, the reader is referred to Section 4.4 for an explanation of the methods used to determine the effective passing time and the fraction of emissions entrained into the air parcel.

Next, the program prints the actual number of moles entrained into the air parcel from each point source and its vertical distribution. Figure 4-11 is an example of this listing, which first shows the output species names and the ranges of elevations of each vertical cell to which the emissions are allocated. These ranges of elevations are determined from the vertical mesh geometry input to the program. For each point source in the schedule, the listing shows the effective

ERT EMISSIONS MODULE -- ST. LOUIS -- 6/29/76 (181)

AREA SOURCE EMISSION FLUXES MOLE FRACTION - METERS/MINUTE

TIME	UTM-X	UTM-Y	NOX	PARF	OLEF	AROM	ALDE	CO	SOX
360.0	680.4	4224.0	0.	0.	0.	0.	0.	0.	0.
394.4	683.0	4229.9	6.2948E-10	4.7461E-11	3.0947E-11	2.1613E-11	1.5232E-11	3.9946E-09	3.4242E-11
420.0	685.2	4234.1	6.3056E-10	4.7468E-11	3.0953E-11	2.1616E-11	1.5238E-11	3.9951E-09	3.4242E-11
421.5	685.3	4234.4	1.0695E-09	3.1599E-10	2.1080E-10	9.0539E-11	5.7753E-11	1.6214E-08	4.7693E-11
426.4	685.3	4235.0	1.3987E-09	3.8370E-10	2.8401E-10	1.3581E-10	8.4703E-11	2.3710E-08	6.0059E-11
480.0	689.5	4242.1	1.2794E-09	3.3902E-10	2.4868E-10	1.2101E-10	7.5869E-11	2.0861E-08	5.6217E-11
486.1	690.0	4242.9	1.1568E-09	4.8803E-10	2.7318E-10	1.0276E-10	6.2245E-11	1.6862E-08	5.0526E-11
496.4	692.0	4245.0	7.0699E-10	2.2365E-10	9.5628E-11	3.4384E-11	1.9921E-11	4.7868E-09	3.6179E-11
511.6	695.0	4248.1	5.2604E-10	6.1504E-11	3.4266E-11	2.8534E-11	1.6236E-11	4.1231E-09	2.7948E-11
521.1	696.9	4250.0	5.3719E-10	1.2717E-10	5.9833E-11	2.6795E-11	1.6348E-11	4.1926E-09	2.8485E-11
537.1	700.0	4253.2	3.8360E-10	1.2554E-10	5.1388E-11	2.6212E-11	1.2376E-11	2.9728E-09	1.9811E-11
540.0	700.7	4253.6	3.8517E-10	1.1836E-10	4.8767E-11	2.6215E-11	1.2384E-11	2.9734E-09	1.9811E-11
551.0	703.5	4255.0	5.4923E-10	9.9462E-10	4.6101E-10	1.4888E-10	9.6169E-11	9.8887E-09	3.1663E-11
556.6	705.0	4255.6	1.1345E-09	6.7646E-11	3.6272E-11	3.0139E-11	2.0793E-11	4.2536E-09	6.2029E-11
564.1	707.0	4256.4	2.4895E-10	4.7590E-11	1.8350E-11	2.2275E-11	8.6010E-12	2.0085E-09	1.2472E-11
570.0	708.6	4257.0	2.4895E-10	4.7590E-11	1.8350E-11	2.2275E-11	8.6010E-12	2.0085E-09	1.2472E-11
571.6	709.0	4257.2	4.7047E-10	2.2978E-10	6.0799E-11	1.0862E-10	2.7733E-11	4.9042E-09	1.3762E-11
579.1	711.0	4258.0	9.8934E-10	1.1668E-09	3.5381E-10	2.8772E-10	6.8099E-11	1.0966E-08	1.9000E-11
586.7	713.0	4258.7	6.8719E-09	5.9080E-09	3.2800E-09	1.0595E-09	6.5059E-10	1.9173E-07	1.8814E-10
589.1	713.6	4259.0	1.4868E-09	3.3767E-09	1.2984E-09	2.4808E-10	9.4978E-11	2.1869E-08	3.8482E-11
594.2	715.0	4259.5	3.3096E-09	3.5885E-09	2.6686E-09	1.5946E-09	1.0644E-09	1.3795E-07	1.3013E-10
598.6	716.2	4260.0	6.3348E-09	1.2495E-08	7.3298E-09	3.0559E-09	2.1032E-09	2.7206E-07	2.6124E-10
600.0	716.5	4260.2	6.8837E-09	1.2722E-08	7.5204E-09	3.1374E-09	2.1592E-09	2.8914E-07	2.7658E-10
602.0	717.0	4260.4	6.9314E-09	4.3280E-09	3.0233E-09	1.4255E-09	8.6568E-10	2.4549E-07	2.2275E-10

Figure 4-6

ERT EMISSIONS MODULE -- ST. LOUIS -- 6/29/76 (181)

TRAJECTORY DATA

LEG NO.	TIME (MIN)	X (KM)	Y (KM)	VELOCITY (KM/HR)	THETA (DEGREES)
1	360.0	680.44	4224.05	11.090	66.270
2	420.0	684.90	4234.20	8.421	62.380
3	480.0	688.81	4241.66	16.930	45.980
4	540.0	700.57	4253.84	17.180	21.530
5	600.0	716.55	4260.14	15.400	29.040
6	660.0	730.02	4267.62	15.380	25.000
7	720.0	743.96	4274.12	21.630	42.260
8	780.0	759.96	4288.66	18.180	46.720
9	840.0	772.43	4301.90	19.680	30.470
10	900.0	789.39	4311.88	21.530	28.280
11	960.0	808.35	4322.06	24.820	15.540
12	1020.0	832.26	4328.73	24.820	15.540
13	1080.0	856.18	4335.38	24.820	15.540

Figure 4-7

STABILITY CLASS DATA

TIME	CLASS
360	2
660	5
720	2
900	3
960	2

SURFACE TEMPERATURE DATA

TIME	TEMPERATURE
360	21.5
420	22.4
480	24.4
540	26.5
600	27.7
660	26.5
720	28.5
780	29.6
840	29.4
900	29.3
960	29.7
1020	29.6

MIXING LAYER DEPTH DATA

TIME	DEPTH
360	200.0
420	270.0
480	340.0
540	410.0
600	490.0
660	600.0
720	710.0
780	820.0
840	930.0
900	1040.0
960	1080.0

Figure 4-8

ERT EMISSIONS MODULE -- ST. LOUIS -- 6/29/76 (181)

REGIONAL POINT SOURCE EMISSIONS (MOLES/HOUR)

TIME	PART	SOX	NOX	THC	CO	NRHC	PARF	OLEF	AROM	ALDE
360.0	4.97ME+03	1.882E+06	7.693E+05	1.148E+05	6.114E+05	2.509E+04	2.313E+04	4.987E+03	1.420E+04	4.113E+02
420.0	7.614E+03	2.083E+06	8.422E+05	1.780E+05	6.205E+05	3.078E+04	3.763E+04	5.319E+03	2.321E+04	4.528E+02
480.0	7.674E+03	2.121E+06	8.566E+05	1.781E+05	6.310E+05	3.102E+04	3.763E+04	5.327E+03	2.321E+04	4.609E+02
540.0	7.776E+03	2.201E+06	9.079E+05	1.785E+05	6.278E+05	3.158E+04	3.765E+04	5.346E+03	2.322E+04	4.798E+02
600.0	7.863E+03	2.214E+06	9.193E+05	1.787E+05	6.285E+05	3.184E+04	3.766E+04	5.354E+03	2.322E+04	4.883E+02
660.0	7.857E+03	2.138E+06	8.712E+05	1.785E+05	6.174E+05	3.147E+04	3.765E+04	5.342E+03	2.322E+04	4.760E+02
720.0	7.864E+03	2.142E+06	8.811E+05	1.786E+05	6.179E+05	3.175E+04	3.766E+04	5.351E+03	2.322E+04	4.851E+02
780.0	7.858E+03	2.131E+06	8.663E+05	1.783E+05	6.172E+05	3.132E+04	3.764E+04	5.337E+03	2.322E+04	4.709E+02
840.0	7.763E+03	2.128E+06	8.641E+05	1.785E+05	6.170E+05	3.125E+04	3.764E+04	5.335E+03	2.322E+04	4.687E+02
900.0	4.315E+03	2.129E+06	8.635E+05	1.526E+05	6.169E+05	2.925E+04	3.161E+04	5.213E+03	1.937E+04	4.673E+02
960.0	4.268E+03	2.124E+06	8.611E+05	1.519E+05	6.167E+05	2.915E+04	3.141E+04	5.210E+03	1.933E+04	4.646E+02
1020.0	4.185E+03	2.120E+06	8.578E+05	1.518E+05	6.164E+05	2.905E+04	3.140E+04	5.207E+03	1.933E+04	4.616E+02

Figure 4-9

ERT EMISSIONS MODULE -- ST. LOUIS -- 6/29/76 (181)

POINT SOURCE EMISSION RATES -- MOLES/HOUR

TIME	FRACTION	DISTANCE	NOX	PARF	OLEF	AROM	ALDE	CO	SOX
709.80	.1953	.206	0.	0.	0.	0.	0.	0.	0.
711.40	.1929	1.202	0.	7.4593E+02	4.3073E+03	0.	1.5240E+00	0.	0.
711.48	.1896	.347	0.	0.	0.	0.	0.	7.4596E+02	4.3073E+03
711.50	.1898	.334	0.	1.5240E+00	0.	0.	0.	0.	8.1746E+00
719.64	.1725	.617	8.1746E+00	0.	0.	0.	0.	0.	0.
721.87	.1662	.961	2.0038E-01	6.6793E-02	2.0038E-01	0.	7.4610E+02	4.3073E+03	0.
721.87	.1662	.961	1.5240E+00	0.	0.	8.5893E-03	3.2580E-02	8.1746E+00	0.
722.70	.1677	.655	0.	5.4431E+00	0.	0.	0.	0.	0.
722.70	.1677	.655	0.	0.	0.	0.	7.5530E+02	4.3026E+03	0.
725.51	.1541	1.203	0.	0.	7.4598E+02	4.3073E+03	0.	1.5240E+00	0.
728.18	.1757	.313	0.	2.9484E+01	0.	0.	0.	0.	0.
728.42	.1668	.764	0.	0.	0.	7.5473E+02	4.3023E+03	0.	1.9202E+00
728.42	.1668	.764	0.	0.	0.	3.3566E+01	0.	0.	0.
728.42	.1668	.764	0.	0.	0.	0.	0.	0.	7.5469E+02
728.69	.1402	1.590	0.	0.	0.	8.1746E+00	0.	0.	0.
728.95	.1760	.307	0.	0.	0.	0.	7.5444E+02	4.3027E+03	0.
728.97	.1761	.300	0.	3.7467E+02	0.	0.	0.	0.	0.
729.04	.1776	.093	0.	0.	0.	0.	0.	0.	0.
729.29	.1758	.337	0.	7.5485E+02	4.3027E+03	0.	2.8956E+00	0.	0.
729.30	.1691	.688	0.	0.	0.	1.7690E+01	0.	0.	0.
729.30	.1691	.688	0.	0.	0.	0.	0.	0.	7.5499E+02
729.30	.1691	.688	0.	0.	0.	0.	0.	7.5499E+02	4.3023E+03
729.30	.1691	.688	0.	2.3774E+00	0.	0.	0.	5.5792E+01	0.
729.30	.1691	.688	0.	0.	0.	0.	0.	0.	0.

Figure 4-10

ERT EMISSIONS MODULE -- ST. LOUIS -- 6/29/76 (181)

POINT SOURCE EMISSIONS -- MOLES AND VERTICAL DISTRIBUTION

TIME	SPECIES / ELEVATIONS	NOX 0- 60	PARF 60- 210	OLEF 210- 450	AROM 450- 900	ALDE 900-1200	CO	SOX
709.80		0.	3.7508E+00	1.3748E+00	0.	0.	0.	0.
		.1283	.2913	.3656	.2149	0.0000		
711.40		3.1277E+01	0.	0.	0.	0.	0.	8.0815E+01
		.1787	.2098	.4552	.1564	0.0000		
711.48		1.7948E+00	1.3127E-01	2.8174E-01	1.8576E-02	5.6348E-02	5.1601E+01	1.6125E+00
		.1389	.2223	.4210	.2178	0.0000		
711.50		1.7963E+00	1.3138E-01	2.8197E-01	1.8592E-02	5.6395E-02	5.1643E+01	1.6139E+00
		.1389	.2223	.4210	.2178	0.0000		
719.64		0.	2.1834E+00	8.0031E-01	0.	0.	0.	0.
		.1325	.3020	.3877	.1778	0.0000		
721.87		0.	6.0416E+00	0.	8.4123E-01	0.	0.	0.
		.1416	.3200	.4085	.1299	0.0000		
721.87		0.	9.1424E-04	0.	2.1213E-04	0.	0.	0.
		.1416	.3200	.4085	.1299	0.0000		
722.70		0.	6.2444E+00	2.2888E+00	0.	0.	0.	0.
		.1562	.3083	.3409	.1946	0.0000		
722.70		0.	1.1765E+00	4.3123E-01	0.	0.	0.	0.
		.1533	.3086	.3421	.1960	0.0000		
725.51		0.	3.3133E+00	1.2144E+00	0.	0.	0.	0.
		.1329	.3013	.3873	.1785	0.0000		
728.18		0.	0.	0.	0.	0.	0.	2.6075E+01
		.1599	.2947	.3210	.2244	0.0000		
728.42		2.7704E-02	3.8232E-05	3.8232E-05	1.2744E-05	3.8232E-05	6.4478E-03	9.9561E-05
		.1615	.3133	.3445	.1808	0.0000		
728.42		2.7704E-02	3.8232E-05	3.8232E-05	1.2744E-05	3.8232E-05	6.4478E-03	9.9561E-05
		.1615	.3133	.3445	.1808	0.0000		
728.42		2.7704E-02	3.8232E-05	3.8232E-05	1.2744E-05	3.8232E-05	6.4478E-03	9.9561E-05
		.1615	.3133	.3445	.1808	0.0000		
728.69		0.	4.5731E-01	1.6762E-01	0.	0.	0.	0.
		.1338	.3034	.3876	.1752	0.0000		
728.95		0.	0.	0.	0.	0.	0.	1.4749E+00
		.1590	.3090	.3393	.1927	0.0000		
728.97		0.	0.	0.	0.	0.	0.	1.0145E+00
		.1673	.3082	.3356	.1869	0.0000		

Figure 4-11

passing time and moles of each species entrained on one line. The following line indicates the normalized vertical distribution of the emissions with respect to the vertical cells. At the conclusion of this listing (not shown), the program prints the total number of moles entrained into the air parcel during the trajectory.

Lastly, the program prints the same schedule of point source emissions in the units required by the chemical-diffusion module. Figure 4-12 shows an example of this listing where the species emissions are shown in the units of mole-fraction meters.

4.3.3 Punched Outputs

The program punches, on logical unit 1, the area source emissions schedule shown in Figure 4-6, in a format suitable for input to the chemical-diffusion module. Similarly, the program punches out the point source emissions data shown in Figure 4-12 and the vertical distributions shown in Figure 4-11 for each source in the schedule.

4.4 Computational Procedures

The computational procedures of the EMMOD program will be discussed in the order in which they occur in the program. The computational procedures of the program are quire simple for area source emissions and quite complex for point source emissions. The complexity in the treatment of point sources stems from the fact that the approximate effects of lateral (cross-trajectory) dispersion of point source emissions are accounted for in the emission schedules.

4.4.1 Area Source Emission Tabulations

Once the program has read the general inputs, the grid square crossing schedule, and default emissions from the card deck, the program reads and stores the emission grid square identifiers and areas (subroutine GRIDIT). It then proceeds to insert entries for the even hours in the grid square crossing schedule (subroutine ADHOUR). These additional entries are necessary since new emissions data is available for each hour.

Program control is then passed to subroutine AREAEM, which reads the area source emissions file and generates the schedule of emissions.

ERT EMISSIONS MODULE -- ST. LOUIS -- 6/29/76 (181)

POINT SOURCE EMISSIONS -- MOLE FRACTION-METERS

TIME	NOX	PARF	OLEF	AROM	ALDE	CO	SOX
709.80	0.	9.2240E-08	3.3810E-08	0.	0.	0.	0.
711.40	7.6917E-07	0.	0.	0.	0.	0.	1.9874E-06
711.48	4.4139E-08	3.2283E-09	6.9287E-09	4.5684E-10	1.3857E-09	1.2690E-06	3.9656E-08
711.50	4.4175E-08	3.2310E-09	6.9344E-09	4.5722E-10	1.3869E-09	1.2700E-06	3.9689E-08
719.64	0.	5.3696E-08	1.9682E-08	0.	0.	0.	0.
721.87	0.	1.4858E-07	0.	2.0688E-08	0.	0.	0.
721.87	0.	2.2484E-11	0.	5.2169E-12	0.	0.	0.
722.70	0.	1.5356E-07	5.6287E-08	0.	0.	0.	0.
722.70	0.	2.8933E-08	1.0605E-08	0.	0.	0.	0.
725.51	0.	8.1482E-08	2.9866E-08	0.	0.	0.	0.
728.18	0.	0.	0.	0.	0.	0.	6.4126E-07
728.42	6.8132E-10	9.4022E-13	9.4022E-13	3.1341E-13	9.4022E-13	1.5857E-10	2.4485E-12
728.42	6.8132E-10	9.4022E-13	9.4022E-13	3.1341E-13	9.4022E-13	1.5857E-10	2.4485E-12
728.42	6.8132E-10	9.4022E-13	9.4022E-13	3.1341E-13	9.4022E-13	1.5857E-10	2.4485E-12
728.69	0.	1.1246E-08	4.1222E-09	0.	0.	0.	0.
728.95	0.	0.	0.	0.	0.	0.	3.6272E-08
728.97	0.	0.	0.	0.	0.	0.	2.4949E-08
729.04	0.	0.	0.	0.	0.	0.	1.0422E-06
729.29	0.	0.	0.	0.	0.	0.	2.4811E-07
729.30	0.	0.	0.	3.6492E-09	0.	0.	0.
729.30	0.	0.	0.	1.8246E-08	0.	0.	0.
729.30	0.	0.	0.	9.2889E-09	0.	0.	0.
729.30	0.	0.	0.	3.9312E-08	0.	0.	0.
729.30	0.	0.	0.	3.5663E-08	0.	0.	0.

Figure 4-12

AREAEM skips data in the file pertaining to hours before the trajectory start time. It then enters an hourly time cycle to determine the largest grid square identifier in the current hour's grid square crossing list. It proceeds to read the data in the file up to the point where it passes the data for the largest identifier, and skips the remaining data for that hour. If the user has requested printing of the regional sums, of course, all of the data for that hour are read. The program then locates the species' emission rates for the grid squares being crossed, and adjusts the rates to represent moles per hour for grid squares with areas defined by the air parcel length and width inputs. The hourly time cycle is repeated until the program has located and adjusted the emission rates for all grid squares crossed by the trajectory.

If the program encounters a parity error when reading the area source file, it uses the data from the previous hour for the subset of grid squares with unreadable data.

The program prints the schedule of area source emission rates in moles per hour, and then proceeds to convert the data to the units required by the chemical-diffusion module. The required conversion of moles per hour to mole fraction-meters per minute is described below.

Let: ϕ = emission rate in moles per hour

M = molecular weight of air (28.97)

ρ = density of air (grams per cubic meters)

A = air parcel area (square kilometers)

α = 10^6 square meters per square kilometer

β = 60 minutes per hour

ϕ' = emission rate in mole fraction-meters per minute

Then: $$\phi' = \frac{M\phi}{\alpha\beta\rho A} \qquad (4\text{-}1)$$

Following this conversion, the program proceeds to print and, optionally, punch the area souce emissions schedule.

4.4.2 Point Source Emission Tabulations

The objective of the point source algorithm is to determine the pollutant emissions entrained into a Lagrangian air parcel travelling

across the modeling region. To accomplish this objective, the program examines the meteorologically-dependent characteristics of point source plumes intersecting the trajectory path. Plumes are treated on an individual basis, assuming that the lateral and vertical dispersion of emissions is Gaussian in nature. Dispersion of effluent from tall stacks and short (less than 100m) stacks are simulated by using separate sets of diffusion coefficients. The rise of an individual plume near the trajectory is calculated with consideration of inversion penetration by buoyant plumes.

To comprehend the algorithm, the reader should first understand how plumes are believed to be entrained into a fixed size Lagrangian air parcel. Figure 4-13 schematically illustrates a side view of a parcel passing a point source. Assuming for the moment that the source is within the air parcel, the parcel entrains the emissions for the time period required to pass the source. Once these emissions have been entrained, lateral dispersion occurs, which may carry a portion of the pollutants out the sides of the parcel. Similarly, the parcel may pass a source whose perpendicular distance from the trajectory's center line is greater than the parcel's half width, and its plume may disperse into the parcel at a distance downwind of the source. This lateral spreading of plumes into and out of air parcels is illustrated in Figure 4-14.

According to the K-theory of diffusion, lateral dispersion of entrained emissions continues until the lateral concentration gradients become negligible. For practical considerations, the program makes the assumption that the majority of lateral spreading occurs during a fixed time interval, currently set at one hour. Hence, the portion of stack's emissions that remains in the air parcel after one hour of dispersion forms the basis of the point source emissions schedule used by the chemical-diffusion module. The vertical distribution of the entrained emissions depends on the meteorological conditions and individual stack parameters; and, these distributions are also generated for use by the chemical-diffusion module by the program.

The point source algorithm's computations begin after the trajectory data, vertical mesh, temperatures, stability class, and mixing heights have been read by the main program, and program control is passed to subroutine PONTEM. PONTEM performs several initialization tasks, which include calling SEGMET. SEGMET determines the distances

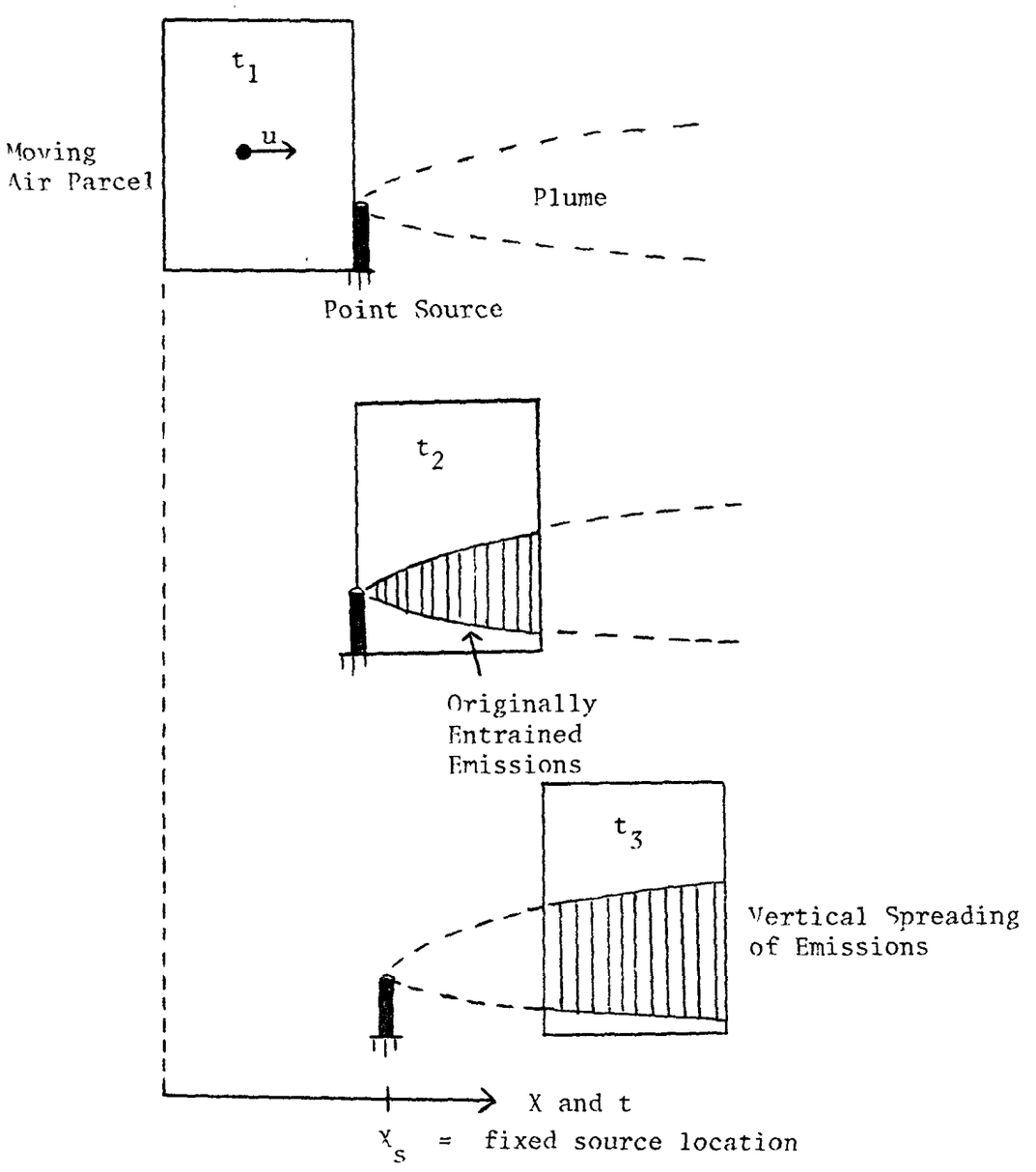


Figure 4-13 Plume Entrainment in a Lagrangian Air Parcel (Side Views at time = t_1, t_2, t_3)

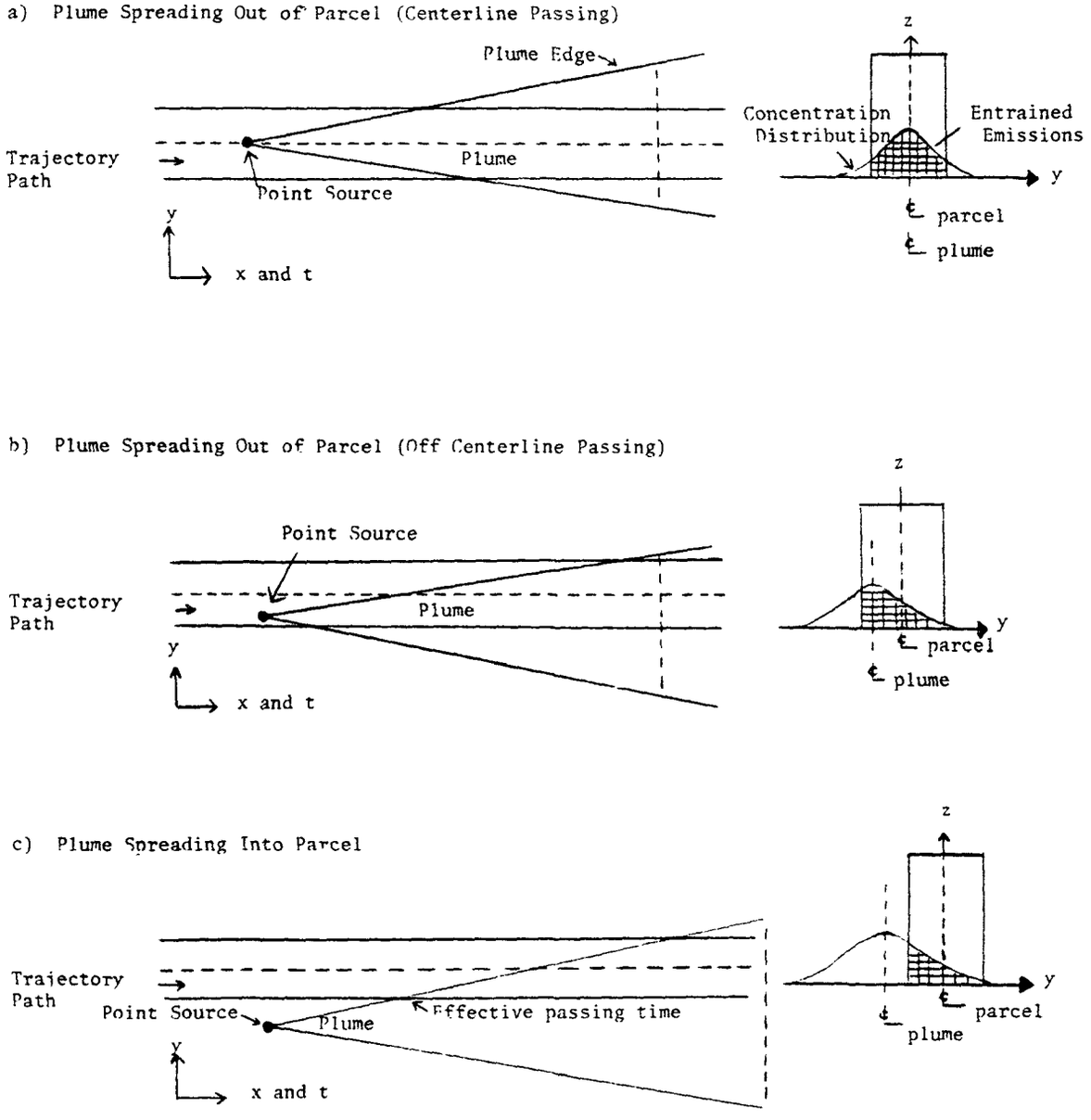


Figure 4-14 Lateral Spreading of Plumes Into and Out of Moving Air Parcels

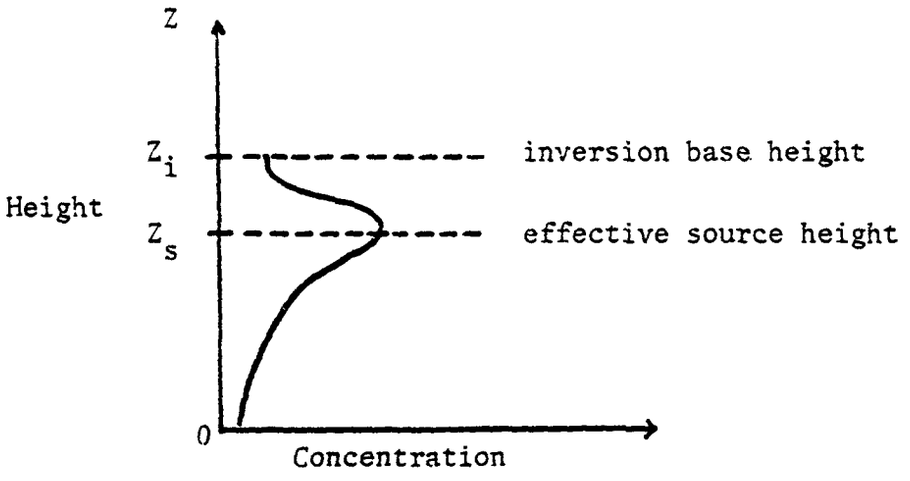


Figure 4-15 Vertical Distribution of Emission Below Inversion Base

between trajectory nodes and the bisector angles between trajectory segments. It also determines certain maximum distance criteria, which are functions of stability class and the maximum segment length. These criteria are subsequently used by subroutine LOCATE to efficiently select a subset of plumes along a segment, which may intersect the trajectory path. PONTEM then calls subroutine STACKS to read the RAPS/St. Louis point source file for the first hour of the trajectory.

Following the initializations tasks, PONTEM passes control to an outer loop, which cycles through each segment of the trajectory. Subroutine GESTAB is called to retrieve the current hour's and subsequent hour's atmospheric stability classes. Also, the time period for entrainment is calculated from the current wind speed and the parcel's length.

Control is then passed to an inner loop, which considers each point source in the current hour's emissions inventory. Subroutine LOCATE examines the location of a source relative to the trajectory segment end points and bisector angles to determine if it meets certain preliminary or relative position criteria. If a source meets the criteria, subroutine PLUMAS is called. Here, special treatment is given to sources passed with a perpendicular distance greater than the parcel half width. PLUMAS determines whether or not the edge of such a plume intersects the side of the parcel within one hour after passing the source. For practical purposes, the program considers the edge of the plume to be the 10% centerline concentration boundary (i.e., $\sigma_y = 2.14$). Plumes which fail this test are ignored. If the plume edge is found to intersect the parcel within the current hour, the program resets the passing time to the time at which the plume edge is intercepted. For sources within the air parcel path, the passing time is simply the time at which parcel center is perpendicular to the source. PLUMAS next evaluates the fraction of entrained emissions which exist in the air parcel one hour after the effective passing time. This computation involves the lateral integration of the Gaussian concentration distribution over the width of the air parcel. Changes in wind speed and/or stability class during the hour are accounted for in the assumed Gaussian distribution.

Given the fraction of emissions entrained from a source, PONTEM proceeds to calculate the vertical characteristics of the plume. Subroutine DHPLUM is called to determine the source's plume rise and inversion penetration fraction, using formulas developed by Briggs, 1969 and

1975. This information requires the following information:

- Ambient and stack effluent temperatures
- Wind speed and atmospheric stability class
- Stack height and volumetric flow rate
- Inversion height or mixing layer depth
- Potential temperature gradient in the inversion layer

The effective plume centerline height is determined from the source height and plume rise. Next, subroutine PARTIT is called to evaluate and integrate the vertical concentration distribution. This distribution is currently evaluated 20 minutes after the time of passing. For a source below the inversion base, the program assumes the distribution is Gaussian with reflections at the surface and inversion base as shown in Figure 4-15. Subroutine PARTIT integrates this profile over each vertical cell to determine the vertical distribution of emissions. The fraction of emissions which penetrate the inversion base are allocated to the cell which bounds the inversion, unless the inversion base is within ten meters of the next vertical cell's lower boundary. When this occurs, the penetration fraction is allocated to the next vertical cell.

The computations within the inner loop are complete when the program has stored the passing time, source emission rates, entrained mass, and vertical distribution for each qualifying source on a trajectory segment. Program control is then returned to the outer loop which updates the temperature and mixing height data and reads another hour's data from the RAP point source file. This cycle of computations is repeated until all the plumes intersected along the trajectory have been evaluated.

PONTEM then orders the data chronologically and converts the units of the entrained emissions to the units required by the chemical-diffusion module. Given the entrained emissions in moles, the conversion is performed as follows:

Let: ϕ = moles entrained

M = molecular weight of air (28.97)

ρ = density of air (grams per cubic meter)

A = air parcel area (square kilometers)

α = 10^6 square meters per square kilometers

ϕ' = mole fraction-meters entrained

Then: $\phi' = \frac{M\phi}{\alpha\rho A}$

Following the units conversion, PONTEM proceeds to print, and optionally punch, the point source emissions schedule.

4.5 Internally Specified Data

The emissions module has numerous internally specified parameters of which the user should be aware. Table 4-2 lists the parameter's code variable names, functions, and location in the code. Only one of the parameters, the density of air, is specific to a particular modeling region. The user should change the surface air density value when applying the model to a region with surface elevations significantly above sea level.

Two additional tables are provided which contain information on plume dispersion coefficients used in the model. Table 4-3 list the plume dispersion coefficients, σ_y and σ_z, for non-tall stacks. Table 4-4 lists the coefficients, a, b, c, and d, used in the formulas to determine tall stack plume dispersion coefficients. The data in the tables was obtained from Turner, 1969, and the ASME, 1968.

4.6 Program Modifications

As previously mentioned, the program has been equipped with retrieval software to read the RAPS grid and emissions files. For adaptation of EMMOD to a region other than St. Louis where the data formats of available data are quite similar in structure, the user may redesign the retrieval software in subroutines GRIDIT, AREAEM, and STACKS to suit the data base.

The point source portion of the program is considered to be quite general. If the user designs a new subroutine STACKS to fill the appropriate arrays with the proper data, the program should run without difficulty. The user should note that the arrays in STACKS are presently dimensioned for 505 point sources per hour, and can easily be enlarged.

The area source portion of the program is not considered to be general enough to utilize without major modifications for most regions other than St. Louis. Users adapting the program to other regions may find it more convenient to acquire ERT's FLXGEN program. The FLXGEN program is a general-purpose regional emissions processor capable of generating area source emission schedules for the chemical-diffusion module. It assembles inventories for up to fifty different area source categories and uses constant-size grid square I and J indices instead of variable-size grid square identifiers.

TABLE 4-2

EMISSIONS MODULE INTERNALLY SPECIFIED DATA

Variable Name	Function	Location
ROAIR	Density of air (gm/meters^3) (currently set equal to 1178)	EMMOD
SIGEDG	The lateral edge of a Gaussian plume, defined by the number of standard deviations from the plume's centerline. (currently equals 2.14)	EMMOD
DTFREZ	The time interval for consideration of lateral diffusion from a point source (currently set at 60 minutes).	EMMOD
VDXFRC	The non-dimensional fraction of DTFREZ at which a plume's vertical distribution is evaluated (currently equals .33).	PONTEM
DTDZ	Potential temperature gradient (^{o}C/m) in the inversion layer (currently equals .0137).	PONTEM
NDY	The number of distances downwind for which lateral dispersion coefficients (σ_y) are tabulated (currently equals 4).	BLOCK DATA
XDY	The distances downwind corresponding to the tabulated lateral dispersion coefficients (km).	BLOCK DATA
SIGYD	The tabulated lateral dispersion coefficients (σ_y) as functions of atmospheric stability class and downwind distances (km).	BLOCK DATA
NDZ	The number of distances downwind for which vertical dispersion coefficients (σ_z) are tabulated (max = 12).	BLCOK DATA
XDZ	The distances downwind corresponding to the tabulated vertical dispersion coefficients (meters).	BLOCK DATA
SIGZD	The tabulated vertical dispersion coefficients (σ_z) as functions of atmospheric stability class and downwind distance (meters).	BLOCK DATA
NCAT	Number of tabulated atmospheric stability classes (max = 6).	BLOCK DATA
ATALL, BTALL	The "a" and "b" coefficients by stability class in the tall stack lateral dispersion coefficient formula $\sigma_y = ax^b$.	BLOCK DATA

TABLE 4-2 (CONTINUED)

Variable Name	Function	Location
CTALL, DTALL	The "c" and "d" coefficients by stability class in the tall stack vertical dispersion coefficient formula $\sigma_z = cx^d$.	BLOCK DATA

TABLE 4-3

PLUME DISPERSION COEFFICIENTS FOR NON-TALL STACKS

a) Lateral Dispersion Coefficients (σ_y in kilometers)

Distance Downwind (km)	Stability Class A	B	C	D	E	F
0.1	.027	.019	.0125	.008	.006	.004
1.0	.212	.157	.104	.068	.050	.034
10.0	1.57	1.19	.84	.555	.410	.273
80.0	9.0	6.8	5.05	3.35	2.73	1.68

b) Vertical Dispersion Coefficients (σ_z in meters)

Distance Downwind (km)	Stability Class A	B	C	D	E	F
0.1	14.	10.8	7.4	4.6	3.5	2.3
0.2	29.5	20.3	13.9	8.5	6.38	4.05
0.3	48.	30.2	20.1	12.1	8.8	5.6
0.5	105.	51.	32.	18.6	13.0	8.5
1.0	450.	110.	61.	31.5	21.3	14.0
2.0	1950.	232.	116.	50.0	33.7	21.5
3.0	4600.	365.	169.	64.5	42.7	26.5
5.0	13562.	640.	267.	89.0	56.0	34.0
10.0	58818.	1350.	500.	137.	79.0	46.5
20.0	255084.	2900.	950.	202.	110.	60.0
50.0	1774117.	7968.	2170.	328.	153.	79.0
100.0	7694071.	17117.	4000.	455.	183.	93.0

TABLE 4-4

PLUME DISPERSION COEFFICIENTS FOR TALL STACKS

Where: $\sigma_y = ax^b$

$\sigma_z = cx^d$

x = downwind distance

Coefficient	A	B	C & D	E & F
a	.40	.36	.32	.31
b	.91	.86	.78	.71
c	.40	.33	.22	.06
d	.91	.86	.78	.71

5. THE CHEMICAL-DIFFUSION MODULE

This section provides instructions for the use of the chemical-diffusion module program KEMOD. The basic function of the program is solution of the atmospheric diffusion equation in the Lagrangian reference frame for the pollutant concentrations at vertically discretized heights. The solution is obtained by numerically integrating finite difference approximations to the equation shown below:

$$\frac{\partial C_i(z,t)}{\partial t} = \frac{\partial}{\partial z}\left(K_z(z,t)\,\frac{\partial C_i(z,t)}{\partial z}\right) + \dot{S}_i + \dot{R}_i(C_i,C_j,\ldots,C_r) \quad (5\text{-}1)$$

Where:

$C_i(z,t)$ = concentration of ith species

$K_z(z,t)$ = eddy diffusivity coefficient

$\dot{S}_i$ = source emissions of ith species

$\dot{R}_i$ = chemical rate of change of ith species

z = elevation above ground

t = time

The program utilizes the schedules of emissions data, K_z coefficients, temperatures, and sky clearness parameters. In addition, the user provides initial concentrations and chemical rate constants for the model. The program solves the equations and outputs the concentrations as functions of time and height along the trajectory.

5.1 Control Parameters and Input Data

The input parameters will be discussed in the order in which they are input to the model. The reader should refer to the figures and table described below to obtain a better picture and understanding of the KEMOD input structure. Figure 5-1 illustrates the listing of the sample problem input deck produced by subroutine PREDAT. Table 5-1 lists the input variables by code variable name, and describes their functions and input formats. Figure 5-2 contains a flow chart of the input variable sequence and options. The user will find these aids quite helpful in developing input decks for the program.

TABLE 5-1

CHEMICAL-DIFFUSION MODULE INPUT VARIABLE DESCRIPTION

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
1	TITLE	Label for each page of output.	(20A4)	May contain any user comments in columns 1-80.
2	GOAHED	Chemistry and Diffusion are to be computed after copying the input stream (YES) or (NO).	(40X,A4)	
3	FLXUNT	Units of emission data equal to MOLE or MASS fraction.	(40X,A4)	When FLXUNT ≠ MOLE, the molecular weights in WTMOLE are used in the conversion of fluxes to a molal basis.
4	NUMFLX	Number of species with area source emissions.	(40X,I10)	Maximum is seven species.
5	FLXTIM, FLXIN(1,I) ...,FLXIN (7,I)	Area source emission times and rates in mole or mass fraction-meters/minute.	(20X,F10.2, 4E12.4,/, 30X,3E12.4)	The first card has a time and emissions data for four species. The second card has emission data for three species. The pair of inputs may be repeated up to 200 times.
6	NPSFLX	Number of species with point source emissions.	(40X,I10)	Maximum of seven species.
7	TPASS(I), PTSR(1,I), ..., PTSR (7,I),VFR (1,I)..., VFR(5,I)	Point source emission times and mole fraction or mass fraction-meters entrained into the air parcel. Vertical distribution (0. to 1.) of the emissions with respect to the vertical mesh points.	(20X,F10.2, 4E12.4,/, 30X,3E12.4,/, 18X,5E12.4)	The first card has the time and emissions data for four species. The second card has emission data for three species. The third card has the vertical distribution. This trio of cards is then repeated up to 200 times.

TABLE 5-1 (CONTINUED)

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
8	DELT	Initial value of integration time step size in minutes.	(40X,F10.0)	1.E-3>DELT>1.E-5
9	BIGSTP	Maximum integration time step size in minutes.	(40X,F10.0)	8.>BIGSTP>4.
10	EPS	Integration error control criteria.	(40X,F10.0)	1.E-3>EPS>1.E-4
11	UPDINT	The time interval in minutes at which the program checks the various schedules of time-varying parameters for new data.	(40X,F10.0)	20.>UPDINT>10.
12	PRNTIN	Time interval at which printed output is desired in minutes.	(40X,A4)	Must be equal or multiple of UPDINT.
13	TIMOUT	The integration progress variables are to be printed (YES) or (NO).	(40X,A4)	The integration index, last time step size, last order, number of steps, number of functions, and number of Jacobians are printed.
14	SUNGEN	The photodissociation rates are to be generated internally (YES) or (NO).	(40X,A4)	A schedule of NO_2 and HCHO photolysis rates must be read in if SUNGEN ≠ YES.
15	HIRATE	The photodissociation rates vary with elevation (YES or NO).	(40X,A4)	
16	QCLOUD	A schedule of sky clearness ratios are input (YES) or (NO).	(40X,A4)	Clear sky photodissociation rates are used when QCLOUD = NO.

TABLE 5-1 (CONTINUED)

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
17	VTEMP	A schedule of atmospheric temperatures are input (YES) or (NO).	(40X,A4)	T = 25°C is used when VTEMP = NO.
18	SPUNCH	The computed ground concentration histories are to be punched (YES) or (NO).	(40X,A4)	When SPUNCH = YES, the time concentration history is punched at each print interval in the format E12.4,/, 7(6E12.4,/).
19	NUMSTP	Maximum number of integration time steps.	(40X,I10)	NUMSTP should be equal to at least 75 steps per hour of real time simulation.
20	NOREAC	Number of chemical reactions	(40X,I10)	Equals 51 for the ERT mechanism.
21	NOSPEC	Total number species	(40X,I10)	Equal to 30 for the ERT mechanism.
22	NSTDY	Number of species approximated by the psuedo steady-state assumption or held constant.	(40X,I10)	Equal to 1 for the ERT mechanism.
23	NOSTAT	Number of vertical mesh points or cells.	(40X,I10)	$4 \leq NOSTAT \leq 5$. NOSTAT = 5 is recommended.
24	ZEE	Height of vertical mesh points in meters above the ground.	(40X,F10.0)	NOSTAT cards are read.
25	SPEC, WTMOLE, BCFLAG	Vectors of species names, molecular weights, and surface boundary condition codes.	(40X,A4,6X, F10.2,I10)	NOSPEC cards are read. See Section 5.1.3.
26	LOCFLX	A vector of species' indices corresponding to the area source emissions species.	(40X,I10)	NUMFLX cards read. See Section 5.1.4.

TABLE 5-1 (CONTINUED)

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
27	LOCPSF	A vector of species' indices corresponding to the point source emissions species.	(40X,I10)	NPSFLX cards used. See Section 5.1.4.
28	LOCDPF, DPRATE, DEPOWR	The index of a species with surface deposition, their deposition velocities in meters/min., and the exponent of the concentrations.	(40X,I10, 2F10.2)	The number cards must equal the number of species with BCFLAG = 1.
29	PPM	A two-dimensional array of initial concentrations of each species at each mesh point in ppm.	(20X,5E10.4)	NOSPEC cards read, with NOSTAT concentration for one species per card.
30	RATKON	A vector of rate constants in units consistent with the order of the reactions. (e.g., the units of a 2nd order reaction rate constant are ppm^{-1}-$minute^{-1}$).	(40X,F10.0)	NOREAC CARDS
31	REACT	An alphametric description of the chemical reaction mechanism.	(20A4)	NOREAC CARDS
32	RLAT	Latitude to be used in the zenith angle computation in degrees.	(40X,F10.0)	
33	RLONG	Longitude to be used in the zenith angle computation in degrees.	(40X,F10.0)	

TABLE 5-1 (CONTINUED)

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
34	TMZONE	Time zone of the region (e.g. 5. for New York and 8. for Los Angeles).	(40X,F10.0)	Also equal to the time difference between Greenwich, England, time and standard time in the modeling region.
35	JDATE	Date of day being simulated YRMODA (e.g., 760629 is June 29, 1976).	(40X,I10)	
36	NVRATE	Number of independent photodissociation rate arrays.	(40X,I10)	Maximum of two.
37	LOCVRT	The indices of the chemical reactions associated with the photodissociation rate arrays.	(40X,I10)	NVRATE cards
38	RATK1, RATK2	Arrays of the NO_2 and H_2CO photodissociation rates as a function of time and height.	(20X,5E12.4)	Required only if SUNGEN ≠ YES. RATK1 and RATK2 must be input on a time interval equal to UPDINT.
39	CLOUDT, CLOUDF	A table of sky clearness ratios as a function of time.	(40X,2F10.0)	Required only with QCLOUD = YES.
40	TMTEMP, TEMP	A table of temperatures (^{o}C) as a function of time.	(40X,2F10.0)	Required only with VTEMP = YES.
41	HTINV	A table of eddy diffusion coefficients (K_z) as a function of time and height.	(20X,F10.0, 10X,3F10.0,/, 20X,3F10.0)	

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
   1  ERT CHEMICAL-DIFFUSION MODULE SIMULATION    --   ST. LOUIS -  6/29/76   (181)
   2  DO WE COMPUTE CHEMICAL DIFFUSION          YES
   3  UNITS OF EMISSION FLUXES                  MOLE FRACTION
   4  NUMBER OF AREA SOURCE EMISSION SPECIE              7
   5  AREA SOURCES   1         360.00  0.           0.           0.           0.
   6                                   0.           0.           0.
   7  AREA SOURCES   2         394.40  6.2948E-10   4.7461E-11   3.0947E-11   2.1613E-11
   8                                   1.5232E-11   3.9946E-09   3.4242E-11
   9  AREA SOURCES   3         420.00  6.3056E-10   4.7468E-11   3.0953E-11   2.1616E-11
  10                                   1.5238E-11   3.9951E-09   3.4242E-11

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  11  AREA SOURCES   4         421.46  1.0695E-09   3.1599E-10   2.1080E-10   9.0539E-11
  12                                   5.7753E-11   1.6214E-08   4.7693E-11
  13  AREA SOURCES   5         426.39  1.3987E-09   3.8370E-10   2.8401E-10   1.3581E-10
  14                                   8.4703E-11   2.3710E-08   6.0059E-11
  15  AREA SOURCES   6         480.00  1.2794E-09   3.3902E-10   2.4868E-10   1.2101E-10
  16                                   7.5869E-11   2.0861E-08   5.6217E-11
  17  AREA SOURCES   7         486.07  1.1568E-09   4.8803E-10   2.7318E-10   1.0276E-10
  18                                   6.2245E-11   1.6862E-08   5.0526E-11
  19  AREA SOURCES   8         496.41  7.0699E-10   2.2365E-10   9.5628E-11   3.4384E-11
  20                                   1.9921E-11   4.7868E-09   3.6179E-11

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  21  AREA SOURCES   9         511.58  5.2604E-10   6.1504E-11   3.4266E-11   2.8534E-11
  22                                   1.6236E-11   4.1231E-09   2.7948E-11
  23  AREA SOURCES  10         521.06  5.3719E-10   1.2717E-10   5.9833E-11   2.6795E-11
  24                                   1.6348E-11   4.1926E-09   2.8485E-11
  25  AREA SOURCES  11         537.08  3.8360E-10   1.2554E-10   5.1388E-11   2.6212E-11
  26                                   1.2376E-11   2.9728E-09   1.9811E-11
  27  AREA SOURCES  12         540.00  3.8517E-10   1.1836E-10   4.8767E-11   2.6215E-11
  28                                   1.2384E-11   2.9734E-09   1.9811E-11
  29  AREA SOURCES  13         551.01  5.4923E-10   9.9462E-10   4.6101E-10   1.4888E-10
  30                                   9.6169E-11   9.8887E-09   3.1663E-11

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  31  AREA SOURCES  14         556.62  1.1345E-09   6.7646E-11   3.6272E-11   3.0139E-11
  32                                   2.0793E-11   4.2536E-09   6.2029E-11
  33  AREA SOURCES  15         564.13  2.4895E-10   4.7590E-11   1.8350E-11   2.2275E-11
  34                                   8.6010E-12   2.0085E-09   1.2472E-11
  35  AREA SOURCES  16         570.04  2.4895E-10   4.7590E-11   1.8350E-11   2.2275E-11
  36                                   8.6010E-12   2.0085E-09   1.2472E-11
  37  AREA SOURCES  17         571.63  4.7047E-10   2.2978E-10   6.0799E-11   1.0862E-10
  38                                   2.7733E-11   4.9042E-09   1.3762E-11
  39  AREA SOURCES  18         579.14  9.8934E-10   1.1668E-09   3.5381E-10   2.8772E-10
  40                                   6.8099E-11   1.0966E-08   1.9000E-11
```

Figure 5-1 Chemical-Diffusion Module Sample Problem Input Listing

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  41  AREA SOURCES  19        586.65  6.8719E-09  5.9080E-09  3.2800E-09  1.0595E-09
  42                                  6.5059E-10  1.9173E-07  1.8814E-10
  43  AREA SOURCES  20        589.06  1.4868E-09  3.3767E-09  1.2984E-09  2.4808E-10
  44                                  9.4978E-11  2.1869E-08  3.8482E-11
  45  AREA SOURCES  21        594.15  3.3096E-09  3.5885E-09  2.6686E-09  1.5946E-09
  46                                  1.0644E-09  1.3795E-07  1.3013E-10
  47  AREA SOURCES  22        598.58  6.3348E-09  1.2495E-08  7.3298E-09  3.0559E-09
  48                                  2.1032E-09  2.7206E-07  2.6124E-10
  49  AREA SOURCES  23        600.00  6.8837E-09  1.2722E-08  7.5204E-09  3.1374E-09
  50                                  2.1592E-09  2.8914E-07  2.7658E-10

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  51  AREA SOURCES  24        601.97  6.9314E-09  4.3280E-09  3.0233E-09  1.4255E-09
  52                                  8.6568E-10  2.4549E-07  2.2275E-10
  53  AREA SOURCES  25        610.88  6.9818E-10  6.5939E-10  2.0154E-10  1.8025E-10
  54                                  4.5322E-11  7.5543E-09  1.5773E-11
  55  AREA SOURCES  26        614.85  1.9784E-08  1.1513E-08  7.9774E-09  3.1679E-09
  56                                  2.0370E-09  5.8296E-07  6.4022E-10
  57  AREA SOURCES  27        619.79  6.9015E-10  3.8022E-10  9.8824E-11  1.7992E-10
  58                                  4.4960E-11  7.5380E-09  1.5773E-11
  59  AREA SOURCES  28        628.70  1.4073E-08  7.3394E-09  6.3035E-09  2.5720E-09
  60                                  1.9403E-09  6.3982E-07  3.2850E-10

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  61  AREA SOURCES  29        630.90  3.5109E-08  1.9592E-08  1.5709E-08  6.1848E-09
  62                                  4.5420E-09  1.4549E-06  9.1715E-10
  63  AREA SOURCES  30        637.61  6.9160E-09  1.1519E-08  7.4915E-09  3.8365E-09
  64                                  2.5417E-09  4.0971E-07  2.5992E-10
  65  AREA SOURCES  31        646.52  7.0907E-08  1.6750E-08  1.2083E-08  5.0369E-09
  66                                  3.6717E-09  1.0389E-06  2.9897E-09
  67  AREA SOURCES  32        646.94  8.5201E-08  3.5487E-08  2.6511E-08  1.1264E-08
  68                                  7.4444E-09  2.0575E-06  2.3277E-09
  69  AREA SOURCES  33        655.43  5.0874E-08  3.6667E-08  2.4756E-08  9.8677E-09
  70                                  6.4254E-09  1.9298E-06  1.5394E-09

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  71  AREA SOURCES  34        660.00  5.5525E-08  3.9251E-08  2.6834E-08  1.0684E-08
  72                                  7.0150E-09  2.1113E-06  1.6813E-09
  73  AREA SOURCES  35        663.44  2.1423E-08  1.5587E-08  1.0238E-08  5.0057E-09
  74                                  2.8840E-09  8.2212E-07  6.8713E-10
  75  AREA SOURCES  36        664.20  9.7319E-08  2.7934E-08  2.1000E-08  1.1056E-08
  76                                  6.3545E-09  2.4187E-06  8.6423E-09
  77  AREA SOURCES  37        672.81  8.9402E-08  3.9616E-08  2.6604E-08  1.2319E-08
  78                                  7.4717E-09  2.0123E-06  2.3221E-09
  79  AREA SOURCES  38        681.42  2.2408E-07  6.7122E-08  4.9568E-08  2.1947E-08
  80                                  1.4158E-08  3.5628E-06  5.0201E-09
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  81  AREA SOURCES  39        681.90  1.2276E-08  9.4233E-09  5.0348E-09  3.9136E-09
  82                                  1.6197E-09  3.9667E-07  4.3259E-10
  83  AREA SOURCES  40        685.72  1.4732E-07  4.4158E-08  3.4444E-08  1.4622E-08
  84                                  9.7926E-09  2.5547E-06  4.0982E-09
  85  AREA SOURCES  41        690.03  2.4403E-07  7.2917E-08  5.6342E-08  2.4167E-08
  86                                  1.6213E-08  4.3681E-06  5.9780E-09
  87  AREA SOURCES  42        691.13  3.0036E-08  1.6861E-08  7.5976E-09  7.1257E-09
  88                                  2.4762E-09  6.0171E-07  1.1920E-09
  89  AREA SOURCES  43        694.33  7.7762E-08  4.1972E-08  2.4156E-08  9.7636E-09
  90                                  5.5584E-09  1.4500E-06  2.3799E-09

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  91  AREA SOURCES  44        698.64  6.6096E-07  1.0102E-07  6.9284E-08  3.1858E-08
  92                                  1.9275E-08  7.1032E-06  2.6708E-08
  93  AREA SOURCES  45        700.36  4.7469E-08  3.9551E-08  1.9406E-08  9.9850E-09
  94                                  4.4210E-09  1.1573E-06  1.7010E-09
  95  AREA SOURCES  46        702.94  2.6406E-07  1.0139E-07  7.0569E-08  3.0225E-08
  96                                  2.0506E-08  4.4130E-06  7.3346E-09
  97  AREA SOURCES  47        707.25  6.0129E-08  2.0018E-08  1.5328E-08  9.4348E-09
  98                                  7.6152E-09  5.1955E-07  4.6735E-09
  99  AREA SOURCES  48        709.59  1.0194E-06  1.6117E-07  1.2060E-07  5.5458E-08
 100                                  3.4766E-08  1.3522E-05  4.1097E-08

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 101  AREA SOURCES  49        711.55  8.3940E-08  4.8356E-08  3.1299E-08  1.4111E-08
 102                                  1.1100E-08  1.0924E-06  4.6711E-09
 103  AREA SOURCES  50        715.85  3.6132E-08  4.4707E-09  3.2550E-09  1.7789E-09
 104                                  1.7254E-09  3.7041E-07  2.5173E-09
 105  AREA SOURCES  51        720.00  3.6127E-08  4.4706E-09  3.2550E-09  1.7789E-09
 106                                  1.7254E-09  3.7041E-07  2.5172E-09
 107  AREA SOURCES  52        723.60  6.4366E-08  1.6453E-08  1.2999E-08  7.6082E-09
 108                                  6.9096E-09  4.3301E-07  5.3222E-09
 109  AREA SOURCES  53        723.89  8.2130E-08  4.4793E-09  3.8673E-09  1.9805E-09
 110                                  1.5649E-09  7.6500E-07  4.5628E-09

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 111  AREA SOURCES  54        727.64  1.6197E-07  3.0791E-08  2.6676E-08  1.1577E-08
 112                                  8.3109E-09  3.3317E-06  6.8794E-09
 113  AREA SOURCES  55        727.72  1.2944E-07  3.5657E-08  2.8664E-08  1.1405E-08
 114                                  8.6244E-09  3.1642E-06  5.2913E-09
 115  AREA SOURCES  56        731.38  2.9946E-08  1.2529E-08  5.6089E-09  2.4802E-09
 116                                  1.9029E-09  2.5645E-07  2.1766E-09
 117  AREA SOURCES  57        731.85  2.0728E-08  1.5666E-08  7.7027E-09  3.6459E-09
 118                                  1.9100E-09  4.6731E-07  8.0803E-10
 119  AREA SOURCES  58        735.13  1.9631E-07  4.7822E-08  3.3037E-08  1.5582E-08
 120                                  9.9254E-09  3.9886E-06  8.0359E-09
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 121  AREA SOURCES  59        735.97   4.6300E-08  1.0985E-07  4.5899E-08  8.7987E-09
 122                                    4.8330E-09  1.2811E-06  1.6610E-09
 123  AREA SOURCES  60        738.88   8.1440E-08  4.2957E-08  2.8854E-08  1.3311E-08
 124                                    8.1882E-09  2.1326E-06  3.2988E-09
 125  AREA SOURCES  61        740.10   4.7636E-08  2.0120E-08  1.2847E-08  4.9779E-09
 126                                    3.5494E-09  9.2125E-07  1.5318E-09
 127  AREA SOURCES  62        742.63   2.2528E-08  1.2990E-08  6.5250E-09  2.7563E-09
 128                                    1.7477E-09  3.7916E-07  1.1440E-09
 129  AREA SOURCES  63        744.22   6.7886E-08  2.0953E-08  1.5628E-08  5.6098E-09
 130                                    5.9397E-09  1.0349E-06  6.3957E-09

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 131  AREA SOURCES  64        746.38   5.1974E-08  9.6121E-09  5.5713E-09  1.4503E-09
 132                                    3.3554E-09  1.2986E-07  6.6724E-09
 133  AREA SOURCES  65        748.35   1.7501E-08  7.1070E-09  5.1910E-09  2.3606E-09
 134                                    1.4522E-09  4.0337E-07  6.4837E-10
 135  AREA SOURCES  66        750.13   5.5456E-08  1.0458E-08  6.0765E-09  2.1443E-09
 136                                    2.1263E-09  5.2400E-07  3.8953E-09
 137  AREA SOURCES  67        752.47   2.9751E-08  8.5048E-09  6.4970E-09  2.5843E-09
 138                                    1.8640E-09  4.8754E-07  8.0619E-10
 139  AREA SOURCES  68        753.87   4.2545E-08  2.4172E-08  1.8510E-08  6.9910E-09
 140                                    5.0765E-09  1.5675E-06  1.2504E-09

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 141  AREA SOURCES  69        756.60   1.1158E-07  4.1985E-08  3.2555E-08  1.1986E-08
 142                                    9.0309E-09  2.6394E-06  2.6261E-09
 143  AREA SOURCES  70        757.62   7.7618E-08  2.4618E-08  1.6697E-08  5.7647E-09
 144                                    4.1363E-09  9.9341E-07  1.9096E-09
 145  AREA SOURCES  71        760.72   2.6556E-08  1.2410E-08  1.1216E-08  4.4364E-09
 146                                    3.5042E-09  1.1621E-06  6.8794E-10
 147  AREA SOURCES  72        761.37   2.2253E-09  2.4085E-10  9.9396E-11  7.9723E-11
 148                                    5.2463E-11  1.0590E-08  1.2042E-10
 149  AREA SOURCES  73        764.85   9.7855E-09  3.8962E-09  3.0311E-09  1.1224E-09
 150                                    9.1493E-10  2.7167E-07  3.8702E-10

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 151  AREA SOURCES  74        780.00   1.0082E-08  4.0294E-09  3.1500E-09  1.1693E-09
 152                                    9.5131E-10  2.8343E-07  3.9404E-10
 153  AREA SOURCES  75        780.14   1.6879E-08  4.7010E-09  3.7089E-09  1.4773E-09
 154                                    1.1489E-09  2.9758E-07  5.5979E-10
 155  AREA SOURCES  76        786.01   1.2894E-08  5.8629E-09  4.2625E-09  1.5699E-09
 156                                    1.1814E-09  3.5647E-07  4.7985E-10
 157  AREA SOURCES  77        804.21   1.8912E-08  5.0814E-09  3.6166E-09  1.3528E-09
 158                                    1.0023E-09  2.3522E-07  5.8661E-10
 159  AREA SOURCES  78        808.68   4.2857E-09  6.7493E-10  3.5794E-10  1.1574E-10
 160                                    2.1933E-10  1.1440E-08  4.0399E-10
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 161  AREA SOURCES  79        809.03  2.0977E-09  2.0957E-10  9.1448E-11  7.0555E-11
 162                                   4.8433E-11  1.0015E-08  1.1491E-10
 163  AREA SOURCES  80        817.74  2.1125E-09  5.0296E-10  1.9941E-10  7.0904E-11
 164                                   4.8863E-11  1.0034E-08  1.1491E-10
 165  AREA SOURCES  81        818.65  1.5446E-08  6.6220E-09  3.9139E-09  1.1770E-09
 166                                   8.1311E-10  2.0427E-07  4.1824E-10
 167  AREA SOURCES  82        826.81  1.0788E-08  2.2318E-09  1.6973E-09  6.9017E-10
 168                                   5.0024E-10  1.1919E-07  3.2680E-10
 169  AREA SOURCES  83        828.28  3.0009E-09  1.6689E-09  7.6653E-10  1.4068E-10
 170                                   1.0657E-10  2.7944E-08  1.5557E-10

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 171  AREA SOURCES  84        831.34  9.7101E-09  2.0080E-09  1.5118E-09  6.0814E-10
 172                                   4.4745E-10  1.0672E-07  3.1587E-10
 173  AREA SOURCES  85        840.00  1.1121E-08  2.3653E-09  1.7899E-09  7.1755E-10
 174                                   5.2452E-10  1.2570E-07  3.4617E-10
 175  AREA SOURCES  86        849.08  5.9993E-09  1.4448E-09  9.9115E-10  3.7468E-10
 176                                   2.7190E-10  6.9699E-08  2.3467E-10
 177  AREA SOURCES  87        858.58  6.4817E-09  1.4557E-09  1.0764E-09  4.3601E-10
 178                                   3.2000E-10  8.3000E-08  2.6381E-10
 179  AREA SOURCES  88        866.76  0.          0.          0.          0.
 180                                   0.          0.          0.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 181  AREA SOURCES  89        900.00  0.          0.          0.          0.
 182                                   0.          0.          0.
 183  AREA SOURCES  90        960.00  0.          0.          0.          0.
 184                                   0.          0.          0.
 185  AREA SOURCES  91       1020.00  0.          0.          0.          0.
 186                                   0.          0.          0.
 187  LAST AREA SOURCES       -10.00
 188  LAST AREA SOURCES       -10.00
 189  NUMBER OF POINT SOURCE EMISSION SPECIE          7
 190  POINT SOURCES  1        709.80  0.          9.2240E-08  3.3810E-08  0.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 191                                   0.          0.          0.
 192                  1    1.2828E-01  2.9127E-01  3.6558E-01  2.1488E-01  0.
 193  POINT SOURCES  2        711.40  7.6917E-07  0.          0.          0.
 194                                   0.          0.          1.9874E-06
 195                  2    1.7865E-01  2.0977E-01  4.5516E-01  1.5641E-01  0.
 196  POINT SOURCES  3        711.48  4.4139E-08  3.2283E-09  6.9287E-09  4.5684E-10
 197                                   1.3857E-09  1.2690E-06  3.9656E-08
 198                  3    1.3890E-01  2.2231E-01  4.2099E-01  2.1701E-01  0.
 199  POINT SOURCES  4        711.50  4.4175E-08  3.2310E-09  6.9344E-09  4.5722E-10
 200                                   1.3869E-09  1.2700E-06  3.9689E-08
```

Figure 5-1 (Continued)

```
      1         11          21          31          41          51          61          71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 201                    4  1.3889E-01  2.2231E-01  4.2097E-01  2.1783E-01  0.
 202  POINT SOURCES     5       719.64  0.          5.3696E-08  1.9682E-08  0.
 203                                    0.          0.          0.
 204                    5  1.3251E-01  3.0197E-01  3.8773E-01  1.7778E-01  0.
 205  POINT SOURCES     6       721.87  0.          1.4858E-07  0.          2.0688E-08
 206                                    0.          0.          0.
 207                    6  1.4159E-01  3.2003E-01  4.0849E-01  1.2988E-01  0.
 208  POINT SOURCES     7       721.87  0.          2.2484E-11  0.          5.2169E-12
 209                                    0.          0.          0.
 210                    7  1.4159E-01  3.2003E-01  4.0849E-01  1.2988E-01  0.

      1         11          21          31          41          51          61          71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 211  POINT SOURCES     8       722.70  0.          1.5356E-07  5.6287E-08  0.
 212                                    0.          0.          0.
 213                    8  1.5620E-01  3.0831E-01  3.4087E-01  1.9462E-01  0.
 214  POINT SOURCES     9       722.70  0.          2.8933E-08  1.0605E-08  0.
 215                                    0.          0.          0.
 216                    9  1.5333E-01  3.0859E-01  3.4212E-01  1.9596E-01  0.
 217  POINT SOURCES    10       725.51  0.          8.1482E-08  2.9866E-08  0.
 218                                    0.          0.          0.
 219                   10  1.3294E-01  3.0126E-01  3.8733E-01  1.7847E-01  0.
 220  POINT SOURCES    11       728.18  0.          0.          0.          0.

      1         11          21          31          41          51          61          71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 221                                    0.          0.          6.4126E-07
 222                   11  1.5995E-01  2.9466E-01  3.2098E-01  2.2442E-01  0.
 223  POINT SOURCES    12       728.42  6.8132E-10  9.4022E-13  9.4022E-13  3.1341E-13
 224                                    9.4022E-13  1.5857E-10  2.4485E-12
 225                   12  1.6146E-01  3.1331E-01  3.4446E-01  1.8077E-01  0.
 226  POINT SOURCES    13       728.42  6.8132E-10  9.4022E-13  9.4022E-13  3.1341E-13
 227                                    9.4022E-13  1.5857E-10  2.4485E-12
 228                   13  1.6146E-01  3.1331E-01  3.4446E-01  1.8077E-01  0.
 229  POINT SOURCES    14       728.42  6.8132E-10  9.4022E-13  9.4022E-13  3.1341E-13
 230                                    9.4022E-13  1.5857E-10  2.4485E-12

      1         11          21          31          41          51          61          71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 231                   14  1.6146E-01  3.1331E-01  3.4446E-01  1.8077E-01  0.
 232  POINT SOURCES    15       728.69  0.          1.1246E-08  4.1222E-09  0.
 233                                    0.          0.          0.
 234                   15  1.3385E-01  3.0337E-01  3.8761E-01  1.7518E-01  0.
 235  POINT SOURCES    16       728.95  0.          0.          0.          0.
 236                                    0.          0.          3.6272E-08
 237                   16  1.5899E-01  3.0905E-01  3.3927E-01  1.9269E-01  0.
 238  POINT SOURCES    17       728.97  0.          0.          0.          0.
 239                                    0.          0.          2.4949E-08
 240                   17  1.6733E-01  3.0824E-01  3.3558E-01  1.8885E-01  0.
```

Figure 5-1 (Continued)

```
      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 241  POINT SOURCES 18          729.04  0.         0.         0.         0.
 242                                    0.         0.         1.0422E-06
 243                18     1.6734E-01  3.0825E-01  3.3558E-01  1.8882E-01  0.
 244  POINT SOURCES 19          729.29  0.         0.         0.         0.
 245                                    0.         0.         2.4811E-07
 246                19     1.6385E-01  3.0867E-01  3.3716E-01  1.9031E-01  0.
 247  POINT SOURCES 20          729.30  0.         0.         0.         3.6492E-09
 248                                    0.         0.         0.
 249                20     2.6556E-01  2.1556E-01  3.3444E-01  1.8443E-01  0.
 250  POINT SOURCES 21          729.30  0.         0.         0.         1.8246E-08

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 251                                    0.         0.         0.
 252                21     2.6556E-01  2.1556E-01  3.3444E-01  1.8443E-01  0.
 253  POINT SOURCES 22          729.30  0.         0.         0.         9.2889E-09
 254                                    0.         0.         0.
 255                22     1.5009E-01  2.9552E-01  3.2529E-01  2.2910E-01  0.
 256  POINT SOURCES 23          729.30  0.         0.         0.         3.9312E-08
 257                                    0.         0.         0.
 258                23     2.5326E-01  2.0558E-01  3.1895E-01  2.2221E-01  0.
 259  POINT SOURCES 24          729.30  0.         0.         0.         3.5663E-08
 260                                    0.         0.         0.

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 261                24     1.5162E-01  2.9538E-01  3.2462E-01  2.2838E-01  0.
 262  POINT SOURCES 25          729.30  0.         0.         0.         3.6492E-09
 263                                    0.         0.         0.
 264                25     2.5326E-01  2.0558E-01  3.1895E-01  2.2221E-01  0.
 265  POINT SOURCES 26          729.30  0.         0.         0.         2.1564E-09
 266                                    0.         0.         0.
 267                26     1.4824E-01  2.9569E-01  3.2610E-01  2.2997E-01  0.
 268  POINT SOURCES 27          730.04  2.4348E-07  8.9601E-10  8.9601E-10  2.9867E-10
 269                                    8.9601E-10  5.3334E-08  3.6797E-07
 270                27     1.6425E-01  1.7565E-01  4.2315E-01  2.3695E-01  0.

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 271  POINT SOURCES 28          730.11  8.3123E-08  2.2942E-11  2.2942E-11  7.6474E-12
 272                                    2.2942E-11  3.8692E-09  5.9744E-11
 273                28     1.7395E-01  1.6796E-01  4.4093E-01  2.1716E-01  0.
 274  POINT SOURCES 29          730.43  5.7404E-08  2.1125E-10  2.1125E-10  7.0417E-11
 275                                    2.1125E-10  1.2574E-08  8.6755E-08
 276                29     1.7551E-01  1.6628E-01  4.4360E-01  2.1461E-01  0.
 277  POINT SOURCES 30          730.59  5.2771E-08  1.9420E-10  1.9420E-10  6.4732E-11
 278                                    1.9420E-10  1.1559E-08  7.9752E-08
 279                30     1.7647E-01  1.6521E-01  4.4524E-01  2.1308E-01  0.
 280  POINT SOURCES 31          731.29  4.7517E-09  4.3715E-12  4.3715E-12  1.4572E-12
```

Figure 5-1 (Continued)

```
      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 281                                  4.3715E-12 4.3368E-10 2.0404E-08
 282                31    1.5967E-01 2.0883E-01 4.7483E-01 1.5667E-01 0.
 283  POINT SOURCES 32         731.32 2.8522E-07 7.8721E-11 7.8721E-11 2.6240E-11
 284                                  7.8721E-11 1.3276E-08 2.0500E-10
 285                32    1.4429E-01 2.0637E-01 4.3343E-01 2.1591E-01 0.
 286  POINT SOURCES 33         731.63 4.7094E-09 4.3327E-12 4.3327E-12 1.4442E-12
 287                                  4.3327E-12 4.2982E-10 2.0223E-08
 288                33    1.5953E-01 2.0890E-01 4.7459E-01 1.5698E-01 0.
 289  POINT SOURCES 34         732.00 2.8082E-07 7.7508E-11 7.7508E-11 2.5836E-11
 290                                  7.7508E-11 1.3072E-08 2.0184E-10

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 291                34    1.4113E-01 2.1599E-01 4.2728E-01 2.1560E-01 0.
 292  POINT SOURCES 35         733.89 0.         0.         0.         8.0997E-07
 293                                  0.         1.8246E-05 0.
 294                35    1.6241E-01 3.0735E-01 3.3717E-01 1.9307E-01 0.
 295  POINT SOURCES 36         734.71 8.9666E-08 3.2997E-10 3.2997E-10 1.0999E-10
 296                                  3.2997E-10 1.9641E-08 5.0286E-07
 297                36    1.7464E-01 1.6727E-01 4.4244E-01 2.1565E-01 0.
 298  POINT SOURCES 37         735.11 0.         0.         0.         0.
 299                                  0.         0.         1.7949E-08
 300                37    1.6668E-01 2.1072E-01 4.8124E-01 1.4136E-01 0.

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 301  POINT SOURCES 38         735.15 6.9893E-10 1.0128E-10 8.3046E-09 4.7262E-10
 302                                  0.         3.6456E-10 5.0236E-08
 303                38    1.5875E-01 3.1832E-01 3.6481E-01 1.5813E-01 0.
 304  POINT SOURCES 39         735.39 0.         3.6550E-08 0.         5.0892E-09
 305                                  0.         0.         0.
 306                39    1.5077E-01 2.9744E-01 3.2587E-01 2.2591E-01 0.
 307  POINT SOURCES 40         735.58 0.         1.7957E-09 6.5819E-10 0.
 308                                  0.         0.         0.
 309                40    1.5427E-01 3.1039E-01 3.4256E-01 1.9278E-01 0.
 310  POINT SOURCES 41         735.58 0.         2.6442E-08 9.6920E-09 0.

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 311                                  0.         0.         0.
 312                41    1.5427E-01 3.1039E-01 3.4256E-01 1.9278E-01 0.
 313  POINT SOURCES 42         735.58 0.         1.2432E-09 4.5569E-10 0.
 314                                  0.         0.         0.
 315                42    1.5427E-01 3.1039E-01 3.4256E-01 1.9278E-01 0.
 316  POINT SOURCES 43         735.58 0.         5.0280E-08 1.8429E-08 0.
 317                                  0.         0.         0.
 318                43    1.5427E-01 3.1039E-01 3.4256E-01 1.9278E-01 0.
 319  POINT SOURCES 44         735.58 0.         4.4746E-08 1.6401E-08 0.
 320                                  0.         0.         0.
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  321                  44    1.5814E-01  3.1004E-01  3.4086E-01  1.9096E-01  0.
  322 POINT SOURCES  45        735.97  6.3644E-10  9.2221E-13  7.5621E-11  4.3036E-12
  323                                   0.          3.3197E-10  4.5745E-08
  324                  45    1.5917E-01  3.1953E-01  3.6474E-01  1.5656E-01  0.
  325 POINT SOURCES  46        736.41  1.2917E-08  9.3005E-12  9.3005E-12  3.1002E-12
  326                                   9.3005E-12  1.5685E-09  2.4220E-11
  327                  46    1.6205E-01  3.1071E-01  3.3899E-01  1.8825E-01  0.
  328 POINT SOURCES  47        737.04  1.6074E-08  1.1573E-11  1.1573E-11  3.8576E-12
  329                                   1.1573E-11  1.9518E-09  3.0138E-11
  330                  47    1.7613E-01  1.6564E-01  4.4498E-01  2.1325E-01  0.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  331 POINT SOURCES  48        737.35  1.7640E-08  1.2701E-11  1.2701E-11  4.2336E-12
  332                                   1.2701E-11  2.1420E-09  3.3074E-11
  333                  48    1.7585E-01  1.6596E-01  4.4451E-01  2.1369E-01  0.
  334 POINT SOURCES  49        737.51  1.3700E-08  9.8642E-12  9.8642E-12  3.2881E-12
  335                                   9.8642E-12  1.6636E-09  2.5688E-11
  336                  49    1.7077E-01  3.0615E-01  3.3291E-01  1.9017E-01  0.
  337 POINT SOURCES  50        739.81  4.7400E-09  3.4128E-12  3.4128E-12  1.1376E-12
  338                                   3.4128E-12  5.7556E-10  4.4437E-13
  339                  50    1.5421E-01  3.3138E-01  3.7875E-01  1.3567E-01  0.
  340 POINT SOURCES  51        740.23  8.3047E-08  5.9794E-11  5.9794E-11  1.9931E-11

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  341                                   5.9794E-11  1.0084E-08  1.5571E-10
  342                  51    1.5144E-01  1.9266E-01  4.0274E-01  2.5316E-01  0.
  343 POINT SOURCES  52        740.23  6.3142E-08  0.          0.          0.
  344                                   0.          0.          0.
  345                  52    1.6552E-01  3.0916E-01  3.3587E-01  1.8945E-01  0.
  346 POINT SOURCES  53        740.32  4.4604E-09  3.2115E-12  3.2115E-12  1.0705E-12
  347                                   3.2115E-12  5.4162E-10  8.3633E-12
  348                  53    1.5286E-01  3.3208E-01  3.7954E-01  1.3552E-01  0.
  349 POINT SOURCES  54        740.60  4.3951E-09  3.1645E-12  3.1645E-12  1.0548E-12
  350                                   3.1645E-12  5.3369E-10  4.1204E-13

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  351                  54    1.4938E-01  3.3226E-01  3.8140E-01  1.3696E-01  0.
  352 POINT SOURCES  55        741.83  6.9778E-08  0.          0.          0.
  353                                   0.          0.          0.
  354                  55    1.6375E-01  3.0930E-01  3.3662E-01  1.9033E-01  0.
  355 POINT SOURCES  56        741.88  0.          0.          0.          0.
  356                                   0.          0.          1.1199E-07
  357                  56    1.5627E-01  3.3154E-01  3.7767E-01  1.3452E-01  0.
  358 POINT SOURCES  57        742.24  1.9817E-08  5.4696E-11  5.4696E-11  1.8232E-11
  359                                   5.4696E-11  2.1705E-09  7.4542E-08
  360                  57    1.7204E-01  1.8905E-01  4.8182E-01  1.5709E-01  0.
```

Figure 5-1 (Continued)

```
      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 361  POINT SOURCES 58         742.30   5.7709E-08  0.          0.          0.
 362                                    0.          0.          2.6396E-11
 363                   58    1.6635E-01  3.1006E-01  3.3670E-01  1.8690E-01  0.
 364  POINT SOURCES 59         742.31   1.0225E-08  3.7627E-11  3.7627E-11  1.2542E-11
 365                                    3.7627E-11  2.2397E-09  6.1438E-08
 366                   59    1.4436E-01  3.3200E-01  3.8401E-01  1.3963E-01  0.
 367  POINT SOURCES 60         742.37   1.3872E-08  5.1049E-11  5.1049E-11  1.7016E-11
 368                                    5.1049E-11  3.0386E-09  8.3353E-08
 369                   60    1.6622E-01  3.3448E-01  3.7935E-01  1.1995E-01  0.
 370  POINT SOURCES 61         742.38   2.3494E-08  6.4843E-12  6.4843E-12  2.1614E-12

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 371                                    6.4843E-12  1.0936E-09  1.6886E-11
 372                   61    1.7287E-01  1.8789E-01  4.8306E-01  1.5618E-01  0.
 373  POINT SOURCES 62         742.51   1.5186E-08  5.5887E-11  5.5887E-11  1.8629E-11
 374                                    5.5887E-11  3.3266E-09  9.1252E-08
 375                   62    1.7186E-01  1.8926E-01  4.8146E-01  1.5741E-01  0.
 376  POINT SOURCES 63         742.51   2.3189E-08  6.4001E-12  6.4001E-12  2.1334E-12
 377                                    6.4001E-12  1.0794E-09  1.6667E-11
 378                   63    1.7289E-01  1.8789E-01  4.8308E-01  1.5615E-01  0.
 379  POINT SOURCES 64         742.70   0.          0.          0.          0.
 380                                    0.          0.          1.3493E-07

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 381                   64    1.6395E-01  3.3004E-01  3.7344E-01  1.3257E-01  0.
 382  POINT SOURCES 65         751.10   1.2966E-08  1.1260E-11  1.1260E-11  3.7533E-12
 383                                    1.1260E-11  1.5632E-09  6.6436E-10
 384                   65    1.7301E-01  1.7207E-01  4.3993E-01  2.1498E-01  0.
 385  POINT SOURCES 66         751.10   1.2966E-08  1.1260E-11  1.1260E-11  3.7533E-12
 386                                    1.1260E-11  1.5632E-09  6.6436E-10
 387                   66    1.7301E-01  1.7207E-01  4.3993E-01  2.1498E-01  0.
 388  POINT SOURCES 67         754.31   5.4798E-09  3.9455E-12  3.9455E-12  1.3152E-12
 389                                    3.9455E-12  6.6540E-10  1.0274E-11
 390                   67    1.6241E-01  3.1530E-01  3.4066E-01  1.8162E-01  0.

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 391  POINT SOURCES 68         760.71   3.2740E-08  1.5862E-11  1.5862E-11  5.2875E-12
 392                                    1.5862E-11  1.6970E-09  4.8280E-09
 393                   68    1.7791E-01  1.7212E-01  4.5038E-01  1.9959E-01  0.
 394  POINT SOURCES 69         766.27   0.          3.4072E-09  5.4929E-11  1.5794E-09
 395                                    0.          0.          0.
 396                   69    1.4929E-01  2.9969E-01  3.2750E-01  2.2352E-01  0.
 397  POINT SOURCES 70         768.67   5.5572E-08  5.8430E-11  5.8430E-11  1.9477E-11
 398                                    5.8430E-11  2.6085E-09  2.0007E-07
 399                   70    1.7277E-01  1.7219E-01  4.3982E-01  2.1522E-01  0.
 400  POINT SOURCES 71         774.38   0.          6.2820E-10  2.3026E-10  0.
```

Figure 5-1 (Continued)

```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 401                                 0.          0.          0.
 402                 71    1.5718E-01  3.0883E-01  3.3990E-01  1.9409E-01  0.
 403 POINT SOURCES 72          800.86  1.8367E-09  5.0694E-12  5.0694E-12  1.6898E-12
 404                                 5.0694E-12  2.0117E-10  1.2498E-09
 405                 72    1.6603E-01  3.1068E-01  3.3514E-01  1.8815E-01  0.
 406 POINT SOURCES 73          821.76  2.1268E-09  2.2362E-12  2.2362E-12  7.4539E-13
 407                                 2.2362E-12  9.9828E-11  2.0964E-10
 408                 73    1.6694E-01  3.2300E-01  3.4218E-01  1.6788E-01  0.
 409 POINT SOURCES 74          821.87  2.1023E-09  2.2104E-12  2.2104E-12  7.3680E-13
 410                                 2.2104E-12  9.8678E-11  2.0723E-10

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 411                 74    1.6693E-01  3.2297E-01  3.4217E-01  1.6793E-01  0.
 412 POINT SOURCES 75          822.07  2.0650E-09  2.1712E-12  2.1712E-12  7.2375E-13
 413                                 2.1712E-12  9.6930E-11  2.0355E-10
 414                 75    1.6692E-01  3.2294E-01  3.4216E-01  1.6798E-01  0.
 415 POINT SOURCES 76          822.18  2.0411E-09  2.1461E-12  2.1461E-12  7.1536E-13
 416                                 2.1461E-12  9.5806E-11  2.0119E-10
 417                 76    1.6691E-01  3.2291E-01  3.4216E-01  1.6803E-01  0.
 418 LAST PTSR                -100.00
 419 LAST PTSR
 420 INITIAL INTEGRATION TIME STEP SIZE          1.0E-04      (MINUTES)

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 421 MAXIMUM INTEGRATION TIME STEP SIZE       6.0             (MINUTES)
 422 INTEGRATION ERROR CONTROL CRITERIA          1.0E-03
 423 UPDATE INTERVAL                          15.0            (MINUTES)
 424 PRINT INTERVAL                           30.0            (MINUTES)
 425 DO WE OUTPUT INTEGRATION PARAMETERS?     YES
 426 DO WE GENERATE PHOTODISSOCIATION RATES?  YES
 427 SHOULD THEY VARY WITH ELEVATION?         YES
 428 DO WE HAVE SKY CLEARNESS RATIOS INPUT?   YES
 429 DO WE HAVE TEMPERATURES INPUT?           YES
 430 SHOULD WE PUNCH GROUND CONCENTRATIONS?   NO

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 431 MAXIMUM NUMBER OF INTEGRATION STEPS            1000
 432 NUMBER OF CHEMICAL REACTIONS                     51
 433 NUMBER OF SPECIE                                 30
 434 NUMBER OF SPECIE TREATED AS CONSTANTS             1
 435 NUMBER OF VERTICAL MESH POINTS                    5
 436 ELEVATION OF FIRST  VERTICAL MESH POINT 0.0          (METERS ABOVE SURFACE)
 437 ELEVATION OF SECOND VERTICAL MESH POINT 120.0
 438 ELEVATION OF THIRD  VERTICAL MESH POINT 300.0
 439 ELEVATION OF FOURTH VERTICAL MESH POINT 600.0
 440 ELEVATION OF FIFTH  VERTICAL MESH POINT 1200.0
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 441  SPECIES NAME, MOLE WT, AND BOUND COND    NO        30.
 442                                 2         NO2       46.
 443                                 3         O3        48.                 1
 444                                 4         HONO      47.
 445                                 5         CO        28.
 446                                 6         HCHO      30.
 447                                 7         RCHO      58.
 448                                 8         OLEF      42.
 449                                 9         PA        58.
 450                                10         NO3       62.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 451                                11         HO2       33.
 452                                12         PAN       121.
 453                                13         AR        92.
 454                                14         ARCO      156.
 455                                15         RO2       89.
 456                                16         RCO3      103.
 457                                17         AO2       74.
 458                                18         PAO2      89.
 459                                19         D         74.
 460                                20         HNO4      79.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 461                                21         ARO       123.
 462                                22         AROH      107.
 463                                23         AO        58.
 464                                24         OH        17.
 465                                25         O         16.
 466                                26         PAO       73.
 467                                27         N2O5      108.
 468  SO2 WITH DEPOSITION           28         SO2       64.                 1
 469                                29         SO4       96.                 1
 470                                30         H2O       18.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 471  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 1          1
 472  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 2          9
 473  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 3          8
 474  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 4         13
 475  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 5          6
 476  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 6          5
 477  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 7         28
 478  SPECIE INDEX FOR POINT SOURCE FLUX NO. 1          1
 479  SPECIE INDEX FOR POINT SOURCE FLUX NO. 2          9
 480  SPECIE INDEX FOR POINT SOURCE FLUX NO. 3          8
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 481  SPECIE INDEX FOR POINT SOURCE FLUX NO. 4        13
 482  SPECIE INDEX FOR POINT SOURCE FLUX NO. 5         6
 483  SPECIE INDEX FOR POINT SOURCE FLUX NO. 6         5
 484  SPECIE INDEX FOR POINT SOURCE FLUX NO. 7        28
 485  DEPOSITION SPECIE INDEX,VELOCITY,EXP.            3     .3      1.0
 486  DEPOSITION SPECIE INDEX,VELOCITY,EXP.           28     .48     1.0
 487  DEPOSITION SPECIE INDEX,VELOCITY,EXP.           29     .18     1.0
 488  NO      IC              3.0E-03   2.5E-03   1.5E-03   1.5E-03   1.5E-03
 489  NO2        COMPUTED
 490  O3      IC            .029      .043      .058      .058      .058

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 491  HONO       COMPUTED
 492  CO      IC            .133      .110      .066      .066      .066
 493  HCHO    IC              4.0E-04   3.4E-04   2.0E-04   2.0E-04   2.0E-04
 494  RCHO    IC              2.0E-04   1.7E-04   1.0E-04   1.0E-04   1.0E-04
 495  OLEF    IC              6.0E-04  5.10E-04   3.0E-04   3.0E-04   3.0E-04
 496  PA      IC              3.0E-03  2.54E-03   1.5E-03   1.5E-03   1.5E-03
 497  NO3        COMPUTED
 498  HO2     IC              1.0E-06   1.0E-06   1.0E-06   1.0E-06   1.0E-06
 499  PAN        COMPUTED
 500  AR      IC              6.0E-04  5.10E-04   3.0E-04   3.0E-04   3.0E-04

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 501  ARCO       COMPUTED
 502  RO2        COMPUTED
 503  RCO3       COMPUTED
 504  AO2        COMPUTED
 505  PAO2       COMPUTED
 506  O          COMPUTED
 507  HNO4       COMPUTED
 508  ARO        COMPUTED
 509  AROH       COMPUTED
 510  AO         COMPUTED

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 511  OH      IC               1.E-08    1.E-08    1.E-08    1.E-08    1.E-08
 512  O          COMPUTED
 513  PAO        COMPUTED
 514  N2O5       COMPUTED
 515  SO2                     3.2E-03   .28E-03   1.6E-03   1.6E-03   1.6E-03
 516  SO3                     3.2E-04   .28E-03   1.6E-03   1.6E-03   1.6E-03
 517  H2O     IC              1.7E 04   1.7E 04   1.7E 04   1.7E 04   1.7E 04
 518  RATE CONSTANT OF REACTION NO. 1                    COMPUTED FROM SOLAR
 519             REACTION NO.        2            4.12E 06  1/MIN
 520             REACTION NO.        3            2.50E 01
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 521              REACTION NO.        4          2.20E-09
 522              REACTION NO.        5          1.40E-03
 523              REACTION NO.        6                    COMPUTED FROM SOLAR
 524              REACTION NO.        7          1.50E 04
 525              REACTION NO.        8          1.50E 04
 526              REACTION NO.        9          4.40E 02
 527              REACTION NO.       10          1.20E 04
 528              REACTION NO.       11          1.70E 03
 529              REACTION NO.       12          8.40E 03
 530              REACTION NO.       13          5.00E-02

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 531              REACTION NO.       14          2.80E 04
 532              REACTION NO.       15          9.30E 02
 533              REACTION NO.       16          5.00E-05
 534              REACTION NO.       17          2.40E 01
 535              REACTION NO.       18          3.70E 04
 536              REACTION NO.       19          2.90E 04
 537              REACTION NO.       20          4.10E 05
 538              REACTION NO.       21                    COMPUTED FROM TEMPERATURE
 539              REACTION NO.       22          1.90E-02
 540              REACTION NO.       23          2.30E 04

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 541              REACTION NO.       24          5.20E 03
 542              REACTION NO.       25          2.90E 04
 543              REACTION NO.       26          3.80E 03
 544              REACTION NO.       27          2.90E 04
 545              REACTION NO.       28          2.60E 03
 546              REACTION NO.       29          1.40E 05   1/MIN
 547              REACTION NO.       30          2.90E 04
 548              REACTION NO.       31          6.70E 04   1/MIN
 549              REACTION NO.       32          2.30E 04
 550              REACTION NO.       33          2.20E 04

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 551              REACTION NO.       34                    COMPUTED FROM SOLAR
 552              REACTION NO.       35          2.90E 04
 553              REACTION NO.       36          1.70E 04
 554              REACTION NO.       37                    COMPUTED FROM TEMPERATURE
 555              REACTION NO.       38                    COMPUTED FROM SOLAR
 556              REACTION NO.       39          2.10E 04
 557              REACTION NO.       40          4.20E 03
 558              REACTION NO.       41          2.24E 04
 559              REACTION NO.       42          5.02E 04
 560              REACTION NO.       43          2.90E 04
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 561              REACTION NO.       44             5.60E 03
 562              REACTION NO.       45             2.90E 04
 563                                 46              8.4
 564                                 47             1.76E 03
 565                                 48              3.0E-02
 566                                 49              2.9E 03
 567                                 50              8.0
 568                                 51              8.0
 569    1.NO2   +    HV                =   1.O     +  1.NO
 570    1.O     +    O2    +    M      =   1.O3    +    M

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 571    1.O3    +  1.NO                =   1.NO2   +    O2
 572    1.NO    +  1.NO2   +  1.H2O    =   2.HONO
 573    2.HONO                         =   1.NO    +  1.NO2   +  1.H2O
 574    1.HONO  +    HV                =   1.OH    +  1.NO
 575    1.OH    +  1.NO    +    M      =   1.HONO  +    M
 576    1.OH    +  1.NO2   +    M      =   1.HNO3  +    M
 577    1.OH    +  1.CO                =   1.HO2   +    CO2
 578    1.HO2   +  1.NO                =   1.NO2   +  1.OH
 579    1.HO2   +  1.NO2               =   1.HNO4
 580    2.HO2                          =     H2O2  +    O2

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 581    1.NO2   +  1.O3                =   1.NO3   +    O2
 582    1.NO3   +  1.NO                =   2.NO2
 583    1.NO3   +  1.NO2               =   1.N2O5
 584    1.N2O5  +  1.H2O               =   2.HNO3
 585    1.N2O5                         =   1.NO3   +  1.NO2
 586    1.OH    +  1.OLEF              =   1.AO2
 587    1.AO2   +  1.NO                =   1.NO2   +  1.AO
 588    1.AO    +    O2                =   1.RCHO  +  1.HCHO  +  1.HO2
 589    1.HNO4                         =   1.HO2   +  1.NO2
 590    1.O3    +  1.OLEF              =   .5HCHO +.5RCHO +.25HO2 +.25RCO3 +.5OH +.5O

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 591    1.O     +  1.NO2               =   1.NO3   +  .5HCHO  +  .5RCHO
 592    1.O     +  1.OLEF              =   .3EPOX  +  .3RCHO  +  .4HO2   +  .4RO2
 593    1.O     +  1.NO                =   1.NO2   +  .5HCHO  +  .5RCHO
 594    1.OH    +  1.PA                =   1.H2O   +  1.PAO2
 595    1.PAO2  +  1.NO                =   1.NO2   + .85PAO   + .15RO2
 596    1.PAO2  +  1.NO                =     NTRA
 597    1.PAO                          =   1.RO2   +  .5HCHO  +  .5RCHO
 598    1.RO2   +  1.NO                =   1.NO2   +  1.PAO
 599    1.PAO   +    O2                =   .5KET   +  .5RCHO  +  1.HO2
 600    1.PAO   +  1.NO2               =  .85NTRA  + .15RCHO  + .15HONO
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 601    1.OH    +  1.RCHO             =    1.RCO3 +  1.H2O
 602    1.RCHO  +    HV               =    1.RO2  +  1.HO2  +  1.CO
 603    1.RCO3  +  1.NO               =    1.RO2    +  1.NO2  +    CO2
 604    1.RCO3  +  1.NO2              =    1.PAN
 605    1.PAN                         =    1.RCO3 +  1.NO2
 606    1.HCHO  +    HV               =   .67HO2  +  1.CO
 607    1.HCHO  +  1.OH               =    1.H2O  +  1.HO2  +  1.CO
 608    1.RO2   +  1.HO2              =      RO2H +    O2
 609    1.AR    +  1.OH               =    1.AROH +  1.HO2
 610    1.AROH  +  1.OH               =    1.ARCO

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 611    1.ARCO  +  1.NO               =    1.NO2  +  1.RCHO
 612    1.AR    +  1.OH               =    1.H2O  +  1.ARO
 613    1.ARO   +  1.NO               =    1.NO2  +  1.HO2  +    BALD
 614    1.SO2   +  1.O                =    1.SO4
 615    1.SO2   +  1.OH               =    1.SO4
 616    1.SO2   +  1.HO2              =    1.SO4  +  1.OH
 617    1.SO2   +  1.O                =    1.SO4  +  1.RCHO
 618    1.SO2   +  1.RO2              =    1.SO4  +  1.PAO
 619    1.SO2   +  1.AO2              =    1.SO4  +  1.AO
 620  LATITUDE                              38.6

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 621  LONGITUDE                             90.2
 622  TIME ZONE                             6.0
 623  DATE   (YRMODA)                          760629
 624  NUMBER OF TIME-VARYING REACTION RATES          2
 625  REACTION NO. OF 1ST RATE CONSTANT ARRAY        1
 626  REACTION NO. OF 2ND RATE CONSTANT ARRAY       38
 627  SKY CLEARNESS RATIOS (TIME & CLEAR)     360.       .549
 628                                          420.       .703
 629                                          480.       .758
 630                                          540.       .725

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 631                                          600.       .695
 632                                          660.       .545
 633                                          720.       .568
 634                                          750.       .784
 635                                          765.       .890
 636                                          780.       1.00
 637                                          810.       .86
 638                                          840.       .727
 639                                          870.       .85
 640                                          900.       .975
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 641                                          960.      1.00
 642  LAST                                    -10.
 643  SURFACE TEMPERATURES                      360.     21.50
 644                                            420.     22.41
 645                                            480.     24.39
 646                                            540.     26.46
 647                                            600.     27.67
 648                                            660.     26.46
 649                                            720.     28.51
 650                                            780.     29.61

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 651                                            840.     29.43
 652                                            900.     29.27
 653                                            960.     29.66
 654                                           1020.     29.64
 655  LAST DUMMY SURFACE TEMP (NEGATIVE)         - 10.
 656   DIFFUSIVITIES 1          360.0                 9.0       9.0      235.8
 657                            326.5     126.2     126.2
 658   DIFFUSIVITIES 2          420.0                41.1      41.1      165.8
 659                            140.1      26.6      26.6
 660   DIFFUSIVITIES 3          480.0               486.9     486.9      388.8

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 661                            841.9     507.3     507.3
 662   DIFFUSIVITIES 4          540.0              1477.0    1477.0      536.8
 663                            866.3     526.5     526.5
 664   DIFFUSIVITIES 5          600.0              1781.4    1781.4     6200.8
 665                           9842.7    1766.4    1766.4
 666   DIFFUSIVITIES 6          660.0               137.9     137.9      738.8
 667                           1495.3     363.1     363.1
 668   DIFFUSIVITIES 7          720.0              3099.0    3099.0    11193.6
 669                          18901.8    6755.4    6755.4
 670   DIFFUSIVITIES 8          780.0              2136.2    2136.2     7653.7

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 671                          13582.4   10763.3   10763.3
 672   DIFFUSIVITIES 9          840.0              1711.2    1711.2     7410.8
 673                          13552.5   10854.4   10854.4
 674   DIFFUSIVITIES10          900.0              1416.5    1416.5     5997.4
 675                          13092.3   10930.4   10930.4
 676   DIFFUSIVITIES11          960.0               103.1     103.1     5351.8
 677                           7507.8    2519.1    2519.1
 678   DIFFUSIVITIES12         1020.0               103.1     103.1     5352.0
 679                           7508.0    2519.2    2519.2
 680  LAST                      -10.0

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 681  LAST                      -10.0
 682  END
```

Figure 5-1 (Continued)

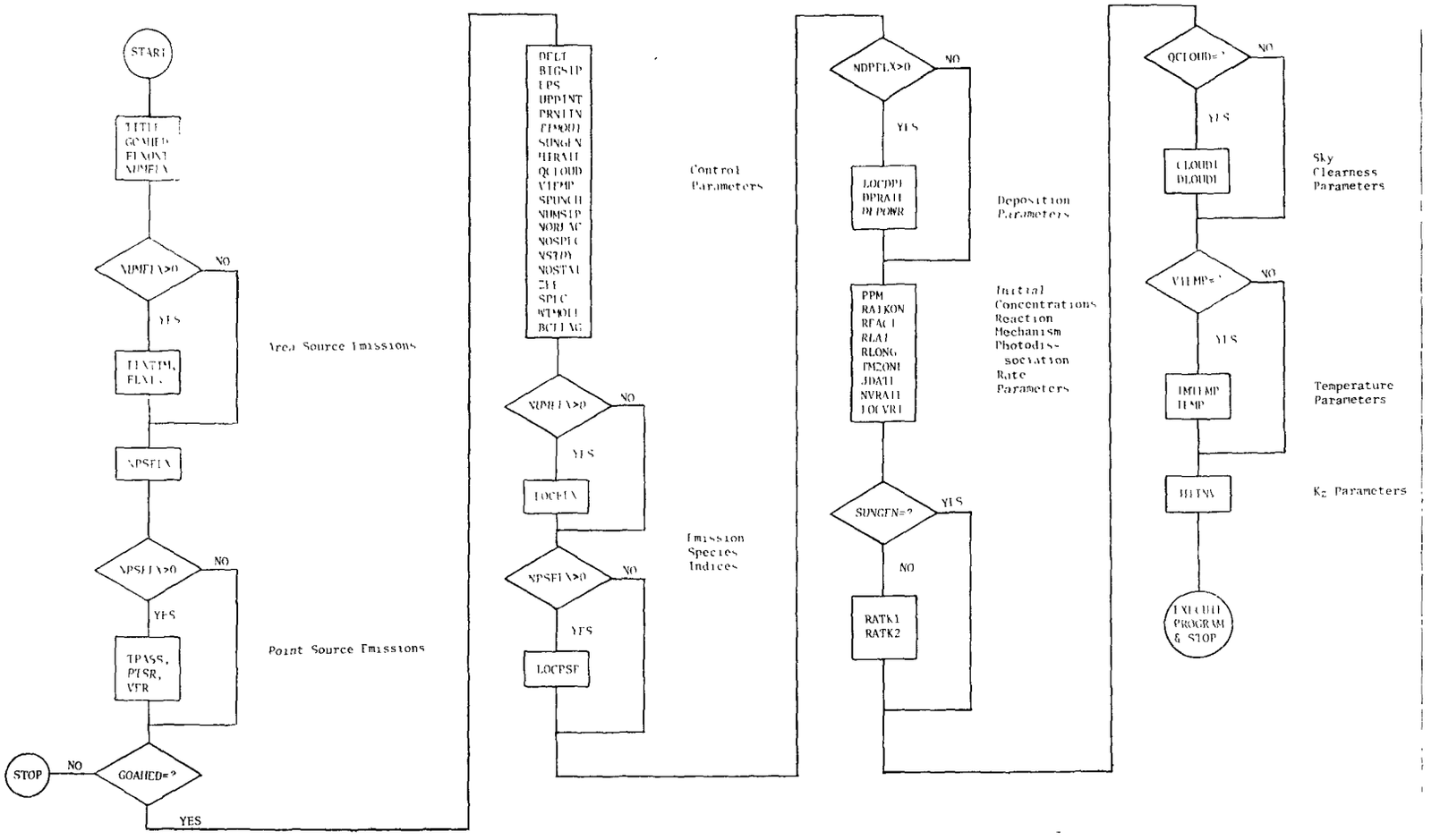


Figure 5-2 KEMOD Input Variable Sequence and Options

The program's inputs begin with the 80-character page label (TITLE) similar to that used in the other modules of the code. The page label is followed by a parameter (GOAHED), which indicates whether or not the program should proceed beyond the PREDAT listing of the input deck and reading of the emissions data schedule.

5.1.1 Source Emissions Data

The first card in the block of emissions data parameters specifies whether the emission data is input on the mass fraction or mole fraction basis. The user should specify either the word MOLE or MASS starting in column 41. When the emissions data are input on the mass fraction basis, the program converts it to a mole (volume) basis, using the molecular weights specified later in the input stream.

The next card read indicates the number of species with area source emissions data inputs. The user should specify a value of seven (7) corresponding to the species or species groups listed below:

1) Nitrogen oxides
2) Alkanes (paraffinic hydrocarbons)
3) Alkenes (olefinic hydrocarbons)
4) Aromatic hydrocarbons
5) Aldehyde hydrocarbons
6) Carbon monoxide
7) Sulfur oxides

Next, the area source emission fluxes are specified using a pair of cards for each entry in the schedule. Each pair contains a time (0-1440 minute clock) and seven species emission fluxes in the units of mole or mass fraction meters per minute. In the St. Louis application, the order of species' fluxes on the cards is indicated in the list shown above. In general, the order of the species' fluxes is arbitrary, subject to the constraint that nitrogen oxides (NO_x) and aldehydes must be first and fifth in the list. This constraint is necessary because the program internally partitions these emissions data to more than one species. More specifically, the program assumes the NO_x emissions are

90% nitric oxide (NO) and 10% nitrogen dioxide (NO_2). It assumes the aldehyde emissions are 60% formaldehyde (HCHO) and 40% other aldehyde species (RCHO). The list of area source emissions flux cards is terminated with a pair of cards, the first of which has a negative entry in columns 21-30. It is important to note that the program uses the initial and final times in the area source emissions schedule as the start time and stop time of the simulation.

Following the area source inputs, the program reads the point source emissions inputs. The number of species with point source emissions data (NPSFLX) is read first. The number of species here is normally seven, but may be less in situations where the user includes all hydrocarbon and/or carbon monoxide emissions sources as area sources. This card is followed by a temporal schedule, which normally (i.e., when NPSFLX>4) has three cards per point source. The first pair of cards indicates the time and mole or mass fraction-meters of species emissions entrained into the air parcel moving along the trajectory. (See Section 4 for an explanation of these units). The order in which the species emissions are listed on the cards is subject to the same constraints discussed above for area source emissions species. The third card of the set contains the normalized vertical distribution of the emissions with respect to the vertical mesh points. It is important to point out that the sum of the elements of this vertical distribution must equal one for proper representation of the source emissions. The point source data schedule is terminated by a pair of cards, the first of which has a negative entry in column 21-30.

5.1.2 Control Parameters and the Vertical Mesh

The program next reads a series of control parameters. The first card contains the initial integration time step size. This parameter is specified in minutes and must be quite small (10^{-3} to 10^{-5}) for proper start-up of the integration algorithm. Next, the program reads a card with the maximum integration time step size in minutes. The program's integration method is a variable step method, which chooses the time step at each point in the solution on the basis of certain convergence criteria. In practice, the maximum step size permitted is a necessary input to this algorithm, and values of 4 to 8 minutes are recommended.

Next, the user specifies the integration error control criteria. This is a dimensionless and somewhat empirical parameter, which should be input in the range of 10^{-3} to 10^{-4}. The program then reads a card which contains the update time interval in minutes. This is the interval at which the program checks for new values in the various schedules of time-varying input parameters. The update interval is used to revise the schedules of several inputs for use by the program, as described in Section 5.5. Values between 10 and 20 minutes are recommended.

The next card in the input deck specifies the time interval between printouts of the pollutant concentrations. This print interval must be chosen to be a multiple of the update interval for proper operation of the code.

The program next reads a series of cards with YES or NO inputs for various program options. The first card here indicates whether or not the program should print six integration progress parameters at each output interval. These parameters are discussed in Section 5.4. The next card indicates whether the user wants the program to generate the schedules of photodissociation rates (YES), or whether the user wants the data read in directly (NO). The following card also pertains to the photodissociation rates, and it indicates whether (YES), or not (NO), they vary with elevation. These last two cards should be input as YES under almost all conditions. See Section 5.3 for elucidation of the photodissociation rates used by the program. Next, the program reads an option flag, which indicates whether (YES), or not (NO), sky clearness ratios are included in the input deck. The program assumes clear sky conditions when they are absent. Continuing, the program reads a card which specifies whether (YES or NO) a schedule of temperatures is input. The program assumes an atmospheric temperature of $25^{o}C$ when this input is NO. The final YES or NO input indicates whether or not the user wants the computed ground concentrations punched at each print interval. This option is useful if the modeler wants to save the output in a form convenient for subsequent processing.

The program next reads a series of integer constants. The first is the maximum number of integration time steps permitted in a run. This number should be selected sufficiently large (i.e., $N \geq 75$ steps per hour of real-time simulation) so that termination of a run on the basis of this criterion occurs only in abnormal/erroneous conditions. Next,

the user specifies the number of chemical reactions (NOREAC) and number of chemical species (NOSPEC) in the chemical mechanism. For the ERT photochemical mechanism, these parameters have values of 51 and 30, respectively. In addition, the user must specify the number of species (a subset of NOSPEC) which are treated as constants or treated as steady-state species in the model. There is only one such species, H_2O, in the ERT photochemical mechanism. The last integer constant read in the section is the number of vertical mesh points (NOSTAT) input. This card is followed by NOSTAT cards with the mesh point elevations specified in order of ascending elevations, as done in the input decks for other program modules.

5.1.3 Chemical Species List

The program reads a series of cards which each contain the name (up to 4 characters) of one chemical species, its molecular weight, and a code for the type of surface boundary condition it has. The species' names must be input in the order shown in Figure 5-1 for the ERT photochemical mechanism. The molecular weight data is completely optional when emissions data are specified in moles. When the emissions data is input on the mass fraction basis, only the molecular weights of the species with emissions data need be specified. The boundary condition code is input with an integer value of one when the user wishes to include surface deposition for that species. Otherwise, the user may input a zero or leave the field blank to indicate the default (no deposition) surface boundary condition.

5.1.4 Emission Species and Surface Deposition Species

The user inputs a set of cards which indicates the correspondence between area source emissions species and the species number (index) in the chemical species list. The first card indicates the chemical species index for the first entry on the area source emissions data card pair. In the sample problem, the first entry of emissions data is for NO_x; so the corresponding species index is 1 for NO. Similarly, for the second card here, the species index is input as 9 (for PA) since the alkane emissions are specified second on the emissions data card pairs, etc. A similar list of species indices is input for the point source emissions

species, following the cards for the area source species indices.

Next, the program reads a set of cards with data for the species with surface deposition boundary conditions. The program reads one card for each species that has a boundary condition code equal to one. Each card contains one of the depositing species' index (from the species list), its surface deposition velocity (v) in meters/minute, and the exponent (p) of the concentration (c) in the analytic deposition flux (Φ_d) expression shown below:

$$\Phi_d = v*c^p$$

This exponent has a value of one for dry deposition, but may take on other values when modeling higher order surface chemical reactions.

5.1.5 Initial Concentrations

Initial pollutant concentrations are input using one card per species and with the species ordered as shown in the species list. Each card contains the initial concentrations at the vertical mesh points in the units of parts per million. Ground-level concentration is entered in columns 21-30; the concentration at the second mesh point is in columns 31-40; etc.

The reader can observe from Figure 5-1 that some cards are labeled COMPUTED and their concentration input fields are blank. Initial concentrations for these species are computed internally using the psuedo-steady-state assumption. Table 5-2 lists the species which require initial concentration input data and those which are computed. Note that one card for every species in the species list must be included, even though some cards have blank input fields.

5.1.6 Chemical Reaction Mechanism Inputs

The chemical reaction rate constants are specified next in the input deck. One rate constant per card is input in columns 41-50, and a total of NOREAC (See Section 5.1.2) cards must be input. The units of the rate constants are ppm^{-1} - $minute^{-1}$ for bimolecular reactions. The rate constants for ERT reaction numbers 1, 6, 21, 34, 37, and 38 may be left blank since they are either photodissociation rates or temperature

TABLE 5-2

USER SPECIFIED AND INTERNALLY COMPUTED INITIAL CONCENTRATIONS

User Specified	Internally Computed
NO	NO2
O3	HONO
CO	NO3
HCHO	PAN
RCHO	ARCO
OLEF	RO2
PA	RCO3
HO2	AO2
AR	PAO2
OH	D
SO2	HNO4
SO4	ARO
H2O	AROH
	AO
	O
	PAO
	N2O5

dependent rates which are determined elsewhere in the model.

The program next reads a block of cards (total number equals NOREAC) which describe the chemical mechanism. Each reaction is specified in alphanumeric format in columns 1-80. The information on these cards is used only to reproduce the mechanism in the output listing; it is not used in the computation. The stoichiometric coefficients for the ERT photochemical mechanism are contained in the program's explicit chemical rate expressions; thus, they are not a part of the input stream.

5.1.7 Photodissociation Rate Input Parameters

The program internally stores photodissociation rate data as functions of solar zenith angle and elevation above the surface. The user is required to specify the latitude, longitude, time zone, and date, so that the program can determine a temporal schedule of solar zenith angles and, thus, photodissociation rates. The longitude and latitude of the approximate center of the modeling region are first specified in degrees. Next, the time zone is indicated by the number of hours difference between Greenwich, England 0° meridian time and local standard time, followed by the date in the year-month-day form (YRMODA).

Next, the program reads the number and reaction number(s) of the independent photodissociation reactions. The program has built-in photodissociation rate data for NO_2 and HCHO photolysis, the two independent photodissociation rates. The program has been equipped to determine the photolysis rates for other species (HONO and RCHO) on the basis of the independent rates data. Thus, the user should specify this data exactly as shown in Figure 5-1 for the ERT photochemical mechanism.

The program has a seldom-used option which allows the user to input photodissociation rate constants. The user should specify SUNGEN = NO in the control parameter section of the input, and input two arrays for the photodissociation rates of NO_2 and HCHO. When the user wants to input different rates for the different mesh point elevations, HIRATE = YES is specified. The program then reads a block of cards which each has NOSTAT (See Section 5.1.2) entries of photodissociation rates for one reaction in the fields 21-32, 33-44, 45-56, 57-68, 69-80. The first card in the block must correspond to the start time of the simulation, and subsequent cards must be input for times at the frequency of the update interval. The last card in the block is input with a negative

entry in columns 21-32. A second block of cards is input for the photodissociation rates of the second reaction with the same format. If the user selects HIRATE = NO, only one entry per card is required in columns 21-32.

5.1.8 Sky Clearness, Temperature, and Diffusivity Schedules

The program reads a schedule of sky clearness ratios. Each card in the schedule contains a time in minutes and the ratio of actual ultraviolet radiation to the expected clear sky radiation for that time, date and location. The ratio may have values between zero and one, where one indicates clear sky conditions. These ratios are used by the program to adjust the photodissociation rates to represent the actual ultraviolet radiation intensity. The last card in the schedule is input with a negative entry in columns 41-50.

Next, the program reads a temporal schedule of temperatures. Each card contains the time in minutes and temperature in degrees Celcius. The temperatures specified here are used by the model to determine the rate constants of chemical reaction numbers 21 and 37. The last entry in the schedule contains a negative value in columns 41-50.

Lastly, the program reads the schedule of eddy diffusivity (K_z) coefficients. A pair of cards is input for each hour which contains a time in minutes and NOSTAT +1 K_z coefficients. The first card contains the time in columns 21-30 and three K_z coefficients in columns 41-50, 51-60, and 61-70. The second card contains the remaining K_z coefficients starting in column 21. The coefficients on the cards are ordered by ascending elevations. As described in Section 3, these K_z coefficients are the average K_z coefficients between vertical mesh point elevations. The first and last K_z coefficients on the card pair are assigned values equal to the next innermost K_z value for use by the program. The reading of this schedule is terminated when the program reads a pair of cards with the entry for time set less than zero.

Each of the schedules described here are output by the meteorological module in the proper format. Nevertheless, it is important for the user to know that the program requires each of these schedules to start at the same time as the simulation start time. The schedules should extend to a time when the input parameter becomes invariant. For example, the user may find it convenient to end the sky clearness ratio

schedule prior to the stop time if the sky is perfectly clear after a certain time of day. However, it is unlikely that this would be the case for temperature or K_z coefficients.

5.2 ERT Photochemical Mechanism

The model's chemical mechanism, as shown in Table 5-3, simulates the reactions occurring in an irradiated hydrocarbon/NO_x/SO_x mixture in air. The hydrocarbon reactivity is delineated by the use of five classes of hydrocarbon species in the mechanism. The primary purpose of the mechanism is to provide the chemical rates which control the formation of ozone, nitrogen dioxide, and sulfates in the atmosphere. This mechanism was developed by ERT and systematically tested against smog chamber data from the University of California, Riverside, chamber facility. The reader is referred to Table 5-4, which defines the chemical species symbols used in the mechanism.

The hydrocarbon reaction rate constants used in atmospheric application of the chemical mechanism are approximations based on the estimated reactivity of typical atmospheric hydrocarbon mixtures. The reactivity of the atmospheric alkanes is assumed to be somewhat greater than that of pure propane, but less than n-butane. The atmospheric alkenes are assumed to be as reactive as propene. The atmospheric aromatics have been assigned a reactivity greater than toluene but less than m-xylene. These rate constants may require adjustment for regions where the typical atmospheric hydrocarbon mixture is more or less reactive than St. Louis.

5.3 Photodissociation Rates

The ERT photochemical mechanism has four reactions which involve photolysis by ultraviolet radiation. These reactions play a particularly important role in the chemical processes which control ozone formation. In fact, without ultraviolet radiation, the rate of ozone formation in the troposhere would be negligible. They are also important as a source of the free radical species (OH, HO_2, and RO_2) that influence the HC/NO_x mechanism and the rate of sulfate formation.

The photlysis steps in the mechanism are for nitrogen dioxide, nitric acid, formaldehyde, and other aldehydes. The clear sky photodissociation rates for nitrogen dioxide and formaldehyde, as functions of

TABLE 5-3

THE ERT PHOTOCHEMICAL REACTION MECHANISM

		Rate Constants (ppm^{-1} min^{-1}) at 1 Atmosphere
1	$NO_2 + h\nu = O + NO$	Ultraviolet dependent
2	$O + O_2 (+ M) = O_3 (+ M)$	4.12E+06
3	$O_3 + NO = NO_2 + O_2$	2.50E+01
4	$NO + NO_2 + H_2O = 2HONO$	2.20E-09
5	$HONO + HONO = NO + NO_2 + H_2O$	1.40E-03
6	$HONO + h\nu = OH + NO$	Ultraviolet dependent
7	$OH + NO + M = HONO + M$	1.50E+04
8	$OH + NO_2 + M = HNO_3 + M$	1.50E+04
9	$OH + CO = HO_2 + CO_2$	4.40E+02
10	$HO_2 + NO = NO_2 + OH$	1.20E+04
11	$HO_2 + NO_2 = HNO_4$	1.70E+03
12	$HO_2 + HO_2 = H_2O_2 + O_2$	8.40E+03
13	$NO_2 + O_3 = NO_3 + O_2$	5.00E-02
14	$NO_3 + NO = 2NO_2$	2.80E+04
15	$NO_3 + NO_2 = N_2O_5$	9.30E+02
16	$N_2O_5 + H_2O = 2HNO_3$	5.00E-05
17	$N_2O_5 = NO_3 + NO_2$	2.40E+01
18	$OH + OLEF = AO_2$	3.70E+04
19	$AO_2 + NO = NO_2 + AO$	2.90E+04
20	$AO + O_2 = RCHO + HCHO + HO_2$	4.10E+05
21	$HNO_4 = HO_2 + NO_2$	Temperature dependent

TABLE 5-3 (CONTINUED)

		Rate Constants (ppm^{-1} min^{-1}) at 1 Atmosphere
22	O_3 + OLEF = 0.5HCHO + 0.5RCHO + 0.25HO_2 + 0.25RCO_3 + 0.5OH + 0.5$RCHO_2$	1.90E-02
23	$RCHO_2$ + NO_2 = NO_3 + 0.5HCHO + 0.5RCHO	2.30E+04
24	O + OLEF = .3EPOX + .3RCHO + .4HO_2 + .4RO_2	5.20E+03
25	$RCHO_2$ + NO = NO_2 + 0.5HCHO + 0.5RCHO	2.90E+04
26	OH + PA = H_2O + PAO_2	3.80E+03
27	PAO_2 + NO = NO_2 + .85PAO + .15RO_2	2.90E+04
28	PAO_2 + NO = NTRA	2.60E+03
29	PAO = RO_2 + .5HCHO + .5RCHO	1.40E+05
30	RO_2 + NO = NO_2 + PAO	2.90E+04
31	PAO + O_2 = .5KET + .5RCHO + HO_2	6.70E+04
32	PAO + NO_2 = .85NTRA + .15RCHO + .15HONO	2.30E+04
33	OH + RCHO = RCO_3 + H_2O	2.20E+04
34	RCHO + hν = RO_2 + HO_2 + CO	Ultraviolet dependent
35	RCO_3 + NO = CO_2 + NO_2 + RO_2	2.90E+04
36	RCO_3 + NO_2 = PAN	1.70E+04
37	PAN = RCO_3 + NO_2	Temperature dependent
38	HCHO + hν = .67HO_2 + CO	Ultraviolet dependent
39	HCHO + OH = H_2O + HO_2 + CO	2.10E+04
40	RO_2 + HO_2 = RO_2H + O_2	4.20E+03
41	AR + OH = AROH + HO_2	2.24E+04
42	AROH + OH = ARCO	5.02E+04

TABLE 5-3 (CONTINUED)

		Rate Constants (ppm^{-1} min^{-1}) at 1 Atmosphere
43	$ARCO + NO = NO_2 + RCHO$	2.90E+04
44	$AR + OH = H_2O + ARO$	5.60E+03
45	$ARO + NO = NO_2 + HO_2 + AR'CHO$	2.90E+04
46	$SO_2 + O = SO_4$	8.40E+00
47	$SO_2 + OH = SO_4$	1.76E+03
48	$SO_2 + HO_2 = SO_4 + OH$	3.00E-02
49	$SO_2 + D = SO_4 + RCHO$	2.90E+03
50	$SO_2 + RO_2 = SO_4 + PAO$	8.00E+00
51	$SO_2 + AO_2 = SO_4 + AO$	8.00E+00

TABLE 5-4

CHEMICAL SPECIES SYMBOL DEFINITIONS

Species	Symbol Designation
AO	Alkoxy radical equivalent to AO_2
AO_2	Product of OH addition to olefin in the presence of O_2
AR	Aromatic hydrocarbons
AR'CHO	Aromatic aldehyde
ARCO	Product of addition of OH to a cresol in the presence of O_2
ARO	Product of H-abstraction from side chain alkyl group on benzene ring followed by addition of O_2 to radical formed
AROH	Cresols
CO	Carbon monoxide
CO_2*	Carbon dioxide
D	Criegee intermediate ($RCHO_2$)
EPOX*	Epoxide formed from O atom addition to olefin
HCHO	Formaldehyde
HONO	Nitrous acid
HNO_3*	Nitric acid
HNO_4	Pernitric acid
HO_2	Hydroperoxyl radical
H_2O	Water
H_2O_2*	Hydrogen peroxide
HV	Photon
KET*	Ketones
M*	Any third body, such as N_2 or O_2
NO	Nitric oxide

TABLE 5-4 (CONTINUED)

Species	Symbol Designation
NO_2	Nitrogen dioxide
NO_3	Nitrate radical
N_2O_5	Dinitrogen pentoxide
NTRA*	Organic nitrate
O	Oxygen atom (ground state)
O_2*	Oxygen
O_3	Ozone
OH	Hydroxyl radical
OLEF	Alkenes (olefinic hydrocarbons)
PA	Alkanes (paraffinic hydrocarbons)
PAN	Peroxyacetyl Nitrate
PAO	Alkoxy radical formed by PA
PAO_2	Alkyl peroxy radical from the O_2 addition to the radical formed by H-abstraction from a paraffinic hydrocarbon
R**	Generalized alkyl group (e.g., C_2H_5, C_3H_7, etc.)
RCHO	Aldehydes other than formaldehyde
RCO_3	Acyl peroxy radical
RO	Alkoxyl radical
RO_2	Alkyl peroxy radical
RO_2H*	Product of disproportionation between HO_2 and RO_2
SO_2	Sulfur dioxide
SO_4	Sulfate

* Species' Concentration not computed by the model

**Definition of species group

elevation and solar zenith angle, are presented in Tables 5-5 and 5-6. The rates shown are 95% of the values reported by Peterson, 1977 and Peterson, et al 1976.

5.4 Description of the Output

This section describes the printed output of the KEMOD program. The output of the program includes the PREDAT listing of the input deck, partial verification of the input data, the computed species concentrations, and a printer-plot of the concentration history for five species. There is also an optional listing to verify the schedules of all time-varying parameters used by the program.

The first page of output following the PREDAT listing (see Figure 5-1) is a partial verification of the input control parameters. As shown in Figure 5-3, the program lists the initial and final simulation times, the initial and maximum integration time step sizes, and the number of reactions, species, steady-state species, and vertical mesh points.

The next page of output, Figure 5-4, lists area source emissions data for the seven emission species. The columns labeled NO and HCHO represent the total nitrogen oxides and aldehydes emissions fluxes, respectively. The user should note that the data has been rescheduled such that each entry is at a fixed time interval, the update interval described in Section 5.5. Also, the fluxes have been converted to the units of ppm-meters per minute.

Next, the program lists the sums of the effective point source emission rates by time, species, and vertical mesh points (or cells) as shown in Figure 5-5. For each time shown, the program lists NOSTAT lines of emission rates in ppm per minute. The first line refers to the emissions in the ground cell by species, the second line corresponds to the second cell, etc. The program converts the input data to the emission rates shown, as described in Section 5.5.

The next page of output is a listing of the ERT photochemical mechanism and the rate constants used for each of the reactions (See Figure 5-6). Note that the listing includes the initial photodissociation and temperature-dependent reaction rates computed internally. This page is followed by one with the chemical species list, as shown in Figure 5-7.

TABLE 5-5

NO_2 Photodissociation Rates (per minute)

$$NO_2 + h\nu \xrightarrow{k} NO + O$$

Elevation Above Surface (meters)	Solar Zenith Angle									
	0°	10°	20°	30°	40°	50°	60°	70°	78°	86°
0	.550	.545	.532	.508	.471	.416	.334	.219	.108	.024
150.	.583	.578	.566	.543	.507	.453	.371	.251	.126	.026
360.	.613	.608	.597	.574	.540	.486	.405	.280	.145	.028
640.	.641	.637	.626	.605	.571	.518	.436	.309	.165	.032
980.	.668	.665	.653	.634	.599	.547	.466	.335	.184	.035
1380.	.693	.689	.679	.659	.626	.574	.491	.359	.201	.038
1840.	.714	.711	.691	.681	.649	.597	.514	.380	.218	.041
2350.	.733	.730	.720	.701	.669	.617	.536	.400	.234	.044
2910.	.751	.748	.738	.720	.689	.639	.554	.421	.251	.047
3530.	.768	.765	.755	.737	.708	.659	.578	.443	.269	.052
4210.	.783	.780	.771	.754	.726	.677	.599	.465	.288	.057

TABLE 5-6

HCHO Photodissociation Rates* (per minute)

$$HCHO + h\nu \xrightarrow{k} \begin{cases} .67HO_2 + .33CO \\ .67CO + HCO \end{cases}$$

Solar Zenith Angles

Elevations Above Surface (meters)	0°	10°	20°	30°	40°	50°	60°	70°	78°	86°
0.	.621	.613	.584	.536	.467	.376	.266	.144	.060	.011
150.	.671	.662	.632	.583	.511	.414	.298	.164	.068	.013
360.	.719	.710	.680	.629	.555	.455	.329	.183	.077	.014
640.	.768	.758	.728	.676	.599	.494	.361	.204	.086	.015
980.	.817	.807	.776	.721	.642	.533	.393	.225	.095	.016
1380.	.858	.849	.816	.761	.679	.566	.422	.244	.104	.018
1840.	.907	.897	.864	.807	.723	.605	.454	.266	.114	.019
2350.	.943	.933	.899	.843	.757	.638	.482	.286	.124	.020
2910.	.983	.972	.939	.882	.795	.672	.512	.308	.136	.022
3530.	.983	.972	.939	.882	.795	.672	.512	.308	.136	.022
4210.	.983	.972	.939	.882	.795	.672	.512	.308	.136	.022

*All values shown have the exponent 10^{-2}.

```
:RT CHEMICAL-DIFFUSION MODULE SIMULATION  --  ST. LOUIS - 6/29/76  (181)

:NPUT DATA FOR VERTICAL DIFFUSION WITH PHOTOCHEMISTRY

     INITIAL TIME IS                        360.00 MINUTES
     FINAL TIME IS                         1020.00 MINUTES
     INITIAL TIME STEP SIZE IS             .000100 MINUTES
     MAXIMUM TIME STEP SIZE IS               6.00 MINUTES
     NUMBER OF REACTIONS IS                    51
     NUMBER OF SPECIES IS                      30
     NUMBER OF SPECIES IN STEADY STATE IS       1
     NUMBER OF VERTICAL MESH POINTS IS          5    INCLUDING THE GROUND AND THE EDGE
```

Figure 5-3

:RT CHEMICAL-DIFFUSION MODULE SIMULATION -- ST. LOUIS - 6/29/76 (181)

REVISED AREA SOURCE FLUX SCHEDULE --- PPM-METERS/MINUTE

TIME	NO	PA	OLEF	AR	HCHO	CO	SO2
360.0	0.	0.	0.	0.	0.	0.	0.
375.0	0.	0.	0.	0.	0.	0.	0.
390.0	.1301E-03	.9809E-05	.6396E-05	.4467E-05	.3148E-05	.8256E-03	.7077E-0
405.0	.6295E-03	.4746E-04	.3095E-04	.2161E-04	.1523E-04	.3995E-02	.3424E-0
420.0	.8311E-03	.1606E-03	.1088E-03	.5272E-04	.3435E-04	.9470E-02	.4057E-0
435.0	.1399E-02	.3837E-03	.2840E-03	.1358E-03	.8470E-04	.2371E-01	.6006E-0
450.0	.1399E-02	.3837E-03	.2840E-03	.1358E-03	.8470E-04	.2371E-01	.6006E-04
465.0	.1399E-02	.3837E-03	.2840E-03	.1358E-03	.8470E-04	.2371E-01	.6006E-04
480.0	.1327E-02	.3756E-03	.2687E-03	.1267E-03	.7899E-04	.2190E-01	.5760E-04
495.0	.9742E-03	.3807E-03	.2011E-03	.7500E-04	.4506E-04	.1196E-01	.4470E-04
510.0	.6356E-03	.1597E-03	.7141E-04	.3208E-04	.1847E-04	.4525E-02	.3293E-04
525.0	.5345E-03	.1116E-03	.5377E-04	.2721E-04	.1632E-04	.4176E-02	.2836E-04
540.0	.4313E-03	.1224E-03	.5266E-04	.2639E-04	.1359E-04	.3346E-02	.2246E-04
555.0	.7403E-03	.4262E-03	.1980E-03	.7363E-04	.4702E-04	.6062E-02	.4079E-04
570.0	.4319E-03	.1211E-03	.3691E-04	.5692E-04	.1741E-04	.3386E-02	.1836E-04
585.0	.1992E-02	.2333E-02	.1009E-02	.3830E-03	.1634E-03	.4185E-01	.5007E-04
600.0	.5200E-02	.5881E-02	.3727E-02	.1725E-02	.1127E-02	.1974E+00	.1858E-03
615.0	.8387E-02	.5010E-02	.3380E-02	.1445E-02	.8860E-03	.2507E+00	.2681E-03
630.0	.1780E-01	.9854E-02	.7877E-02	.3173E-02	.2302E-02	.7371E+00	.4582E-03
645.0	.3793E-01	.2061E-01	.1473E-01	.6640E-02	.4405E-02	.1046E+01	.1108E-02
660.0	.6737E-01	.3404E-01	.2401E-01	.1034E-01	.6565E-02	.2048E+01	.3245E-02
675.0	.9343E-01	.3515E-01	.2449E-01	.1184E-01	.7056E-02	.2141E+01	.4570E-02
690.0	.8570E-01	.3253E-01	.2184E-01	.1040E-01	.6053E-02	.1574E+01	.2466E-02
705.0	.3136E+00	.7795E-01	.5353E-01	.2441E-01	.1572E-01	.4309E+01	.1204E-01
720.0	.5842E-01	.1451E-01	.9854E-02	.4694E-02	.3881E-02	.6278E+00	.3545E-02
735.0	.7291E-01	.4343E-01	.2419E-01	.8197E-02	.5368E-02	.1700E+01	.2901E-02
750.0	.4734E-01	.1602E-01	.1128E-01	.4181E-02	.3656E-02	.8330E+00	.3077E-02
765.0	.2359E-01	.8004E-02	.5766E-02	.2070E-02	.1558E-02	.4178E+00	.6708E-03
780.0	.1287E-01	.4408E-02	.3420E-02	.1306E-02	.1033E-02	.2903E+00	.4639E-03
795.0	.1289E-01	.5863E-02	.4263E-02	.1570E-02	.1181E-02	.3565E+00	.4793E-03
810.0	.8390E-02	.2317E-02	.1624E-02	.6246E-03	.4658E-03	.1167E+00	.3038E-03
825.0	.1098E-01	.4351E-02	.2582E-02	.7891E-03	.5514E-03	.1375E+00	.3245E-03
840.0	.1042E-01	.2187E-02	.1651E-02	.6628E-03	.4860E-03	.1162E+00	.3310E-03
855.0	.6665E-02	.1545E-02	.1098E-02	.4268E-03	.3111E-03	.7907E-01	.2540E-03
870.0	.1841E-02	.4134E-03	.3057E-03	.1238E-03	.9088E-04	.2357E-01	.7492E-04
885.0	0.	0.	0.	0.	0.	0.	0.
900.0	0.	0.	0.	0.	0.	0.	0.
915.0	0.	0.	0.	0.	0.	0.	0.
930.0	0.	0.	0.	0.	0.	0.	0.
945.0	0.	0.	0.	0.	0.	0.	0.
960.0	0.	0.	0.	0.	0.	0.	0.
975.0	0.	0.	0.	0.	0.	0.	0.
990.0	0.	0.	0.	0.	0.	0.	0.
1005.0	0.	0.	0.	0.	0.	0.	0.
1020.0	0.	0.	0.	0.	0.	0.	0.

Figure 5-4

-RT CHEMICAL-DIFFUSION MODULE SIMULATION -- ST. LOUIS - 6/29/76 (181)

POINT SOURCE EMISSION RATES BY VERTICAL CELL (PPM/MIN)

TIME	NO	PA	OLEF	AR	HCHO	CO	SO2
750.0	.140E-03	.149E-04	.519E-05	.868E-04	.201E-06	.166E-02	.545E-03
	.745E-04	.119E-04	.411E-05	.618E-04	.893E-07	.125E-02	.219E-03
	.925E-04	.831E-05	.294E-05	.433E-04	.130E-06	.864E-03	.195E-03
	.249E-04	.257E-05	.836E-06	.135E-04	.350E-07	.264E-03	.543E-04
	0.	0.	0.	0.	0.	0.	0.
765.0	.116E-04	.292E-06	.143E-07	.134E-06	.970E-08	.779E-06	.321E-04
	.475E-05	.231E-06	.764E-08	.107E-06	.398E-08	.331E-06	.178E-04
	.729E-05	.161E-06	.862E-08	.739E-07	.612E-08	.488E-06	.196E-04
	.186E-05	.580E-07	.249E-08	.267E-07	.158E-08	.125E-06	.461E-05
	0.	0.	0.	0.	0.	0.	0.
780.0	.857E-05	.345E-06	.318E-07	.133E-06	.718E-08	.418E-06	.197E-04
	.338E-05	.273E-06	.223E-07	.106E-06	.284E-08	.165E-06	.784E-05
	.544E-05	.189E-06	.179E-07	.734E-07	.456E-08	.265E-06	.125E-04
	.137E-05	.666E-07	.539E-08	.265E-07	.117E-08	.667E-07	.326E-05
	0.	0.	0.	0.	0.	0.	0.
795.0	.169E-06	.553E-07	.206E-07	.156E-09	.468E-09	.186E-07	.115E-06
	.127E-06	.435E-07	.162E-07	.117E-09	.350E-09	.139E-07	.863E-07
	.855E-07	.299E-07	.111E-07	.787E-10	.236E-09	.936E-08	.582E-07
	.256E-07	.910E-08	.338E-08	.236E-10	.707E-10	.280E-08	.174E-07
	0.	0.	0.	0.	0.	0.	0.
810.0	.169E-06	.468E-09	.468E-09	.156E-09	.468E-09	.186E-07	.115E-06
	.127E-06	.350E-09	.350E-09	.117E-09	.350E-09	.139E-07	.863E-07
	.855E-07	.236E-09	.236E-09	.787E-10	.236E-09	.936E-08	.582E-07
	.256E-07	.707E-10	.707E-10	.236E-10	.707E-10	.280E-08	.174E-07
	0.	0.	0.	0.	0.	0.	0.
825.0	.773E-06	.813E-09	.813E-09	.271E-09	.813E-09	.363E-07	.762E-07
	.598E-06	.629E-09	.629E-09	.210E-09	.629E-09	.281E-07	.590E-07
	.396E-06	.416E-09	.416E-09	.139E-09	.416E-09	.186E-07	.390E-07
	.104E-06	.109E-09	.109E-09	.363E-10	.109E-09	.487E-08	.102E-07
	0.	0.	0.	0.	0.	0.	0.
840.0	.773E-06	.813E-09	.813E-09	.271E-09	.813E-09	.363E-07	.762E-07
	.598E-06	.629E-09	.629E-09	.210E-09	.629E-09	.281E-07	.590E-07
	.396E-06	.416E-09	.416E-09	.139E-09	.416E-09	.186E-07	.390E-07
	.104E-06	.109E-09	.109E-09	.363E-10	.109E-09	.487E-08	.102E-07
	0.	0.	0.	0.	0.	0.	0.
855.0	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.

Figure 5-5

The molecular weights and boundary condition codes are listed with each of the species numbers and names.

The next page of output, Figure 5-8, describes the vertical mesh geometry and the initial diffusion coefficients. Each vertical mesh point (or station in the listing) has its elevation above ground listed in meters. The initial eddy diffusivity coefficients are listed in a manner which shows that there is one K_z coefficient associated with the first station, one K_z coefficient between each pair of adjacent stations, and one K_z coefficient associated with the top (or edge) station. The vertical distance between the mesh points is printed under the column labeled DELTA Z. The figures in the last column indicate the cell height, i.e., the height of the volume of air associated with each cell.

The subsequent page of output lists the initial concentrations for all of the species at all the mesh point elevations, as shown in Figure 5-9. The list includes the concentrations of the species with internally computed initial concentrations.

The program then begins to output a series of pages at the frequency of the print interval which are similar to Figure 5-10. Each page contains the time in minutes (0-1440) and in hours (0-2400). A summary of several parameters for the surface cell is included. The values of the surface cell parameters shown are the last ones used before the times shown (updating of these and other parameters occurs after the printout is completed). The surface parameters listed are the NO_2 photodissociation rate, the eddy diffusivity coefficient, and the area and point source emission rates in ppm/minute. Note that the list includes emission rates for nine species, which is the result of the program allocating the NO_x emissions to NO and NO_2, and the aldehyde emissions to HCHO and RCHO. Next, the printout lists the species' names and the corresponding computed pollutant concentrations in parts per million at the mesh points heights. These, of course, are the most important results of the model.

Figure 5-10 also shows the optional output of integration progress parameters for the case with TIMOUT = YES. The listing includes the integration indexx which is normally equal to two. If the EPISODE algorithm encounters erroneous/abnormal conditions which prevent successful integration, the integration index is returned with a negative value and an error message describing the error. The listing shows the last

	REACTIONS	RATE CONSTANTS (1/(PPM--MIN))
1.	1.NO2 + HV = 1.O + 1.NO	.7041E-01
2.	1.O + O2 + M = 1.O3 + M	.4120E+07
3.	1.O3 + 1.NO = 1.NO2 + O2	.2500E+02
4.	1.NO + 1.NO2 + 1.H2O = 2.HONO	.2200E-08
5.	2.HONO = 1.NO + 1.NO2 + 1.H2O	.1400E-02
6.	1.HONO + HV = 1.OH + 1.NO	.1971E-01
7.	1.OH + 1.NO + M = 1.HONO + M	.1500E+05
8.	1.OH + 1.NO2 + M = 1.HNO3 + M	.1500E+05
9.	1.OH + 1.CO = 1.HO2 + CO2	.4400E+03
10.	1.HO2 + 1.NO = 1.NO2 + 1.OH	.1200E+05
11.	1.HO2 + 1.NO2 = 1.HNO4	.1700E+04
12.	2.HO2 = H2O2 + O2	.8400E+04
13.	1.NO2 + 1.O3 = 1.NO3 + O2	.5000E-01
14.	1.NO3 + 1.NO = 2.NO2	.2800E+05
15.	1.NO3 + 1.NO2 = 1.N2O5	.9300E+03
16.	1.N2O5 + 1.H2O = 2.HNO3	.5000E-04
17.	1.N2O5 = 1.NO3 + 1.NO2	.2400E+02
18.	1.OH + 1.OLEF = 1.AO2	.3700E+05
19.	1.AO2 + 1.NO = 1.NO2 + 1.AO	.2900E+05
20.	1.AO + O2 = 1.RCHO + 1.HCHO + 1.HO2	.4100E+06
21.	1.HNO4 = 1.HO2 + 1.NO2	.3404E+01
22.	1.O3 + 1.OLEF = .5HCHO +.5RCHO +.25HO2 +.25RCO3 +.50H +.50	.1900E-01
23.	1.O + 1.NO2 = 1.NO3 + .5HCHO + .5RCHO	.2300E+05
24.	1.O + 1.OLEF = .3EPOX + .3RCHO + .4HO2 + .4RO2	.5200E+04
25.	1.O + 1.NO = 1.NO2 + .5HCHO + .5RCHO	.2900E+05
26.	1.OH + 1.PA = 1.H2O + 1.PAO2	.3800E+04
27.	1.PAO2 + 1.NO = 1.NO2 + .85PAO + .15RO2	.2900E+05
28.	1.PAO2 + 1.NO = NTRA	.2600E+04
29.	1.PAO = 1.RO2 + .5HCHO + .5RCHO	.1400E+06
30.	1.RO2 + 1.NO = 1.NO2 + 1.PAO	.2900E+05
31.	1.PAO + O2 = .5KET + .5RCHO + 1.HO2	.6700E+05
32.	1.PAO + 1.NO2 = .85NTRA + .15RCHO + .15HONO	.2300E+05
33.	1.OH + 1.RCHO = 1.RCO3 + 1.H2O	.2200E+05
34.	1.RCHO + HV = 1.RO2 + 1.HO2 + 1.CO	.1377E-03
35.	1.RCO3 + 1.NO = 1.RO2 + 1.NO2 + CO2	.2900E+05
36.	1.RCO3 + 1.NO2 = 1.PAN	.1700E+05
37.	1.PAN = 1.RCO3 + 1.NO2	.1259E-02
38.	1.HCHO + HV = .67HO2 + 1.CO	.4132E-03
39.	1.HCHO + 1.OH = 1.H2O + 1.HO2 + 1.CO	.2100E+05
40.	1.RO2 + 1.HO2 = RO2H + O2	.4200E+04
41.	1.AR + 1.OH = 1.AROH + 1.HO2	.2240E+05
42.	1.AROH + 1.OH = 1.ARCO	.5020E+05
43.	1.ARCO + 1.NO = 1.NO2 + 1.RCHO	.2900E+05
44.	1.AR + 1.OH = 1.H2O + 1.ARO	.5600E+04
45.	1.ARO + 1.NO = 1.NO2 + 1.HO2 + BALD	.2900E+05
46.	1.SO2 + 1.O = 1.SO4	.8400E+01
47.	1.SO2 + 1.OH = 1.SO4	.1760E+04
48.	1.SO2 + 1.HO2 = 1.SO4 + 1.OH	.3000E-01
49.	1.SO2 + 1.O = 1.SO4 + 1.RCHO	.2900E+04
50.	1.SO2 + 1.RO2 = 1.SO4 + 1.PAO	.8000E+01
51.	1.SO2 + 1.AO2 = 1.SO4 + 1.AO	.8000E+01

Figure 5-6

ERT CHEMICAL-DIFFUSION MODULE SIMULATION -- ST. LOUIS - 6/29/76 (181)

	SPECIES	MOLECULAR WEIGHT	BOUNDARY CONDITION
1.	NO	30.0	0
2.	NO2	46.0	0
3.	O3	48.0	1
4.	HONO	47.0	0
5.	CO	28.0	0
6.	HCHO	30.0	0
7.	RCHO	58.0	0
8.	OLEF	42.0	0
9.	PA	58.0	0
10.	NO3	62.0	0
11.	HO2	33.0	0
12.	PAN	121.0	0
13.	AR	92.0	0
14.	ARCO	156.0	0
15.	RO2	89.0	0
16.	RCO3	103.0	0
17.	AO2	74.0	0
18.	PAO2	89.0	0
19.	D	74.0	0
20.	HNO4	79.0	0
21.	ARO	123.0	0
22.	AROH	107.0	0
23.	AO	58.0	0
24.	OH	17.0	0
25.	O	16.0	0
26.	PAO	73.0	0
27.	N2O5	108.0	0
28.	SO2	64.0	1
29.	SO4	96.0	1
30.	H2O	18.0	0

Figure 5-7

ERT CHEMICAL-DIFFUSION MODULE SIMULATION -- ST. LOUIS - 6/29/76 (181) CP TIME = 0.00 DATE

INITIAL DIFFUSION COEFFICIENTS

VALUES SHOWN ARE FOR POINTS HALFWAY BETWEEN STATIONS

STATION	HEIGHT (METERS)	DIFFUSIVITY	DELTA Z	CELL HEIGHT
1	0.00	9.0000000		60.00
		9.0000000	120.00	
2	120.00			150.00
		235.80000	180.00	
3	300.00			240.00
		326.50000	300.00	
4	600.00			450.00
		126.20000	600.00	
5	1200.00	126.20000		300.00

Figure 5-8

TIME = 360.00 MINUTES (600 ON 2400-CLOCK)

INITIAL CONCENTRATION PROFILES IN PARTS PER MILLION

HEIGHT-M.	NO PA AO2 O	NO2 NO3 PAO2 PAO	O3 HO2 O N2O5	HONO PAN HNO4 SO2	CO AR ARO SO4	HCHO ARCO AROH H2O	RCHO RO2 AO	OLEF RCO3 OH
0.00	.30000E-02 .30000E-02 .25517E-08 .41869E-14	.24501E-06 .43568E-11 .12025E-08 .38625E-11	.29000E-01 .10000E-05 .18999E-08 .39950E-16	.22827E-04 .48164E-08 .12237E-09 .32000E-02	.13300E+00 .60000E-03 .38621E-09 .32000E-03	.40000E-03 .15448E-08 .26773E-03 .17000E+05	.20000E-03 .81679E-08 .54146E-12	.60000E-03 .14557E-08 .10000E-07
120.00	.25000E-02 .25400E-02 .26028E-08 .65767E-14	.34134E-06 .10806E-10 .12218E-08 .38217E-11	.43000E-01 .10000E-05 .28733E-08 .13804E-15	.16872E-04 .90004E-08 .17048E-09 .28000E-03	.11000E+00 .51000E-03 .39393E-09 .28000E-03	.34000E-03 .15757E-08 .22757E-03 .17000E+05	.17000E-03 .98732E-08 .46024E-12	.51000E-03 .19527E-08 .10000E-07
300.00	.15000E-02 .15000E-02 .25517E-08 .68131E-14	.31254E-06 .22231E-10 .12025E-08 .25960E-11	.58000E-01 .10000E-05 .37994E-08 .26003E-15	.89473E-05 .10153E-07 .15610E-09 .16000E-02	.66000E-01 .30000E-03 .38621E-09 .16000E-02	.20000E-03 .15448E-08 .13386E-03 .17000E+05	.10000E-03 .11331E-07 .27073E-12	.30000E-03 .24057E-08 .10000E-07
600.00	.15000E-02 .15000E-02 .25517E-08 .89950E-14	.35912E-06 .25544E-10 .12025E-08 .26304E-11	.58000E-01 .10000E-05 .37993E-08 .34330E-15	.77869E-05 .11666E-07 .17936E-09 .16000E-02	.66000E-01 .30000E-03 .38621E-09 .16000E-02	.20000E-03 .15448E-08 .13386E-03 .17000E+05	.10000E-03 .11495E-07 .27073E-12	.30000E-03 .24057E-08 .10000E-07
1200.00	.15000E-02 .15000E-02 .25517E-08 .12486E-13	.42311E-06 .30095E-10 .12025E-08 .26825E-11	.58000E-01 .10000E-05 .37992E-08 .47655E-15	.66092E-05 .13745E-07 .21133E-09 .16000E-02	.66000E-01 .30000E-03 .38621E-09 .16000E-02	.20000E-03 .15448E-08 .13386E-03 .17000E+05	.10000E-03 .11743E-07 .27073E-12	.30000E-03 .24057E-08 .10000E-07

Figure 5-9

ERT CHEMICAL-DIFFUSION MODULE SIMULATION -- ST. LOUIS - 6/29/76 (181)

TIME = 780.00 MINUTES (1300 ON 2400-CLOCK)

SURFACE CELL PARAMETERS:

NO2 PHOTODISSOCIATION RATE = .476 (/MIN)
DIFFUSIVITY COEFFICIENT = 2376.9 (M**2/MIN)

AREA SOURCES (PPM/MIN)		POINT SOURCES (PPM/MIN)	
NO	.193E-03	NO	.771E-05
PA	.735E-04	PA	.345E-06
OLEF	.570E-04	OLEF	.318E-07
AR	.218E-04	AR	.133E-06
HCHO	.103E-04	HCHO	.431E-08
CO	.484E-02	CO	.418E-06
SO2	.773E-05	SO2	.197E-04
NO2	.215E-04	NO2	.857E-06
RCHO	.689E-05	RCHO	.287E-08

CONCENTRATION PROFILES IN PARTS PER MILLION

HEIGHT-M.	NO PA AO2 O	NO2 NO3 PAO2 PAO	O3 HO2 O N2O5	HONO PAN HNO4 SO2	CO AR ARO SO4	HCHO ARCO AROH H2O	RCHO RO2 AO	OLEF RCO3 OH
0.00	.43947E-02	.10463E-01	.45896E-01	.29387E-04	.32687E+00	.12767E-02	.17996E-02	.26958E-02
	.65820E-02	.20050E-06	.66450E-06	.19683E-03	.21745E-02	.64978E-08	.13951E-06	.12369E-07
	.48761E-07	.11227E-07	.30302E-08	.15272E-05	.59559E-08	.26463E-03	.15164E-10	.62325E-07
	.12094E-08	.91666E-10	.78474E-07	.68282E-02	.13768E-02	.17000E+05		
120.00	.41444E-02	.98543E-02	.47092E-01	.29259E-04	.30995E+00	.12397E-02	.17761E-02	.24878E-02
	.62954E-02	.20530E-06	.72493E-06	.19740E-03	.20767E-02	.74607E-08	.15457E-06	.13623E-07
	.51714E-07	.12339E-07	.30356E-08	.15679E-05	.65366E-08	.26444E-03	.15166E-10	.67551E-07
	.11969E-08	.95733E-10	.75670E-07	.68289E-02	.13890E-02	.17000E+05		
300.00	.41026E-02	.94554E-02	.47706E-01	.29435E-04	.30202E+00	.12224E-02	.17664E-02	.23887E-02
	.61491E-02	.20156E-06	.76946E-06	.19803E-03	.20157E-02	.81218E-08	.16583E-06	.14653E-07
	.54113E-07	.13135E-07	.30410E-08	.15966E-05	.69142E-08	.26415E-03	.15710E-10	.72878E-07
	.12064E-08	.10162E-09	.71313E-07	.67400E-02	.13937E-02	.17000E+05		
600.00	.40700E-02	.90171E-02	.48303E-01	.29780E-04	.29425E+00	.12057E-02	.17575E-02	.22923E-02
	.59972E-02	.19617E-06	.82536E-06	.19895E-03	.19407E-02	.89387E-08	.17990E-06	.15996E-07
	.57312E-07	.14138E-07	.30545E-08	.16325E-05	.73468E-08	.26342E-03	.16506E-10	.79795E-07
	.12164E-08	.10932E-09	.66205E-07	.65348E-02	.13974E-02	.17000E+05		
1200.00	.38435E-02	.81287E-02	.49390E-01	.30119E-04	.27652E+00	.11663E-02	.17328E-02	.20827E-02
	.56302E-02	.19145E-06	.93964E-06	.20147E-03	.17225E-02	.10755E-07	.20941E-06	.19029E-07
	.63882E-07	.16283E-07	.30993E-08	.16753E-05	.79995E-08	.25837E-03	.17374E-10	.92446E-07
	.11767E-08	.12010E-09	.58249E-07	.58212E-02	.14026E-02	.17000E+05		

INTEGRATION INDEX = 2 LAST STEP SIZE = .141E+01 LAST ORDER USED = 2

NUMBER OF STEPS = 220 NUMBER OF FUNCTION EVALUATIONS = 336 NUMBER OF JACOBIAN EVALUATIONS = 127

Figure 5-10

integration step size (in minutes) and the last order of the approximation used. The order refers to the number of previous time step concentrations and derivatives used to predict the concentrations at that time. The value of the order varies between one and five, depending on how rapidly the solution is varying. The listing shows the number of time steps, the number of function evaluations, and the number of Jacobian evaluations. Since the method used is a predictor-corrector method, the integration algorithm should always take more function evaluations than time steps. Similarly, the program is designed to minimize the number of Jacobian evaluations; thus, the number of Jacobians should always be less than the number of steps taken.

The last printed page of output from the chemical-diffusion model is a printer-plot of surface concentrations in ppm versus time. Figure 5-11 shows a plot of the concentration histories for the sample problem. The legend indicates the plotting symbols for the five species included in the plot. The nitric oxide, nitrogen dioxide, ozone, sulfur dioxide, and one-tenth carbon monoxide concentrations are plotted on the same page. Note that when two or more species have approximately the same concentrations at a plotting time, only the symbol associated with the last species in the legend with that value appears on the plot.

The user may select a seldom-used option of the KEMOD program which provides verification of the times and values the code has used in updating its parameters from the various temporal schedules. When this option is used, the following data is outputted.

- Photodissociation rates for all mesh point elevations
- Area source emission fluxes
- Point source emission rates
- Temperatures
- Sky clearness ratios
- K_z coefficients

The photodissociation rates, emission rates, and K_z coefficients are printed at the frequency of the update interval. The temperatures and sky clearness ratios are printed at the frequencies with which their values are changed. This results in a very sizable printout. This data

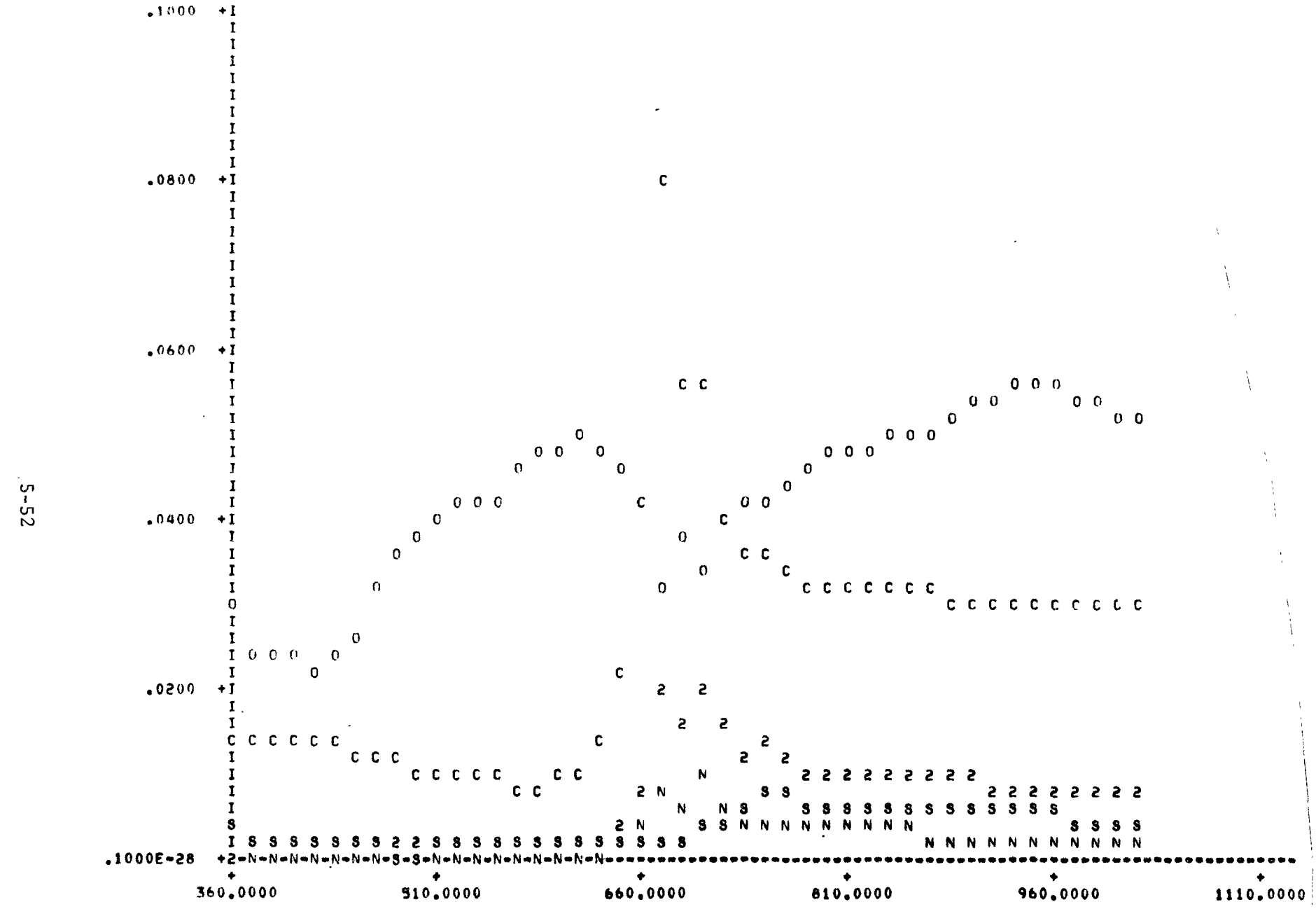

.1000
.0800
.0600
.0400
.0200
.1000E-28
360.0000
510.0000
660.0000
810.0000
960.0000
1110.0000
TIME (MINUTES)
LEGEND: N = NO 2 = NO2 O = O3 S = SO2 C = .1*CO
ERT CHEMICAL-DIFFUSION MODULE SIMULATION -- ST. LOUIS - 6/29/76 (181)

Figure 5-11

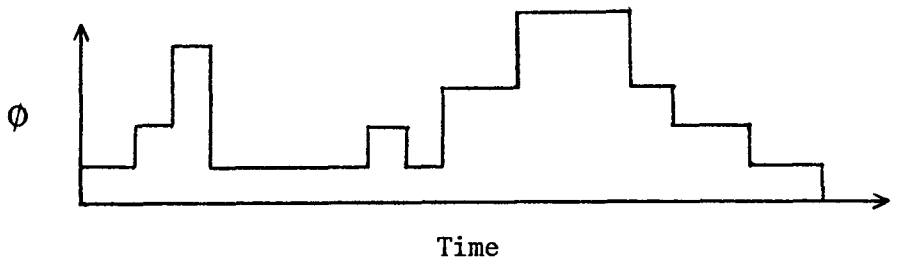


Figure 5-12 (a) Original Schedule

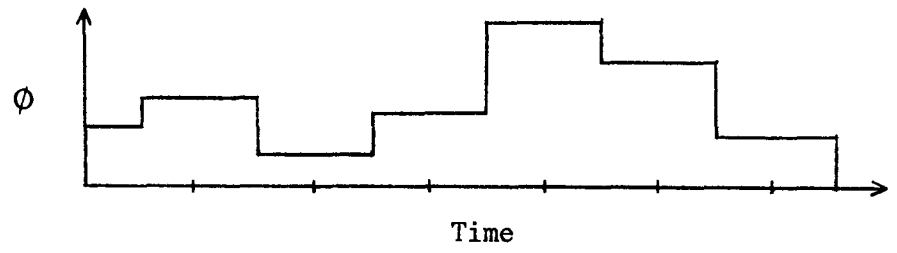


Figure 5-12 (b) Revised Schedule

is printed on logical unit numbers 10 and 11. Thus, to obtain the printout, the user must supply the appropriate installation-dependent job control such that these files are disposed to the line printer following the normal execution of the program.

5.5 Computational Procedures

Once the program has read the input data as described in Section 5.1, it performs various initialization tasks prior to passing control to the integration algorithm where the concentrations are computed. One common feature of most of the initialization tasks involves creating or revising the temporal schedules of parameters so that the entries in the schedules occur at a fixed time interval. In the subsequent discussion, we will refer to the program's 'update schedule', which is a schedule with a fixed time interval equal to the user-specified update interval.

The first initialization task involves generating the schedules of NO_2 and HCHO photodissociation rates for the duration of the run. The schedules are created by subroutine PHOTOD on the program's update schedule with entries for each mesh point elevation.

Next, the program updates the sky clearness ratio and determines the initial photodissociation rates in subroutines UPRAT2 and RATEHI. Also, the initial temperature-dependent rate constants are determined in subroutine TEMPR.

The program then proceeds to transform the area source emissions input data to the program's update schedule. The revision of the schedule involves an Euler integration of the emission rates to find the mean rate at each time in the new schedule. Figure 5-12 illustrates a typical input schedule and revised schedule of emission rates for one species. This transformation, of course, conserves the mass of emissions injected into the air parcel. Following the rescheduling (performed in subroutine SKEDUL), the program converts the area source emission fluxes to the units of ppm-meters/minute.

The next initialization task involves determining the point source emission rates from the input data. The program first truncates the time at which the source is passed by the air parcel to the nearest time in the program's update schedule. Second, it converts the mass or mole fraction-meters of emissions entrained into the air parcel to a rate

assuming an entrainment/mixing period of twice the update interval. Third, it allocates the emission rates to the appropriate vertical mesh points, using the vertical distribution input for each source. The final result of this task is a schedule of the sums of point source emission rates by species and mesh point on the program's update schedule.

The next initialization task involves smoothing the K_z coefficient schedule. The program linearly interpolates the hourly K_z coefficients to determine intermediate values which coincide with the times in the program's update schedule. This task is performed to represent the mixing process as one which gradually changes with time, as opposed to one which may have order-of-magnitude changes in coefficients on the hour.

The last significant initialization task is the computation of the initial concentrations of species not specified in the input stream. The calculation is performed in subroutine ISTATE, which includes explicit expressions for the steady-state concentrations applitable to the ERT photochemical mechanism.

At this point, the program passes control to the integration algorithm which is controlled by subroutine DRIVE. DRIVE and its 16 subprograms perform the integration of the finite-differenced, ordinary differential equations where the chemistry and diffusion are coupled for each time step.

The integration method is one developed by Hindmarsh and Byrne (1975) named EPISODE (Experimental Package for the Integration of Systems or Ordinary Differential Equations). The method is a variable-step, variable-order, implicit backwards differencing scheme that is particularly useful for solving stiff systems of equations. The model's ordinary differential equations are quite stiff, and the algorithm is well-suited for this application. The EPISODE package has been modified to take into account the block tri-diagonal form of the model's differential equations, which greatly improves its efficiency.

Program control is returned from DRIVE to subroutine KEMOD2 at every time in the program's update schedule. KEMOD2 checks certain parameters returned by DRIVE to ensure that the integration is proceeding satisfactorily. It prints the concentrations when appropriate. It stores the concentrations for subsequent plotting, and punches the ground concentrations if this option has been selected. Then, it updates the emissions data, sky clearness ratios, photodissociation rates,

temperature-dependent reaction rates, and K_z coefficients to the current time. If the current time is less than the stop time, program control returns to DRIVE where the integration is performed for another update interval. This cycle of integration, output, and updating is repeated until the end of the simulation. The final task for the program is the generation of printer-plots of ground concentration histories for the more important species.

5.6 Program Modifications

The KEMOD program is quite complicated; only the most probable modifications will be described here. These modifications relate to changing the chemical mechanism, changing the partitioning of emissions data, increasing the number of emission species, and changing the species' concentrations plotted.

5.6.1 Changing the Chemical Mechanism

The program is currently designed to include up to 40 chemical species, 10 of which must be either specified as steady-state approximations, or treated as constants. Thus, up to 30 species can be integrated. These dimensions are felt to be sufficiently large for any practical application of the model.

The program is presently equipped to handle up to 55 chemical reactions. The number of chemical reactions can easily be changed by changing the dimension, N, in the arrays shown below:

LOCATION	ARRAY NAME
COMMON/CHEM2/	RATKON(N), RATEFF(N)
COMMON/CHEM5/	REACT(20,N)
SUBROUTINE RATES	R(N)
SUBROUTINE ISTATE	R(N)

See Appendix A for the list of subroutines which contain common blocks CHEM2 and CHEM5.

To install a new mechanism or change the present mechanism, the user must supply a set of explicit chemical rate equations in subroutine

RATES. There should be one rate equation for each species that is integrated by the program. The indices of the rate equation must correspond to the indices of the species in the species list. The method of determining the components of the rate equations is shown below for one chemical reaction. The expressions, of course, become quite long when a species is involved in many reactions, but the method is the same.

Reaction: $aC_i + bC_j \overset{k_m}{\rightarrow} dC_k + eC_l$

Let

C_n = C(N) = concentration of nth species

k_m = R(M) = rate constant of mth reaction

a,b = A,B = reactant stoichiometric coefficients of mth reaction

d,e = D,E = product stoichiometric coefficients of mth reaction

Rate Equations:

```
RATE(I) = -R(M)* (C(I)**A) * (C(J)**B)

RATE(J) = -R(M)* (C(I)**A) * (C(J)**B)

RATE(K) = +R(M) * D * (C(I) ** A) * (C(J) ** B)

RATE(L) = +R(M) * E * (C(I) ** A) * (C(J) ** B)
```

NOTE: The reactant stoichiometric coefficients are included only when they are not equal to one.

Once the rate equations are determined for the mechanism, the user must supply the matrix of partial derivatives (i.e., the Jacobian) of the rate equations. The elements of the Jacobian are specified in subroutine JACOB as described below:

A(N,N) = an N by N array with the elements of the Jacobian, where N equals the number of species being integrated.

Let: RATE(I) = f(R,C) = f(R(1),...,R(J),...R(NOREAC), C(1),...,C(K),...,C(NOSPEC))

Where: RATE(I) = rate equation of Ith species

R(J) = rate constant of Jth reaction

C(K) = concentration of Kth species

Then: $A(I,K) = \frac{\partial f(R,C)}{\partial C(K)}$

Where: I = 1,...,N

K = 1,...,N

Next, the user must determine if changes need to be made concerning the time-varying reaction rate constants embedded in the program. If the new or revised chemical mechanism has more than two photodissociation rates, the user must explicitly scale the extra photodissociation rates in subroutine RATEHI. For a photodissociation rate of reaction J, which is A times the photodissociation rate of reaction I1, and where I1 is the reaction number of the first independent photodissociation rate input, the user should insert the following two statements in RATEHI:

RATKON(J) = A RAKON(I1)

RATEFF(J) = RATKON(J)

The user must, of course, remove the statements in RATEHI which presently determine the rates of reactions numbers 6 and 34 from the photodissociation rates I1 and I2, respectively, if they are not appropriate to the new mechanism.

In addition, if the user wants to include temperature dependent reaction rates, subroutine TEMPR must be modified. The user must replace the current expressions for the temperature dependence of rate constant numbers 21 and 37, with similar expressions for the appropriate reaction numbers. The expression should be formulated on the basis of temperature (T) in degrees Kelvin. If none of the rates are temperature dependent, subroutine TEMPR should be converted to a dummy subroutine.

Lastly, the user should supply a new version of subroutine ISTATE that contains explicit algebraic equations for the steady-state initial concentrations of selected species. To determine the steady-state

expression for the Ith species, let

```
RATE (I)  =  f(R,C)  =  0
```

and solve for the Ith concentration as a function (g) of the rate constants (R) and the other known concentrations (C').

```
C(I)  =  g(R,C')
```

One may also choose to read in all of the initial conditions and convert ISTATE into a dummy subroutine.

5.6.2 Changing the Partitioning of Emissions Input Data

As explained in Section 5.1.1, the program presently partitions the nitrogen oxides (NO_x) and aldehyde (ALDE) emissions data to more than one species, respectively. The manner in which this is performed is as follows. The program stores the NO_x emissions as NO (species No. 1) and the aldehyde emissions as HCHO (species No. 6). At each update interval, subroutines UPFLX1 and UPSORC are called to retrieve the emissions data for that time. These subroutines perform the partitioning of the emissions data for KEMOD. Let us use as an example the present partitioning of aldehyde emissions as 40% RCHO (species No. 7) and 60% HCHO. Subroutine UPFLX1 would then include the statements:

```
FLXW1(7)  =  .40*FLXW1(6)
FLXW2(7)  =  .40*FLXW2(6)
FLXW1(6)  =  .60*FLXW1(6)
FLXW2(6)  =  .60*FLXW2(6)
```

Subroutine UPSORC would include the statements:

```
      DO 30  J     =  1,NOSTAT
      PSR1(7,J)  =  .40 PSR1(6,J)
      PSR2(7,J)  =  .40 PSR2(6,J)
      PSR1(6,J)  =  .60 PSR1(6,J)
      PSR2(6,J)  =  .60 PSR2(6,J)
   30 CONTINUE
```

If the user wishes to change the partitioning percentages, he/she merely changes the corresponding coefficients shown above.

5.6.3 Increasing the Number of Species With Emissions Inputs

The program's arrays are presently dimensioned for seven emissions species and can easily be modified for up to 17 species. To change the arrays to handle emissions for N species ($7 < N \leq 17$), the following changes should be made:

LOCATION	ARRAY NAME or Statement
COMMON/CHEM5/*	LOCFLX(N)
COMMON/FLUXES/*	FLXIN(N,200)
COMMON/PS1/*	PS(N,5,75),LOCPSF(N)
PROGRAM KEMOD	PTSR(N,200),AF(N)
SUBROUTINE KEMOD2	PTSR(N,200)
SUBROUTINE KEMOD2	CALL XMIT(-N*375,0.0,PS)

Various input/output formats in KEMOD, KEMOD2, UPFLX1, and UPSORC also require modification when the number of emissions species exceeds seven.

5.6.4 Changing the Plot Species

The plotting software is designed to provide a plot of surface concentrations of up to five species. The user may change the number of species plotted (NPTS), the plotting symbols (array KSYM), and the plot species indices (array KPS) by inserting the appropriate statements in subroutine KEMOD2. As an example, suppose the user wishes to plot the 10th and 20th species in the species list with symbols A and B, respectively. The user would insert the statements:

```
DATA KSYM/1HA,1HB/
KPS(1)  =  10
KPS(2)  =  20
NPTS  =  2
```

*See appendix A for the list of subroutines with this common block.

and remove the present statements pertaining to these variables. The user will also wish to modify format statement number 1001 in subroutine COPLOT to define the new legend.

6. REFERENCES

ASME 1968. Recommended Guide for the Prediction of Dispersion of Airborne Effluents. M. Smith (ed), New York, NY.

Baulch, D.L., D.D. Drysdale, and D.G. Horne 1973. "Rate Constants for Reactions of the H_2-N_2-O_2 System" in Chemical Kinetic Data Survey V. Sixty-six Contributed Rate and Photochemical Data Evaluations on Ninety-four Reactions, NBSIR 73-206. D. Garvin, ed., National Bureau of Standards, Washington, DC, pg 49-115.

Briggs, G.A. 1969. Plume Rise. AEC Critical Review Series.

Briggs, G.A. 1975. Plume Rise Predictions. Lectures on Air Pollution and Environmental Impact Analysis, American Meteorological Socity, Boston, MA, September.

Demerjian, K.L., K.L. Shere, and J.T. Peterson 1979. "Theoretical Estimates of Actinic (Spherically Integrated) Flux and Photolytic Rate Constants of Atmospheric Species in the Lower Troposphere." Advances in Environmental Science and Technology, J.N. Pitts (ed); Vol 9 (in press).

Hindmarsh, A.C. and G.D. Byrne 1975. Experimental Package for Integration of Systems of Ordinary Differential Equations (EPISODE). Lawrence Livermore Laboratory, UCID-20112.

Lloyd, A.C., F.W. Lurman, D.G. Godden, J.F. Hutchins, A.Q. Eschenroeder, R.A. Nordsieck 1979. Development of the ELSTAR Photochemical Air Quality Simulation Model and Its Evaluation Relative to the LARPP Data Base, ERT Report 5287 prepared for Coordinating Research Council, New York (in press).

Moortgat, G.EK. F. Slemr, W. Seiler, and P. Warneck 1978. Chem. Physics LeH., in press.

Peterson, J.T. 1977. Dependence of NO_2 Photodissociation Rate Constant on Altitude, Atmospheric Environment, February.

Peterson, J.T., K.L. Demerjian, and K.L. Shere 1976. Actinic Solar Flux and Photolytic Rate in the Troposphere, International Conference on Photochemical Oxidant Pollution and its Control.

Turner, D.B. 1969. Workbook of Atmospheric Dispersion Estimates. U.S. Department of Health, Education & Welfare. Public Health Service Publication No. 999, AP-26.

APPENDIX A

COMMON BLOCK LOCATIONS IN THE CODE

1. Meteorological Module Common Blocks

Common Block	Location
AIRQAL	GETCON, METIN, METMOD, WINDY
CELECT	EDDY, STABIL
CNTRL	CROSIT, METMOD, SETTUP, WINDY
CORIOL	EDDY, STABIL
CROSS	CROSIT
DATES	GETAZV, METIN, METMOD, WINDRD, WINDY,
DIFDAT	EDDY, STADIF, KEKLAY
GRID	CROSIT, EDGE, METMOD, PLACIT, SETIN, SETTUP, WINDY
IMAG	SETPLT
INPUTS	CROSIT, EDDY, GETCON, METIN, METMOD, SETTUP, WINDY
KZINPT	EDDY, KZDATA, METIN, METMOD, WINDY
ORIGIN	BARIER, CROSIT, EDDY, GETAZV, METMOD, RUFNES, SETIN, SETTUP, WINDY
PEVE	EDDY, RUFNES, UXYPOS
REUSE	CROSIT, GETCON, METIN, METMOD, WINDY
TEMPHT	EDDY, KEKLAY, SMOOTH, STADIF
TRAJ	CROSIT, EDDY, METMOD, PLACIT, SETTUP, UXYPOS, WINDY
VGRID	CROSIT, BLOCK DATA
WDATA	ANGTST, AZVDIS, GETAZV, GETCON, METIN METMOD, SETIN, SETTUP, WINDY, WSECLS
WHERE	EDDY, METIN, METMOD, SETIN, SKY, WINDY
WIND	CROSIT, EDDY, METMOD, PLACIT, SETTUP, UXYPOS, WINDY
WINFLD	GETAZV, GETCON, METIN, METMOD, WINDY

2. Emissions Module Common Blocks

Common Block	Locations
ANGLES	LOCATE, PONTEM, SEGMET
ANSWER	AREAEM, EMMOD, GRIDIT, PLUMAS, PONTEM, SEGMET, STACKS
DEGRID	AREAEM, GRIDIT
EMRATE	AREAEM, EMMOD
HISTRY	ADHOUR, AREAEM, EMMOD
INPUT	AREAEM, EMMOD, PONTEM, STACKS
LABLIN	AREAEM, EMMOD, PONTEM, STACKS
LABOUT	AREAEM, EMMOD, PONTEM, STACKS
LINKS	DISTAN, LOCATE, PONTEM, SEGMET
MXHITE	EMMOD, PONTEM
ORIGIN	EMMOD, PONTEM, STACKS
PARCEL	AREAEM, EMMOD, PLUMAS, PONTEM, STACKS
PASSES	PONTEM
PSDAT	PONTEM, STACKS
SIGMAS	PARTIT, PLUMAS, PONTEM, SEGMET, BLOCK DATA
SIGTAL	PARTIT, PLUMAS, BLOCK DATA
TABLES	ADHOUR, AREAEM, EMMOD, PONTEM
TEMPS	EMMOD, PONTEM
TRAJ	EMMOD, LOCATE, PONTEM
WIND	DISTAN, EMMOD, LOCATE, PONTEM, SEGMET
WORKER	ADHOUR, AREAEM, GRIDIT, PONTEM, STACKS

3. Chemical-Diffusion Module Common Blocks

Common Block	Locations
CHEM1	DIFFUN, DRIVE, ISTATE, KEMOD2, PEDERV, PSET, RATES, STEADY, TSTEP
CHEM2	DIFFUN, DRIVE, ISTATE, KEMOD2, PEDERV, PSET, RATEHI, RATES, STEADY, TEMPR, TSTEP
CHEM3	DIFCOF, DIFFUN, DRIVE, KEMOD2, PEDERV, PSET, TSTEP, UPFLX1
CHEM4	KEMOD2, PHOTOD, UPRAT2
CHEM5	DIFFUN, KEMOD2, UPFLX1
EPCOM1	ADJUST, COSET, DRIVE, INTERP, PSET, TSTEP
EPCOM2	DRIVE, PSET, TSTEP
EPCOM3	DRIVE, PSET, TSTEP
EPCOM4	DRIVE, PSET, TSTEP
EPCOM5	DRIVE, PSET, TSTEP
EPCOM6	DRIVE, KEMOD, KEMOD2, PSET, TSTEP
EPCOM7	DRIVE, PSET, TSTEP
EPCOM8	DRIVE, PSET, TSTEP
EPCOM9	DRIVE, PSET, TSTEP
EPCM10	ADJUST, COSET, TSTEP
EPCM12	DRIVE, PSET, TSTEP
FLUXES	KEMOD, KEMOD2, UPFLX1
INPUTS	KEMOD, KEMOD2, PHOTOD
PRPLOT	COPLOT, KEMOD2
PS1	DIFFUN, KEMOD, KEMOD2, UPSORC
PS2	DIFFUN, KEMOD2, UPSORC

APPENDIX B
CODE SUBROUTINE DESCRIPTIONS

1. Meteorological Module Subroutines and Their Use

PROGRAM METMOD

Use. METMOD is the main program for the meteorological module of the code. It reads several key control parameters that determine which submodules are to be called. These parameters specify whether the wind trajectory is to be generated from wind data or is specified in the input data-cards, whether or not air quality data is to be interpolated along the trajectory, and whether or not diffusivity coefficients (K_z) are to be computed. In addition, METMOD controls the multiple case feature of the program.

SUBROUTINE ANGTST (XS,YS,NGOOD,KUSE,OK)

Use. Compares the azimuth from (XS,YS) to the new station against the corresponding azimuths for any previously qualified stations. If any two azimuths are less than 2 degrees apart, ANGTST returns a positive flag; otherwise, it returns a negative flag. ANGTST is called by GETAZV.

Arguments.

XS	=	x-coordinate of current position
YS	=	y-coordinate of current position
NGOOD	=	number of measurement stations qualified thus far
KUSE	=	index of new candidate measurement station
OK	=	output flag. If positive, angle test is OK (no other station within 2 degrees); if negative, test not OK

SUBROUTINE AZVDIS (NP,AZV,DDD,AZM,VEL,IRF)

Use. Given raw azimuth, velocity, and distance data from up to three measurement stations, AZVDIS combines them, according to the prescribed interpolation rule, to produce a single azimuth and velocity. AZVDIS is called by WINDY.

Arguments. NP = number of measurements to combine (up to three)
AZV = array containing raw azimuths and velocities
DDD = array with distances from current position to closest stations
AZM = calculated distance
VEL = calculated velocity
IRF = interpolation flag; if 1, inverse distance weighting is used; if 2, inverse distance squared weighting is used

SUBROUTINE BARIER (X,Y,XS,YS,OK)

Use. Checks whether or not two given points are on the same side of each of a set of (up to 20) finite-length, straight-line barriers. The barrier line segments are stored internally, being defined by the coordinates of their end points. BARIER is called by GETAZV and WINDY.

Arguments. X = x-coordinate of point of interest
Y = y-coordinate of point of interest
XS = x-coordinate of reference point
YS = y-coordinate of reference point
OK = output flag; if positive, both points are on the same side of all barriers; otherwise, the points are on opposite sides of at least one barrier

SUBROUTINE CELAVG (DH,NMESHP,CELVAT,AKZ,NZHI,STORE)

Use. This routine computes average K_z coefficients for use in the chemical-diffusion module from the detailed internal K_z profile. CELAVG is called by EDDY.

Arguments. DH = vertical increment for internal K_z profile
NMESHP = number of vertical mesh points (up to five)
CELVAT = elevations of vertical mesh points
AKZ = array of internal K_z coefficients
NZHI = number of K_z coefficients in AKZ array
STORE = array of average K_z coefficients returned

SUBROUTINE CROSIT

Use. CROSIT generates trajectory grid square crossing schedules and plots the wind trajectory. CROSIT is called by WINDY.

Arguments. None

SUBROUTINE DATE (IOLD,MOVE,IWAY,NEW)

Use. Calculates a new time and date from an old time and date and a prescribed time increment. Time is on a 0-2400 basis and dates are represented as year, month, day (e.g., 780901 for Sept. 1, 1978). DATE is called by WINDY.

Arguments. IOLD = 2-word array with time and date to be changed
MOVE = time increment in minutes
IWAY = if positive, time advances; otherwise, time is decreased
NEW = 2-word output array with new time and date

SUBROUTINE DIFFUS (STAB,ZHT,AKZ,USTAR)

Use. This routine determines the K_z coefficient at a specific elevation within the surface layer. DIFFUS is called by STADIF.

Arguments. STAB = Monin-Obukhov length
ZHT = elevation above ground
AKZ = value of K_z returned
USTAR = friction velocity

FUNCTION DOTTY (A,B)

Use. DOTTY computes the dot product of the vectors A and B. DOTTY is called by ANGTST.

Arguments. A = a 3-component vector
B = a 3-component vector

FUNCTION DTIME (A)

Use. This function converts time from the 0-2400 clock to hours and decimal hours. DTIME is called by EDDY.

Argument. A = time on a 0-2400 hour clock

SUBROUTINE EDDY (KPWIND)

Use. This routine is the driver for the eddy diffusivity submodule. EDDY determines a schedule of atmospheric stability classes for the emissions module and a schedule of diffusion (K_z) coefficients for the chemical-diffusion module. EDDY is called by WINDY.

Argument. KPWIND = Hollerith variable equal to YES when punched output is desired

SUBROUTINE EDGE (X,Y,ITF,IEDGE)

Use. EDGE checks whether or not the point X,Y is outside the modeling region. EDGE is called by WINDY.

Arguments.

X = x-coordinate of point of interest

Y = y-coordinate of point of interest

ITF = direction flag; if positive, the trajectory is proceeding forward; if negative, the trajectory is proceeding backwards

IEDGE = output flag; if zero, the point is inside the modeling region; if non-zero, the point is outside the modeling region with the sign of IEDGE indicating the direction of motion

SUBROUTINE EXTRP (IHOUR,ARY,VAL)

Use. This routine seeks a value corresponding to IHOUR in the 24-word data array ARY, assuming that the values in ARY correspond to clock times of 0000, 0100, 0200, etc. If positive or zero, the value found is simply returned. If the desired value is missing (indicated by a negative entry), EXTRP will use linear interpolation to fill the gap, provided the two adjacent values are non-negative. Failing that,

EXTRP will attempt to fill the gap by extrapolation, provided that the gap is no more than two locations wide and that two consecutive non-negative values exist on one side of the gap. A negative value returned indicates that the data gap could not be filled by either method. EXTRP is called by GETAZV.

Arguments. IHOUR = clock time on a 0-2400 basis
ARY = array of 24 data values one hour apart
VAL = data value selected or calculated to correspond to IHOUR

FUNCTION FULGOL (A,ZO)

Use. This function determines inverse Monin-Obukhov length (1/L). FULGOL is called by STABIL.

Arguments. A = Fulle atmospheric stability parameter
ZO = surface roughness length (meters)

SUBROUTINE GETAZV (KOLD,ID,XS,YS,NC,AZV,NGOOD,DDIS)

Use. This routine attempts to retrieve measurement records from the desired number of neighboring wind measurement stations having valid data for the given date and time. GETAZV is called by WINDY.

Arguments. KOLD = 2-word array containing time (on a 0-2400 basis) and date (year, month, day)
ID = 2-word array containing the station identification (may be blank)
XS = x-coordinate of current position
YS = y-coordinate of current position
NC = desired number of close stations (up to 3)
AZV = array containing an azimuth and velocity for each close station found
NGOOD = number of close stations found
DDIS = array containing distances from current position to close stations

SUBROUTINE GETCON (KNEW,NC,IRF)

Use. This routine interpolates and extrapolates measurement station data stored in the array CON to a trajectory node location. Surface temperature and air quality data are normally stored in the array CON. GETCON is called by WINDY.

Arguments. KNEW = 2-word array with the time and date
NC = desired number of stations (up to 3)
IRF = variable equaling 1 or 2 indicating the exponent in the interpolation formula, 1/Distance** IRF

SUBROUTINE JULIAN (IDATE,JULDAT)

Use. This routine converts a 6-character date (YR-MO-DA) to a Julian date. JULIAN is called by METIN.

Arguments. IDATE = 6-character date
JULDAT = a 2-word array of Julian day and year

SUBROUTINE KEKLAY (LMAX,ZMAX,ZHIGH,AKMIN,DH)

Use. This routine determines the K_z coefficients in the Ekman layer. KEKLAY is called by EDDY.

Arguments. LMAX = number of points in the verical temperature sounding
ZMAX = height of maximum potential temperature
ZHIGH = height of highest point in temperature sounding
AKMIN = minimum K_z coefficient allowed
DH = vertical increment for the internal K_z profile

FUNCTION KLASS (A)

Use. This function returns the atmospheric stability class. KLASS is called by STADIF.

Arguments. A = Fulle atmospheric stability parameter

SUBROUTINE KZDATA (IERR)

Use. This routine reads the input data for the eddy diffusivity submodule. KZDATA is called by EDDY.

Argument. IERR = output flag which is returned with a positive value when an error is encountered in reading the data

SUBROUTINE METIN (IDT,LTAPE)

Use. This subroutine retrieves wind, surface temperature, air quality, and solar radiation data from the RAMS data tape for one day. METIN edits, stores and computes regional averages of various atmospheric parameters. METIN is called by WINDY.

Arguments. IDT = 6-character date (YR-MO-DA)
LTAPE = logical unit number of the peripheral device with the meteorological data

SUBROUTINE PLACIT (DT,L,VX,VY,POS,NDX)

Use. PLACIT computes the coordinates and grid square indices of a new trajectory point. PLACIT is called by CROSIT.

Arguments. DT = time duration of this trajectory segment, minutes
L = index of last trajectory point in trajectory data arrays
VX = x-component of V(L) (input)
VY = y-component of V(L) (input)
POS = 2-word array containing x and y coordinates of new position
NDX = 2-word array containing grid square indices (I,J) of new point

SUBROUTINE RUFNES (ZO,J)

Use. This routine determines the surface roughness length at a particular location. RUFNES is called by EDDY.

Arguments. ZO = surface roughness length (meters)
J = integer corresponding to the hour of the day plus one

SUBROUTINE SETIN

Use. This routine initializes the names and coordinates of all measurement stations. SETIN also designates the total number of measurement stations. SETIN is called by WINDY.

Argument. None

SUBROUTINE SETTUP

Use. SETTUP serves as the initialization routine for the generation of trajectory grid square crossing schedules and the trajectory printer-plot. SETTUP is called by CROSIT.

Argument. None

SUBROUTINE SKY (UV,CLOUDY,IDATE)

Use. SKY generates sky clearness ratios from ultraviolet radiation data. SKY is called by METIN.

Arguments. UV = a 24-word array of hourly ultraviolet radiation data in Langley's per minute
CLOUDY = a 24-word array of hourly sky clearness ratios used in the chemical-diffusion module
IDATE = 6-character date (YR-MO-DA)

SUBROUTINE SMOOTH (ZW,ZO,USFC,ZHIGH,DH,AKMIN,ISKLAS)

Use. This routine smooths the raw vertical temperature profile and divides the profile into layers on the basis of lapse rates. SMOOTH is called by EDDY.

Arguments. ZW = height at which surface wind data was measured (meters)
ZO = surface roughness length (meters)
USFC = wind velocity measured at Z = ZW

ZHIGH = height of highest point in vertical temperature profile

DH = vertical increment for internal K_z profile

AKMIN = minimum value of K_z (m^2/min)

ISKLAS = atmospheric stability class on a scale of 1 to 6 corresponding to Pasquill-Gifford classes A to F

SUBROUTINE STABIL (STAB,J,A,ZTOP,DUDZ,DELT,U,ZHTMIX,USTAR,ZW,ZO)

Use. This routine determines the Fulle stability parameter, the Monin-Obukhov length, and the internal mixing layer height from the vertical temperature gradient within an atmospheric layer. STABIL is called by STADIF.

Arguments. STAB = Monin-Obukhov length (L)

J = index for the atmospheric layer

A = Fulle atmospheric stability parameter

ZTOP = highest elevation in the atmospheric layer

DUDZ = vertical wind shear

DELT = vertical temperature gradient in the atmospheric layer

U = surface wind velocity

ZHTMIX = calculated mixing layer height

USTAR = friction velocity

ZW = height at which surface wind data was measured (meters)

ZO = surface roughness length (meters)

SUBROUTINE STADIF (ZW,ZO,USFC,ZHIGH,DH,AKMIN,ISKLAS)

Use. This routine controls the calculation of stability and diffusivities. STADIF is called by EDDY.

Arguments. ZW = height at which surface wind data was measured (meters)

ZO = surface roughness length (meters)

USFC = surface wind velocity

ZHIGH = height of highest point in temperature sounding

DH = vertical increment for the internal K_z profile

AKMIN = minimum value of K_z

ISKLAS = atmospheric stability class on a scale of 1 to 6 corresponding to Pasquill-Gifford classes A to F

FUNCTION TIMIN (T)

Use. Converts time from 0-2400 basis to minutes past midnight (0-1440). TIMIN is called by CROSIT, SETTUP, and WINDY.

Argument. T = clock time on a 0-2400 basis

FUNCTION UNIDOT (A,B,)

Use. Returns the dot product of a unit vector parallel to A with a unit vector parallel to B, i.e., the cosine of the acute angle separating the vectors A and B. UNIDOT is called by BARIER.

Arguments. A = a 3-component vector

B = a 3-component vector

SUBROUTINE UNITV (A,B)

Use. Returns B, the unit vector parallel to vector A. UNITV is called by ANGTST.

Arguments. A = a 3-component vector

B = a 3-component unit vector parallel to A

SUBROUTINE UXYPOS

Use. This routine establishes the wind speed and location information used by EDDY. UXYPOS is called by EDDY.

Argument. None

SUBROUTINE WINDRD (KPWDAT,LTAPE)

Use. This subroutine is used in non-St. Louis applications to read the wind measurement data. For the St. Louis application, a RETURN statement has been inserted near the beginning of the routine so that it acts like a dummy subroutine. WINDRD is called by WINDY.

Arguments. KPWDAT = a Hollerith variable equaling YES when it is to print the wind data read from cards

LTAPE = logical unit of the peripheral device with the wind data. Use LTAPE = 3 for cards.

SUBROUTINE WINDY (IWAS,LTAPE)

Use. WINDY is the secondary driver of the meteorological module. This routine generates the air trajectories (backward or forward) from a prescribed start location and time. WINDY is called by program METMOD.

Arguments. IWAS = flag which is equal to zero on the first call and greater than zero on subsequent calls

LTAPE = logical unit number of the peripheral device with the meteorological data

SUBROUTINE WSECLS (X,Y,DIS,IDIS)

Use. This routine calculates squared distances from the point X,Y to each measurement station in SSTAN, and then orders the stations by increasing distance out to a maximum value (RMAX). WSECLS is called by GETAZV and WINDY.

Arguments. X = x-coordinate of current position

Y = y-coordinate of current position

DIS = array of squared distances from X,Y to all measurement stations. Those values of DIS less than RMAX are arranged in increasing order at the top of the table

IDIS = array of station indices corresponding to entries in the DIS array. (The indices identify the stations in the SSTAN array).

BLOCK DATA

Use. This block data program contains the variable size grid square identification numbers distributed to a fixed size grid square array, IVG. The current grid square identifiers correspond to grid squares varying in size from 1 x 1 kilometer to 10 x 10 kilometers in a St. Louis grid with origin at UTM coordinates x = 680 (km) and y = 4230 (km). The array (IVG) has been dimensioned 100 by 100 corresponding to a 100 kilometer square region considered as the emissions grid. In the portions of the region where grid square identifiers are absent, the array contains zeroes.

2. Emissions Module Subroutines and Their Use

PROGRAM EMMOD

Use. EMMOD is the main program for the emissions module. It reads all the card-image data into the proper arrays and controls the over-all generation of source emissions schedules along trajectories.

Note. When modified with computer installation specific control statements for rewinding and/or skipping files, the program can be used for multiple trajectories on the same day or different days.

SUBROUTINE ADHOUR

Use. This routine inserts even hour values in an arbitrary temporal schedule and interpolates variables associated with the schedule to the even hours. ADHOUR is called by EMMOD.

Argument. None

SUBROUTINE AREAEM (LUAREA,IPUNCH,KOK)

Use. This routine reads the RAPS area source emissions file and determines the area source emissions schedule along the trajectory. AREAEM is called by EMMOD.

Arguments. LUAREA = logical unit of peripheral device with area source emissions data

IPUNCH = a Hollerith variable equal to YES when punched output is desired

KOK = a completion flag which is returned with a positive value if an error is encountered

SUBROUTINE DHPLUM (TA,TS,U,ISTAB,ACMM,HTINV,HTPS,DH,PENTFR,DTDZ)

Use. This subroutine calculates the plume rise and the inversion layer penetration fraction for a point source. DHPLUM is called by PONTEM.

Arguments.

TA = ambient temperature (°C)

TS = stack effluent temperature (°C)

U = wind speed (meters/second)

ISTAB = atmospheric stability class

ACMM = stack effluent volumetric flow rate (cubic meters/minute)

HTINV = inversion base height above surface (meters)

HTPS = stack height (meters)

DH = plume rise (meters)

PENTFR = inversion penetration fraction (0. to 1.)

DTDZ = potential temperature gradient of stable layer (°C/meter)

FUNCTION DISTAN (IPT,TS,TE)

Use. This function returns the distance along a trajectory between two times. DISTAN is called by PLUMAS.

Arguments.

IPT = index of the last trajectory node

TS = initial time (minutes)

TE = final time (minutes)

FUNCTION ERF (X)

Use. ERF returns the value of the error function of X. ERF is called by PLUMAS and PARTIT.

Argument. X = the independent variable

SUBROUTINE GESTAB (TIME,ISTAB,DTNEXT,NXSTAB)

Use. This routine determines the atmospheric stability at the specified time. In addition, the value of and time difference to the next stability class change are determined. GESTAB is called by PONTEM and PLUMAS.

Arguments.

TIME	=	current time (minutes)
ISTAB	=	current stability class
DTNEXT	=	time difference to next stability class change
NXSTAB	=	next stability class value

SUBROUTINE GRIDIT (LUGRID,KOK)

Use. This subroutine reads the RAPS area source grid identifiers and corresponding areas. GRIDIT is called by EMMOD.

Arguments.

LUGRID	=	logical unit peripheral device with RAPS grid description
KOK	=	a completion flag which is returned positive if an error is encountered

SUBROUTINE LOCATE (IPT,XP,YP,PDIST,TPASS,KFLAG,ISTAB)

Use. This routine examines the location of a point source relative to a trajectory segment. It determines whether or not the point source is passed on a particular trajectory segment and whether or not it is close enough to be considered for the point source emissions schedule. LOCATE is called by PONTEM.

Arguments.

IPT	=	index of the last trajectory node
XP	=	point source x-coordinate (km)
YP	=	point source y-coordinate (km)
PDIST	=	perpendicular distance to XP, YP
TPASS	=	time source is passed (minutes)
KFLAG	=	a variable which is positive when a point source's location meets all criteria to be included on the segment
ISTAB	=	atmospheric stability class

SUBROUTINE PARTIT (X,HT,HTINVR,HTINTF,NOSTAT,ISTAB,TALL,PENTFR,VFRACT)

Use. This routine is used to distribute the elevated point source's emissions to vertically partitioned atmospheric layers. PARTIT is called by PONTEM.

Arguments.

X	=	downwind distance from source (km)
HT	=	height of source above surface (meters)
HTINVR	=	height of inversion base or mixing depth (meters)
HTINTF	=	vector of heights of layer interfaces (meters)
NOSTAT	=	number of layers
ISTAB	=	atmospheric stability class
TALL	=	Hollerith variable indicating whether or not the source is a "tall stack" (YES or NO)
PENTFR	=	inversion penetration fraction (0. to 1.)
VFRACT	=	a vector of the normalized vertical distribution of the source's entrained emissions

SUBROUTINE PLUMAS (IPT,TPASS,PDIST,PLUMFR,ISTAB,DXFREZ,TALL,KPLAG)

Use. PLUMAS is used to determine the fraction of point source's plume which is present in the Lagrangian parcel after a fixed time interval from interception of the plume. PLUMAS is called by PONTEM.

Arguments.

IPT	=	index of the current trajectory node
TPASS	=	input as time of day when source is passed, and returned as time of day when plume is intercepted (minutes)
PDIST	=	perpendicular distance to source at TPASS (km)
PLUMFR	=	fraction of plume's mass present in the parcel after the fixed time interval
ISTAB	=	atmospheric stability class

DXFREZ = a fixed time interval between the plume interception and the lateral distribution evaluation

TALL = a Hollerith variable indicating whether or not the source is a "tall stack" (YES or NO)

KPLAG = a flag returned with a value corresponding to the characteristics listed below:

-1 = plume's lateral spread is too slow to reach the parcel (source ignored)

1 = plume's lateral diffusion out of the parcel is dominant

2 = plume's lateral diffusion into the parcel is dominant

SUBROUTINE PONTEM (LUPONT,IPUNCH,KOK,NDSUMS)

Use. This routine controls the generation of point source emission schedules along a trajectory. PONTEM is called by EMMOD.

Arguments. LUPONT = logical unit of peripheral device with point source emissions data

IPUNCH = a Hollerith variable equal to YES when punched output is desired

KOK = a completion flag which is returned with a negative value when errors are encountered

NDSUMS = a Hollerith variable equal to YES when regional sums of the point source emissions inventory are desired

SUBROUTINE SEGMET (P)

Use. This routine computes the length and bisector angles for each segment of the trajectory. In addition, it computes the maximum radii and perpendicular distance criteria used in evaluating which point sources will have plumes potentially intersecting the trajectory. SEGMET is called by PONTEM.

Argument. P = vector of trajectory node locations

SUBROUTINE STACKS (TIME,LUPONT,KOK,IWAS,NDSUMS)

Use. This routine reads the RAPS point source emissions file each hour and determines the regional point source emission totals if desired. STACKS is called by PONTEM.

Argument.

TIME	=	time of day (minutes)
LUPONT	=	logical unit of peripheral device with the point source emissions data
KOK	=	a flag which returns with a negative value if errors are encountered
IWAS	=	a flag which is input as zero on the first call to the subroutine and input as one on all subsequent calls
NDSUMS	=	a Hollerith variable equal to YES when regional sums of the point source emissions inventory are desired

FUNCTION TRPLOG (X,XS,FS,NXF,NF)

Use. This function performs log-log interpolation of tabular functions. TRPLOG is called by PLUMAS and PARTIT.

Arguments.

X	=	value of independent variable
XS	=	vector of independent variable values corresponding to the tabulated function values
FS	=	array of tabulated function values
NXF	=	number of tabulated function values
NF	=	number of functions in FS array (normally 1)

BLOCK DATA

Use. The emissions module block data program contains Gaussian dispersion coefficients used in the point source submodule. The σ_y and σ_z coefficients for typical stack heights are tabulated by atmospheric stability class and distance. For tall stacks, the coefficients a, b, c, and d used in the equations

$$\sigma_y = ax^b$$

$$\sigma_z = cx^d$$

are included as functions of atmospheric stability class.

3. Chemical-Diffusion Module Subroutines and Their Use

PROGRAM KEMOD

Use. KEMOD is the main program for the chemical-diffusion module. This program reads the emissions data and calls KEMOD2 to perform the remainder of the input/output and program control.

SUBROUTINE ADJUST (Y,NO)

Use. This routine is a part of the EPISODE integration package. It is used to adjust the concentrations history array upon a reduction of the order of the approximation. ADJUST is called by TSTEP.

Arguments. Y = a 2-dimensional array containing the concentrations history
NO = the maximum number of differential equations

SUBROUTINE BLKDEC (QB,NN,KK,IP,A,C,TL,W,NOGO,BCFLAG)

Use. This routine performs LU decomposition of a block-tridiagonal matrix system of linear equations. BLKDEC is called by PSET.

Arguments. QB = a 3-dimensional array containing the diagonal submatrices
NN = order of diagonal submatrices
KK = block order of the system
IP = a 2-dimensional array of order NN*KK used for storing pivot information
A = a vector of (KK-1) coefficients representing the lower diagonal submatrices which are scaled identify matrices in this problem
C = a vector of (KK-1) coefficients representing the upper diagonal submatrices which are scaled identity matrices in this problem

TL = a 3-dimensional array used to store the lower diagonal elements of Q

W = a work vector of length NN

NOGO = a completion code which is returned as one when a singular matrix is encountered

BCFLAG = an integer vector of surface boundary condition codes

Remarks. The matrix is of the form

$$Q = \begin{bmatrix} QB(1) & C(1) & o & o & o \\ A(1) & QB(2) & C(2) & o & o \\ o & . & . & . & o \\ o & o & . & . & C(KK-1) \\ o & o & o & A(KK-1) & QB(KK) \end{bmatrix}$$

BLKDEC is designed only to be used with BLKSOL.

SUBROUTINE BLKSOL (QB,NN,KK,RHS,IP,A,C,TL,W,BCFLAG)

Use. This routine solves a block-tridiagonal matrix system of linear equations QX = RHS after LU decomposition has been performed on Q. BLKSOL is called by TSTEP.

Arguments. Same as BLKDEC except RHS and deletion of NOGO.

RHS = input as the right-hand side vector and returned as the solution vector

SUBROUTINE CHECKY (T,Y,YO,N,KOK)

Use. This routine is used to check for negative concentrations. CHECKY is called by DRIVE.

Arguments. T = time of day (minutes)

Y = a two-dimensional array containing the concentrations history

YO = the vector of concentrations last passed to subroutine DRIVE from KEMOD2

N = the number of differential equations

KOK = a completion code which is returned with a negative value when negative concentrations are encountered

SUBROUTINE COPLOT (NDPT)

Use. This routine controls the generation of a printer-plot of up to five ground concentrations versus time.

Argument. NDPT = the number time increments to plot

SUBROUTINE COSET

Use. This routine is a part of the EPISODE package. It is used to set coefficients for the predictor-corrector formula. COSET is called by TSTEP.

Argument. None

SUBROUTINE DEC (N,NDIM,A,IP,IER)

Use. This subroutine performs the LU decomposition of a matrix. DEC is called by BLKDEC.

Arguments.

N = order of the matrix A

NDIM = declared dimension of the matrix A

A = on input, a matrix to be triangularized; on output, A contains both the upper and lower elements of the L and U matrices

IP = a vector of pivot indices

IER = a flag returned nonzero when A is a singular matrix

Remarks. DEC is designed only to be used with SOL.

SUBROUTINE DIFCOF (NOSTAT)

Use. This routine calculates the scaled eddy diffusivity coefficients used in the finite difference form of the diffusion equation. DIFCOF is called by KEMOD2.

Argument. NOSTAT = number of vertical mesh points

SUBROUTINE DIFFUN (N,T,Y,YDOT)

Use. DIFFUN defines the finite difference ordinary differential equations (YDOT = F[Y,T]). It computes the boundary condition fluxes at time T and the YDOT vector. DIFFUN is called by TSTEP.

Arguments.
- N = number of differential equations
- T = time of day in minutes
- Y = a 2-dimensional array containing the concentrations history
- YDOT = the rate of change of Y with respect to time

SUBROUTINE DRIVE (N,TO,HO,YO,TOUT,EPS,IERROR,MF,INDEX,BIGSTP,KOK)

Use. DRIVE is the driver subroutine for the EPISODE package. It performs initialization and control functions for the EPISODE package.

Arguments.
- N = number of differential equations
- TO = initial time (minutes)
- HO = initial time step size (minutes)
- YO = initial concentration vector
- TOUT = time at which output is desired
- EPS = integration error control criteria
- IERROR = error control method
- MF = method flag which must equal 21 for this version of EPISODE
- INDEX = EPISODE input control parameter and output completion code
- BIGSTP = maximum time step size allowed
- KOK = a completion code returned as negative if an error is encountered

SUBROUTINE FMAX (A,N,B)

Use. This routine determines the maximum value of the elements of the vector A. FMAX is called by COPLOT.

Arguments. A = vector
N = number of elements of A (N≥2)
B = maximum value

SUBROUTINE INTERP (TOUT,Y,NO,YO)

Use. INTERP is used by the EPISODE package to interpolate the values of Y to the desired output time. INTERP is called by DRIVE.

Arguments. TOUT = desired output time (minutes)
Y = a 2-dimensional array containing the concentrations history
NO = the number of differential equations
YO = a vector of the concentrations at T = TOUT

SUBROUTINE ISTATE

Use. This routine calculates initial conditions for 17 species based on the steady-state assumption. ISTATE is called by KEMOD2.

Arguments. None

Remarks. This routine is specific to the ERT 4.1.78 photochemical mechanism.

SUBROUTINE KEMOD2 (INTIM,TSTOP,NUMFLX,FLXUNT)

Use. KEMOD2 is the secondary driver for the chemical-diffusion module. It controls all the input/output except for the emissions data. It calls the EPISODE integration package and controls all the updating processes. KEMOD2 is called by program KEMOD.

Arguments. INTIM = initial time in minutes
TSTOP = final time in minutes
NUMFLX = number of species with area source emissions input
FLXUNT = a Hollerith variable equal to MOLE if the emissions data is input in mole fraction instead of mass fraction

SUBROUTINE MATMUL (A,B,C,N,M,L)

Use. This routine computes the matrix product C = A x B.

Arguments. A = matrix of order N x M
B = matrix of order M x L
C = A x B
N = number of rows of A
M = number of columns of A and rows of B
L = number of columns of B

FUNCTION ITHOUR (TMIN)

Use. ITHOUR converts time of day in minutes to time on the 2400-military clock.

Arguments. TMIN = time of day in minutes

SUBROUTINE JACOB (A,C,R,N)

Use. JACOB computes the Jacobian matrix of the chemical rate equations having the form $dc_i/dt = f(c_1, c_2, \ldots, c_n)$, where c_i is the concentration of the ith species. Each element of the Jacobian is defined by $A_{ij} = \partial f_i/\partial c_j$, i, j = 1, 2, ..., N.

Arguments. A = Jacobian matrix of size N x N
C = vector of species concentrations
R = vector of chemical rate constants
N = order of the Jacobian matrix

Remarks. This routine is specific to the ERT photochemical mechanism of 4.1.78.

SUBROUTINE PEDERV (NO,Y,Q,NKM)

Use. This routine computes the block-tridiagonal Jacobian matrices by calling JACOB for each vertical cell. PEDERV is called by PSET.

Arguments. NO = number of differential equations

Y = a 2-dimensional array containing the concentrations history

Q = a 3-dimensional array containing the block-diagonal Jacobian submatrices

NKM = number of species being integrated

SUBROUTINE PHOTOD (JSTART,JSTOP,NOSTAT,ZEE)

Use. This routine calculates diurnal schedules of elevation dependent photodissociation rates for the following reactions:

$$NO_2 + h\nu \xrightarrow{k_1} O(^3P) + NO$$

$$HCHO + h\nu \begin{cases} \xrightarrow{k'_{38}} HCO + H \\ \xrightarrow{k''_{38}} CO + H_2 \end{cases}$$

$$k_{38} = k'_{38} + k''_{38}$$

PHOTOD is called by KEMOD2.

Arguments. JSTART = start time on the 2400-clock

JSTOP = stop time on the 2400-clock

NOSTAT = number of vertical mesh points

ZEE = vector of vertical mesh point elevations

SUBROUTINE PSET (Y,NO,CON,MITER,IER)

Use. PSET is a part of the EPISODE package. It computes the coefficients of the matrix P = I - [H/EL(2)]*J where I is an identity matrix, H is the time step size, EL(2) is a corrector coefficient, and J is the block-diagonal Jacobian matrices. PSET is called by TSTEP.

Arguments. Y = a 2-dimensional array containing the concentrations history

NO = number of differential equations

CON = scaling coefficient equal to H/EL(2)

MITER = a flag for the method of determining Jacobian which must equal 1 for this version of PSET

IER = a flag denoting that a singular matrix has been encountered

SUBROUTINE RATEHI (K)

Use. This routine retrieves photodissociation rates for the Kth mesh point and computes other photodissociation rates which are proportional to those rates computed by PHOTOD. RATEHI is called by KEMOD2, RATES, PEDERV, and ISTATE.

Argument. K = index of the vertical mesh point for which the rates are desired

SUBROUTINE RATES (Y,YDOT)

Use. This subroutine computes the chemical rates using explicit rate equations.

Arguments. Y = a 2-dimensional array containing the concentrations history

YDOT = a vector of the rate of change of concentrations with respect to time

Remarks. This routine is specific to the 4.1.78 ERT chemical mechanism.

SUBROUTINE SCALE (Y1,Y2,YB,YT)

Use. This routine computes the minimum and maximum printer-plot boundaries such that the increments have a minimum number of significant figures.

Arguments. Y1 = minimum value to be plotted

Y2 = maximum value to be plotted

YB = minimum value of the plot boundary

YT = maximum value of the plot boundary

SUBROUTINE SKEDUL (X,T,NX,NINT,UPDELT,INTIM,TSTOP,DELTIN,FAIL,INT)

Use. This routine can perform three different transformations of tabulated functions.

1) It transforms a tabulated function with arbitrarily spaced independent variable increments to a tabulated function with evenly spaced independent variable increments.

2) It transforms a tabulated function input on one evenly spaced independent variable increment to a tabular function with a different increment.

3) It performs linear interpolation of a sparsely tabulated function to create a tabulated function on an evenly spaced independent variable increment. SKEDUL is called by KEMOD2.

Arguments.

X	=	a tabulated function vector
T	=	an independent variable vector corresponding to X
NX	=	number of elements of X upon input
NINT	=	number of elements of X returned
UPDELT	=	even increment of the independent variable
INTIM	=	initial value of independent variable
TSTOP	=	final value of independent variable
DELTIN	=	even increment of independent variable on input for transformation 2
FAIL	=	a Hollerith variable equal to YES if an error is encountered in the transformation process
INT	=	a flag which is input positive when transformation 3 is desired

SUBROUTINE SOL (N,NDIM,A,B,IP)

Use. This routine solves a linear system of equations AX = B where A matrix has been decomposed into lower and upper triangular form by DEC. SOL is called by BLKSOL.

Arguments. N = order of the A matrix
NDIM = dimensioned size of the A matrix
A = the matrix decomposed into lower and upper triangular form by subroutine DEC
B = the right-hand side vector
IP = the pivot vector obtained from DEC

SUBROUTINE SOL2B2 (A,B)

Use. This routine solves a system of two linear equations of the form AX = B.

Arguments. A = a two-by-two matrix of coefficients of the equations
B = input as the right-hand side vector and returned as the solution vector

SUBROUTINE STEADY (Y,N)

Use. A dummy subroutine as used in this version of the model. It can be modified to compute the concentration of species carried in pseudo-stationary state.

Arguments. Y = a 2-dimensional array containing the concentrations history
N = number of differential equations

SUBROUTINE TEMPR (IT,TIME,T)

Use. This routine computes the temperature dependent rate constants associated with the following two chemical decomposition steps:

$$HNO4 \xrightarrow{k_{21}} NO2 + HO2$$

$$PAN \xrightarrow{k_{37}} NO2 + RCO3$$

Arguments. IT = index of the temperature vector associated with the current temperature
TIME = time in minutes
T = a vector of temperatures in degrees Kelvin

SUBROUTINE TIMEX

Use. This is a dummy subroutine as used in this version of the model.

Argument. None

SUBROUTINE TSTEP (Y,NO)

Use. This routine performs one step of the integration of the differential equations. It is the core integrator of the EPISODE package.

Arguments. Y = a 2-dimensional array containing the concentrations history
NO = number of differential equations

SUBROUTINE UNMIXR

Use. This is a dummy subroutine as used in this version of the model.

Argument. None

SUBROUTINE UPFLX1 (TIME,J)

Use. This routine updates the area source emission fluxes.

Arguments. TIME = time in minutes
J = index of the area source array (FLXIN) corresponding to TIME

SUBROUTINE UPRAT2 (T,IK1,RATEV,NOSTAT,NVRATE,CLOUDY)

Use. This routine updates the photodissociation rates to the current time from the schedule of rates created by subroutine PHOTOD. It adjusts the rates by the sky clearness ratio.

Arguments. T = time in minutes

IK1 = index of the RATK1 and RATK2 arrays corresponding to time T

RATEV = a 2-dimensional array in which the current NO_2 and HCHO photolysis rates at the various elevations are stored

NOSTAT = number of vertical mesh points

NVRATE = number of variable rates

CLOUDY = sky clearness ratio (0 to 1)

SUBROUTINE UPSORC (TIME,IPS,NOSTAT,SPEC)

Use. This routine updates the point source emission rates.

Arguments. TIME = time in minutes

IPS = index of the point source emissions array (PS) corresponding to TIME

NOSTAT = number of vertical mesh points

SPEC = a vector of Hollerith species names

4. Utility Library Subroutines and Their Use

SUBROUTINE FMINF (F,NF,FMIN,NMIN)

Use. FMINF finds the minimum value in a linear array and returns both the minimum value and its array index relative to the first value in the array. Called by WSECLS and PONTEM.

Arguments. F = the first location of a sequence to be checked

NF = the number of consecutive locations to be considered (starting with F)

FMIN = the minimum value among the NF consecutive members of the F array

NMIN = the index of FMIN relative to F, counting F as 1

SUBROUTINE MCHAR (IFC,FROM,ITC,TO,NCHR)

Use. MCHAR moves a string of individual characters from one storage location to another. MCHAR is called by SETPLT.

Arguments.

IFC	=	position of character to move from. The left most character is position 1
FROM	=	source word
ITC	=	position of character to move to
TO	=	destination word
NCHR	=	number of characters to move

SUBROUTINE MDATE (IDATE)

Use. A dummy subroutine that is located in the code in a position where it would access the machine date.

Argument. IDATE = machine date

SUBROUTINE NEWPAG (TITLE,NSKIP,IDATE)

Use. This routine prints the TITLE as a header on each page of output.

Arguments.

TITLE	=	an 80-character array containing any user comment or title
NSKIP	=	number of pages to skip (normally 0)
IDATE	=	machine date

SUBROUTINE PREDAT

Use. This routine reads the input data cards for each module of the code from logical unit number five. It writes the data card images on the output file, logical unit number six. It also writes the data card images on logical unit number three, and rewinds this file, so that they can be read by the program.

Argument. None

SUBROUTINE SECOND (A)

Use. SECOND is a dummy subroutine which is positioned in the code to access the computer time used from the beginning of the run.

Argument. A = computer time used in seconds

SUBROUTINE SETPLT (A,B,C,D)

Use. SETPLT is the initialization entry for a general purpose printer plot routine. SETPLT is called by SETTUP and COPLOT.

Arguments. A = minimum value on the horizontal (x) axis
B = maximum value on the x-axis
C = minimum value on the vertical (y) axis
D = maximum value on the y-axis

ENTRY PLTPNT (A,B,C)

Use. PLTPNT enters a new data point in the printer-plot array.

Arguments. A = x-coordinate of new data point
B = y-coordinate of new data point
C = a Hollerith character to be plotted at x,y

ENTRY PLTOUT (IFILE)

Use. PLTOUT is the final call in a plotting sequence. It writes the plot to the output file.

Argument. IFILE = the logical unit of output file

SUBROUTINE SOLAR (SLA,SLO,TZ,IY,IM,ID,D,TIME,NV)

Use. SOLAR is used to compute the solar elevation in degrees. SOLAR is called by SKI and PHOTOD.

Arguments. SLA = latitude in degrees (south = minus)
SLO = longitude in degrees (east = minus)
TZ = time zone

IY = year
IM = month
ID = day
D = solar elevation angle in degrees
TIME = time on the 0-2400 hour clock
NV = 5 for solar elevation calculation

SUBROUTINE XMIT (N,A,B)

Use. When N > 0, XMIT transfers N words from A to B. When N < 0, the scalar value of A is transmitted to N words of B. XMIT is called in all three modules by many different subroutines.

Arguments. N = number of words to transfer
A = value of vector (or scalar) to transfer
B = vector receiving new values

APPENDIX C

CODE FORTRAN SOURCE LISTINGS

1. Meteorological Module

	Page
METMOD (Main)	C-3
ANGTST	C-5
AZVDIS	C-6
BARIER	C-7
CELAVG	C-9
CLOCKT	C-9
CROSIT	C-10
DATE	C-14
DIFFUS	C-15
DOTTY	C-16
DTIME	C-16
EDDY	C-16
EDGE	C-23
EXTRP	C-23
FULGOL	C-26
GETAZV	C-26
GETCON	C-31
JULIAN	C-33
KEKLAY	C-34
KLASS	C-35
KZDATA	C-35
METIN	C-37
PLACIT	C-44
RUFNES	C-44
SETIN	C-45
SETTUP	C-47
SKY	C-49
SMOOTH	C-49
STABIL	C-51
STADIF	C-52
TIMIN	C-52
UNIDOT	C-53
UNITV	C-53
UXYPOS	C-53
WINDRD	C-54
WINDY	C-56
WSECLS	C-64
BLOCK DATA	C-65

2. Emissions Module

	Page
EMMOD (Main)	C-78
ADHOUR	C-83
AREAEM	C-83
DHPLUM	C-89
DISTAN	C-90
ERF	C-91
GESTAB	C-91
GRIDIT	C-92
LOCATE	C-93
PARTIT	C-94
PLUMAS	C-97
PONTEM	C-98
SEGMET	C-103
STACKS	C-104
TRPLOG	C-107
BLOCK DATA	C-108

3. Chemical-Diffusion Module

	Page
KEMOD (Main)	C-109
ADJUST	C-110
BLKDEC	C-112
BLKSOL	C-114
CHECKY	C-115
COPLOT	C-116
COSET	C-117
DEC	C-119
DIFCOF	C-120
DIFFUN	C-121
DRIVE	C-123
FMAX	C-130
INTERP	C-130
ISTATE	C-131
ITHOUR	C-133
JACOB	C-133
KEMOD2	C-137
MATMUL	C-152
PEDERV	C-152
PHOTOD	C-153
PRODUK	C-156
PSET	C-156
RATEHI	C-158
RATES	C-158
SCALE	C-161
SKEDUL	C-161
SOL	C-163
SOL2B2	C-164
STEADY	C-164
TEMPR	C-165
TIMEX	C-165
TSTEP	C-165
UNMIXR	C-173
UPFLX1	C-173
UPRAT2	C-174
UPSORC	C-175

4. Utility Library

	Page
FMINF	C-176
MCHAR	C-176
MDATE	C-177
NEWPAG	C-177
PREDAT	C-177
SECOND	C-178
SETPLT	C-178
SOLAR	C-182
XMIT	C-183

1. Meteorological Module Listing

```
      PROGRAM METMOD                                                    MM     1
C                                                                       MM     2
C       THIS IS THE DRIVER PROGRAM FOR THE ERT METEORELOGICAL MODULE.   MM     3
C                                                                       MM     4
C       SUBROUTINES AND FUNCTIONS  REQUIRED                             MM     5
C                                                                       MM     6
C           ANGTST    AZVDIS    BARIER    CELAVG    CLOCKT    DATE      MM     7
C           FMINF     FULGOL    GETAZV    GETCON    JULIAN    KEKLAY    MM     8
C           KZDATA    MCHAR     MDATE     METIN     NEWPAG    PLACIT    MM     9
C           PREDAT    RUFNES    SECOND    SETIN     SETPLT    SETTUP    MM    10
C           SKY       SMOOTH    SOLAR     STABIL    STADIF    TIMIN     MM    11
C           UNIDOT    UNITV     UXYPOS    WINDY     WINDRO    WSECLS    MM    12
C           XMIT      CROSIT    KLASS     DTIME                         MM    13
C                                                                       MM    14
C       BLOCK DATA IS REQUIRED FOR VARIABLE GRID SQUARE SIZE APPLICATIOMM    15
C                                                                       MM    16
C       LOGICAL UNITS REQUIRED                                          MM    17
C                                                                       MM    18
C               TAPE 1 = PUNCHED OUTPUT                                 MM    19
C               TAPE 3 = PREDAT OUTPUT / METMOD INPUT                   MM    20
C               TAPE 4 = SETPLT SCRATCH FILE                            MM    21
C               TAPE 5 = INPUT (TO PREDAT)                              MM    22
C               TAPE 6 = PRINTED OUTPUT                                 MM    23
C               TAPES 11-20 = POSSIBLE WIND DATA SCRATCH FILES          MM    24
C               LTAPE = METEOROLOGICAL DATA FILE (USE 3 FOR CARDS)      MM    25
C                                                                       MM    26
C                                                                       MM    27
      COMMON /AIRDAL/ NEEDAQ,   NOSPEC,   CONAM(10),   CON(24,25,10)   MM    28
      COMMON /CNTROL/ KSTOP,             TSUN                         MM    29
      COMMON /DATES/  IDATES(10),  ICT(10),  NFILES                   MM    30
      COMMON  /GRID/  X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT        MM    31
      COMMON/INPUTS/  TITLE(20),         JDATE(10),          NCURV    MM    32
      COMMON /KZINPT/ NEEDKZ,            KZPRTX,             NUMSD,   MM    33
     1                TIMESD(3),        .TEMPSD(50,3),       ZELVSD(50,3)MM 34
     2          ,     TEMPSF(24),        NPTSD(3)       ,    NKZDAT   MM    35
      COMMON /ORIGIN/ UTMXOR, UTMYOR                                  MM    36
      COMMON /REUSE/  KPSTAT,  KPWDAT,  CONVRT,  KXTRA, KPWIND       MM    37
      COMMO. /TRAJ/   TSTART,              P(2,100),          ID(2,100) MM   38
      COMMON /WDATA/  NUMSTA,  SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX MM   39
      COMMON /WHERE/  RLAT,              RLONG,            TMZONE     MM    40
      COMMON /WIND/   T(100),              V(100),            TH(100),  MM   41
     1                NPTS                                           MM    42
      COMMON /WINFLD/ ANDATA(52,25) ,DISAVE(25),   IDSAVE(25)         MM    43
      EQUIVALENCE  (RMOR,MORE)                                        MM    44
      DATA YES,NEG,MORE,END/3HYES,2HNO,4HMORE,3HEND/                  MM    45
      DATA DALITE/4HDAYL/                                             MM    46
      DATA RTD/57.29578/                                              MM    47
      DATA  LIN, LOUT /3,6/                                           MM    48
      IWAS = 0                                                        MM    49
      NCURV = 0                                                       MM    50
      TERM = END                                                      MM    51
C       CREATE DATA CARD IMAGES ....                                  MM    52
100   CALL PREDAT                                                     MM    53
C        TITLE   -- ANY USER SPECIFIED TITLE                          MM    54
C        TRAJEX  -- DID YOU INPUT A TRAJECTORY (T,V,+TH) EXTERNALLY?  MM    55
```

```
C        KPWIND  --  DO YOU WANT THE OUTPUTS PUNCHED?                    MM    56
C        NEEDAQ  --  DO YOU WANT AIR QUALITY DATA INTERPOLATED           MM    57
C                    ALONG THE TRAJECTORY?                               MM    58
C        NEEDKZ  --  VERTICAL DIFFUSITY CALCULATION DESIRED (YES OR NO)  MM    59
C        LTAPE   --  LOGICAL UNIT OF WIND/MET DATA PERIPHERAL.           MM    60
C                                                                        MM    61
      READ(LIN,25) TITLE                                                 MM    62
                                                                         MM    63
      READ(LIN,29) TRAJEX                                                MM    64
      READ(LIN,29) KPWIND                                                MM    65
      READ(LIN,29) NEEDAQ                                                MM    66
      READ(LIN,29) NEEDKZ                                                MM    67
      READ(LIN,28) LTAPE                                                 MM    68
C                                                                        MM    69
      CALL MDATE(JDATE)                                                  MM    70
      IF (TRAJEX.EQ.YES) GO TO 120                                       MM    71
      CALL WINDY (IWAS,LTAPE)                                            MM    72
      IWAS = 1                                                           MM    73
      IF (KSTOP.EQ.1) GO TO 810                                          MM    74
      GO TO 150                                                          MM    75
120   READ(LIN,3) P(1,1),P(2,1)                                          MM    76
      READ(LIN,29) STIME                                                 MM    77
      TSUN = 0.                                                          MM    78
      IF (STIME.EQ.DALITE) TSUN = 0100.                                  MM    79
      DO 130 I = 1,101                                                   MM    80
      READ(LIN,3) A,B,C                                                  MM    81
      IF (A.LT.0.) GO TO 140                                             MM    82
      T(I) = A                                                           MM    83
      V(I) = B                                                           MM    84
      TH(I) = C/RTD                                                      MM    85
130   CONTINUE                                                           MM    86
      WRITE(LOUT,2)                                                      MM    87
      GO TO 810                                                          MM    88
140   NPTS = I-1                                                         MM    89
      CALL TRAJEC(IWAS,LTAPE)                                            MM    90
      IWAS = 1                                                           MM    91
C                                                                        MM    92
150   CONTINUE                                                           MM    93
C                                                                        MM    94
810   READ(LIN,25) TERM                                                  MM    95
      IF (TERM.EQ.RMOR) GO TO 100                                        MM    96
      STOP                                                               MM    97
C                                                                        MM    98
2     FORMAT (1H1,10X,45HDIMENSIONS OF ARRAYS T, V, TH IN COMMON/WIND/   MM    99
     133H HAVE BEEN EXCEEDED. JOB ABORTED.)                              MM   100
3     FORMAT (40X,3F10.3)                                                MM   101
25    FORMAT (20A4)                                                      MM   102
28    FORMAT (40X,I2)                                                    MM   103
29    FORMAT (40X,A4)                                                    MM   104
      END                                                                MM   105
```

```
      SUBROUTINE ANGTST(XS,YS,NGOOD,KUSE,OK)                                MM   106
C                                                                           MM   107
C THIS ROUTINE PERFORMS THE ANGLE TEST                                      MM   108
C                                                                           MM   109
C XS      X-COORDINATE OF POSITION                                          MM   110
C YS      Y-COORDINATE OF POSITION                                          MM   111
C NGOOD   NUMBER OF POINTS-1                                                MM   112
C KUSE    INDEX OF CLOSEST STATION                                          MM   113
C OK      FLAG IF POSITIVE ANGLE TEST IS OK OTHERWISE LOOK FOR NEW POINT    MM   114
C                                                                           MM   115
      COMMON /WDATA/ NUMSTA, SSTAN(2,25), ISTANS(2,25), RMIN, RMAX          MM   116
C                                                                           MM   117
      DIMENSION RVEC(3,3),SVEC(3,3)                                         MM   118
      DATA COSTST/.99939083/                                                MM   119
C                                                                           MM   120
      OK = -1.                                                              MM   121
      NUSE = NGOOD+1                                                        MM   122
      GO TO (100,120,130),NUSE                                              MM   123
C                                                                           MM   124
C STORE FIRST VECTOR                                                        MM   125
                                                                            MM   126
100   CONTINUE                                                              MM   127
      RVEC(1,1) = XS-SSTAN(1,KUSE)                                          MM   128
      RVEC(2,1) = YS-SSTAN(2,KUSE)                                          MM   129
      RVEC(3,1) = 0.0                                                       MM   130
110   CONTINUE                                                              MM   131
      OK = 1.0                                                              MM   132
      RETURN                                                                MM   133
C                                                                           MM   134
C STORE SECOND VECTOR AND CHECK ANGLE                                       MM   135
120   CONTINUE                                                              MM   136
      RVEC(1,2) = XS-SSTAN(1,KUSE)                                          MM   137
      RVEC(2,2) = YS-SSTAN(2,KUSE)                                          MM   138
      RVEC(3,2) = 0.0                                                       MM   139
      CALL UNITV(RVEC(1,1),SVEC(1,1))                                       MM   140
      CALL UNITV(RVEC(1,2),SVEC(1,2))                                       MM   141
      COSEY = DOTTY(SVEC(1,1),SVEC(1,2))                                    MM   142
C                                                                           MM   143
C     PROVIDE A 2 DEGREE ANGLE CHECK                                        MM   144
      IF (COSEY.GT.COSTST) RETURN                                           MM   145
      GO TO 110                                                             MM   146
C                                                                           MM   147
C STORE THIRD VECTOR AND CHECK ANGLE                                        MM   148
130   CONTINUE                                                              MM   149
      RVEC(1,3) = XS-SSTAN(1,KUSE)                                          MM   150
      RVEC(2,3) = YS-SSTAN(2,KUSE)                                          MM   151
      RVEC(3,3) = 0.0                                                       MM   152
      CALL UNITV(RVEC(1,3),SVEC(1,3))                                       MM   153
      COSEY = DOTTY(SVEC(1,1),SVEC(1,3))                                    MM   154
      IF (COSEY.GT.COSTST) RETURN                                           MM   155
      COSEY = DOTTY(SVEC(1,2),SVEC(1,3))                                    MM   156
      IF (COSEY.GT.COSTST) RETURN                                           MM   157
      GO TO 110                                                             MM   158
C                                                                           MM   159
      END                                                                   MM   160
```

```
      SUBROUTINE AZVDIS(NP,AZV,DDD,AZM,VEL,IRF)                          MM   161
C                                                                        MM   162
C THIS SUBROUTINE TAKES THE RAW AZIMUTH AND VELOCITY DATA AND            MM   163
C PRODUCES ONE AZIMUTH AND VELOCITY                                      MM   164
C                                                                        MM   165
C     NP            NUMBER OF RAW DATA POINTS   (UP TO THREE)          MM   166
C     AZV(1)        AZIMUTH IN WIND PT COORDINATES                     MM   167
C     AZV(2)        VELOCITY IN MPH                                    MM   168
C     AZV(3)        NUMBER OF POINTS IN DIGITAL WIND SYSTEM            MM   169
C     AZV(4)        STATION NAME                                       MM   170
C     DDD           DISTANCE FROM INPUT PT TO CLOSEST STATIONS         MM   171
C     AZM           AZIMUTH TO BE CALCULATED                           MM   172
C     VEL           VELOCITY TO BE CALCULATED                          MM   173
C     IRF           INVERSE OR INVERSE SQUARE DISTANCE CALCULATION TO BE MM   174
C                                                                        MM   175
      DIMENSION AZV(12),DDD(3)                                           MM   176
      DIMENSION AZT(3),W(3),R(3),AX(3),AY(3)                             MM   177
      COMMON /WDATA/ NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX    MM   178
      DATA RPD/.01745329/,DPR/57.2957795/                                MM   179
C                                                                        MM   180
      SUM = 0.0                                                          MM   181
      DO 120 I = 1,NP                                                    MM   182
      L = I-1                                                            MM   183
      APT = AZV(4*L+3)                                                   MM   184
      AZUS = AZV(4*L+1)                                                  MM   185
      IF (AZUS.GT.APT) AZUS = APT                                        MM   186
C                                                                        MM   187
C     COMPUTE SECTOR SIZE                                                MM   188
                                                                         MM   189
C                                                                        MM   190
      SDEG = 360./APT                                                    MM   191
C                                                                        MM   192
C     CALCULATE WIND AZIMUTH IN DEGREES CLOCKWISE FROM NORTH             MM   193
C                                                                        MM   194
      AZT(I) = 180.+AZUS*SDEG                                            MM   195
      IF (AZT(I).GT.360.) AZT(I) = AZT(I)-360.                           MM   196
      AX(I) = SIN(AZT(I)*RPD)                                            MM   197
      AY(I) = COS(AZT(I)*RPD)                                            MM   198
C                                                                        MM   199
C     CALCULATE R(I)                                                     MM   200
      IF (IRF.EQ.2) GO TO 100                                            MM   201
C                                                                        MM   202
C     INVERSE DISTANCE LAW                                               MM   203
C                                                                        MM   204
      R(I) = 1./DDD(I)                                                   MM   205
      GO TO 110                                                          MM   206
C                                                                        MM   207
C     INVERSE SQUARE DISTANCE LAW                                        MM   208
C                                                                        MM   209
100   CONTINUE                                                           MM   210
      R(I) = 1./(DDD(I)**2)                                              MM   211
110   CONTINUE                                                           MM   212
      SUM = SUM+R(I)                                                     MM   213
120   CONTINUE                                                           MM   214
C                                                                        MM   215
```

```
C     CALCULATE W(I) AND THE AZIMUTH AND VELOCITY OF INTEREST          MM   216
C                                                                      MM   217
      SUMX = 0.                                                        MM   218
      SUMY = 0.                                                        MM   219
      SUM2 = 0.0                                                       MM   220
      DO 130 I = 1,NP                                                  MM   221
      L = I-1                                                          MM   222
      W(I) = R(I)/SUM                                                  MM   223
      SUMX = SUMX+AX(I)*W(I)                                           MM   224
      SUMY = SUMY+AY(I)*W(I)                                           MM   225
      SUM2 = SUM2+AZV(4*L+2)*W(I)                                      MM   226
130   CONTINUE                                                         MM   227
      AZM = DPR*ATAN2(SUMX,SUMY)                                       MM   228
      IF (AZM.LT.0.) AZM = AZM+360.                                    MM   229
      VEL = SUM2                                                       MM   230
      RETURN                                                           MM   231
C                                                                      MM   232
      END                                                              MM   233



      SUBROUTINE BARIER(X,Y,XS,YS,OK)                                  MM   234
C                                                                      MM   235
C THIS ROUTINE CALCULATES WHAT SIDE OF A BARRIER A POINT IS ON         MM   236
C BARRIERS ARE FINITE LENGTH LINE SEGMENTS                             MM   237
C                                                                      MM   238
C BAS(1,I)      UTM-X COORDINATE OF FIRST END POINT                    MM   239
C BAS(2,I)      UTM-Y COORDINATE OF FIRST END POINT                    MM   240
C BAS(3,I)      UTM-X COORDINATE OF SECOND END POINT                   MM   241
C BAS(4,I)      UTM-Y COORDINATE OF SECOND END POINT                   MM   242
C NUMBAS        NUMBER OF BARRIERS (UP TO 20)                          MM   243
C IFIN          IF 0 FIRST TIME IN ROUTINE SETUP PT SLOPE LINES        MM   244
C SLPIN(1,I)    SLOPE OF LINE                                          MM   245
C SLPIN(2,I)    INTERCEPT OF LINE                                      MM   246
C X             X COORDINATE OF POINT OF INTEREST                      MM   247
C Y             Y COORDINATE OF POINT OF INTEREST                      MM   248
C XS            X COORDINATE OF REFERENCE POINT                        MM   249
C YS            Y COORDINATE OF REFERENCE POINT                        MM   250
C OK            FLAG IF POS (X-Y) IS ON SAME SIDE AS (XS-YS)           MM   251
                                                                       MM   252
C                                                                      MM   253
      DIMENSION BAS(4,20),SLPIN(2,20)                                  MM   254
      DIMENSION A(3), B(3), C(3)                                       MM   255
      COMMON  /ORIGIN/  UTMXOR, UTMYOR                                 MM   256
      DATA IFIN/0/                                                     MM   257
C                                                                      MM   258
C     ---------------------- NOTE ------------------------------------MM   259
C         THE USER MUST SET THE NUMBER OF BARRIERS AND THE  BARRIER    MM   260
C         END POINT COORDINATES (UTM) IN DATA STATEMENTS BELOW.        MM   261
C                                                                      MM   262
C         THERE ARE NO SIGNIFICANT GEOGRAPHIC BARRIERS TO THE WIND     MM   263
C         FIELD IN ST. LOUIS.  NUMBAS AND BAS HAVE BEEN SET TO ZERO.   MM   264
C                                                                      MM   265
      DATA NUMBAS/0/                                                   MM   266
      DATA BAS/80*0./                                                  MM   267
```

```
C                                                                       MM   268
      OK = 1.0                                                          MM   269
      IF (NUMBAS.LE.0) GO TO 140                                        MM   270
      IF (IFIN.GT.0) GO TO 110                                          MM   271
C                                                                       MM   272
C SETUP SLOPE INTERCEPT FORM OF A LINE                                  MM   273
      DO 100 I = 1,NUMBAS                                               MM   274
      DY = BAS(2,I)-BAS(4,I)                                            MM   275
      DX = BAS(1,I)-BAS(3,I)                                            MM   276
      SLPIN(1,I) = DY/DX                                                MM   277
      SLPIN(2,I) = BAS(2,I)-SLPIN(1,I)*BAS(1,I)                         MM   278
100   CONTINUE                                                          MM   279
      IFIN = 1                                                          MM   280
C                                                                       MM   281
110   CONTINUE                                                          MM   282
C                                                                       MM   283
      X = X + UTMXOR                                                    MM   284
      Y = Y + UTMYOR                                                    MM   285
      XS = XS + UTMXOR                                                  MM   286
      YS = YS + UTMYOR                                                  MM   287
C                                                                       MM   288
      C(1) = XS-X                                                       MM   289
      C(2) = YS-Y                                                       MM   290
      A(3) = 0.                                                         MM   291
      B(3) = 0.                                                         MM   292
      C(3) = 0.                                                         MM   293
      DO 120 I = 1,NUMBAS                                               MM   294
      D1 = YS-SLPIN(1,I)*XS-SLPIN(2,I)                                  MM   295
      D2 = Y-SLPIN(1,I)*X-SLPIN(2,I)                                    MM   296
      IF (D1*D2.GE.0.) GO TO 120                                        MM   297
      A(1) = BAS(1,I)-X                                                 MM   298
      A(2) = BAS(2,I)-Y                                                 MM   299
      B(1) = BAS(3,I)-X                                                 MM   300
      B(2) = BAS(4,I)-Y                                                 MM   301
      COSAB = UNIDOT(A,B)                                               MM   302
      COSAC = UNIDOT(A,C)                                               MM   303
      COSBC = UNIDOT(B,C)                                               MM   304
      IF (COSAC.LT.COSAB.OR.COSBC.LT.COSAB) GO TO 120                   MM   305
      OK = -1.0                                                         MM   306
      GO TO 130                                                         MM   307
120   CONTINUE                                                          MM   308
130   CONTINUE                                                          MM   309
C                                                                       MM   310
      X = X - UTMXOR                                                    MM   311
      Y = Y - UTMYOR                                                    MM   312
      XS = XS - UTMXOR                                                  MM   313
      YS = YS - UTMYOR                                                  MM   314
                                                                        MM   315
C                                                                       MM   316
140   CONTINUE                                                          MM   317
      RETURN                                                            MM   318
      END                                                               MM   319
```

```
      SUBROUTINE CELAVG(DH,NMESHP,CELVAT,AKZ,NZHI,STORE)                MM   320
C                                                                       MM   321
C     *CELAVG* CALCULATES THE AVERAGE DIFFUSIVITY COEFFICIENTS          MM   322
C              BETWEEN VERTICAL MESH POINT ELEVATIONS FOR USE           MM   323
C              IN THE CHEMICAL-DIFFUSION MODULE.                        MM   324
C                                                                       MM   325
      DIMENSION CELVAT(1),AKZ(1),STORE(6)                               MM   326
      NP1=NMESHP+1                                                      MM   327
      DO 20 K=2,NMESHP                                                  MM   328
      NTOP=IFIX(CELVAT(K)/DH+.5)                                        MM   329
      NBOT=IFIX(CELVAT(K-1)/DH+.5)+1                                    MM   330
      IF(K.EQ.2) NBOT=2                                                 MM   331
      HTOP=DH*(NTOP-1)                                                  MM   332
      HBOT=DH*(NBOT-1)                                                  MM   333
      SUM=0.                                                            MM   334
      KSUM=0                                                            MM   335
      IF(NTOP.GE.NZHI) NTOP=NZHI                                        MM   336
      IF(NBOT.GE.NZHI) NBOT=NZHI-1                                      MM   337
C                                                                       MM   338
      DO 15 J=NBOT,NTOP                                                 MM   339
      SUM=SUM+AKZ(J)                                                    MM   340
      KSUM=KSUM+1                                                       MM   341
   15 CONTINUE                                                          MM   342
      AVGKZ=SUM/KSUM                                                    MM   343
      STORE(K)=AVGKZ                                                    MM   344
   20 CONTINUE                                                          MM   345
      STORE(1) = STORE(2)                                               MM   346
      STORE(NP1) = STORE(NMESHP)                                        MM   347
      RETURN                                                            MM   348
      END                                                               MM   349



      FUNCTION CLOCKT(TSTART,DT)                                        MM   350
C                                                                       MM   351
C         CLOCKT RETURNS MILITARY CLOCK TIME AT TSTART + DT             MM   352
C              TSTART = INITIAL CLOCK TIME                              MM   353
C                  DT = ELAPSED TIME IN MINUTES                         MM   354
C                                                                       MM   355
      XMIN = AMOD(TSTART,100.)                                          MM   356
      HRS = TSTART-XMIN                                                 MM   357
      XMIN = XMIN+DT                                                    MM   358
      RMIN = AMOD(XMIN,60.)                                             MM   359
      HRS = HRS+100.*(XMIN-RMIN)/60.                                    MM   360
      CLOCKT = HRS+RMIN                                                 MM   361
      IF (CLOCKT.GT.2400.) CLOCKT = AMOD(CLOCKT,2400.)                  MM   362
      RETURN                                                            MM   363
      END                                                               MM   364
```

```
      SUBROUTINE CROSIT                                                 MM   365
C                                                                       MM   366
C         CROSIT GENERATES TRAJECTORY GRID SQUARE ID HISTORY            MM   367
C                                                                       MM   368
      DIMENSION ICHAR(26),  TX(4), KSYM(50), IDVG(501)                  MM   369
      COMMON /CNTROL/ KSTOP, TSUN                                       MM   370
      COMMON /REUSE/  KPSTAT, KPWDAT, CONVRT, KXTRA, KPWIND             MM   371
      COMMON /CROSS/ TC(500), PC(2,500), IDC(2,500), NC                 MM   372
      COMMON /GRID/ X1,X2,Y1,Y2,NX,NY,DELX,DELY,DELT                    MM   373
      COMMON /INPUTS/  TITLE(20),  JDATE(10),  NCURV                    MM   374
      COMMON /ORIGIN/  UTMXOR,  UTMYOR                                  MM   375
      COMMON /TRAJ/ TSTART, P(2,100), ID(2,100)                         MM   376
      COMMON /WIND/ T(100), V(100), TH(100), NPTS                       MM   377
      COMMON /VGRID/  ITGRID,   IVG(100,100)                            MM   378
                                                                        MM   379
C                                                                       MM   380
      DATA    ICHAR/1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1HK,1HL,1HM,MM   381
     1            1HN,1HO,1HP,1HQ,1HR,1HS,1HT,1HU,1HV,1HW,1HX,1HY,1HZ/MM   382
      DATA NCMAX/500/, JTMAX/500/                                       MM   383
      DATA SRD/57.29577951/                                             MM   384
      DATA NONO/4HNO  /                                                 MM   385
      DATA KYES /4HYES /                                                MM   386
      DATA LOUT, LPUNCH, IDFALT /6,1,0/                                 MM   387
C                                                                       MM   388
      KSTOP = 0                                                         MM   389
      CALL SETTUP                                                       MM   390
      IF (NPTS.GE.2) GO TO 100                                          MM   391
      KSTOP = 1                                                         MM   392
      WRITE(LOUT,5)                                                     MM   393
      RETURN                                                            MM   394
100   CONTINUE                                                          MM   395
C                                                                       MM   396
C         TABULATE ALL TIMES ASSOCIATED WITH GRID LINE CROSSINGS        MM   397
C                                                                       MM   398
      NC = 0                                                            MM   399
      KSYM(1) = ICHAR(1)                                                MM   400
      CALL PLTPNT(P(1,1),P(2,1),KSYM(1))                                MM   401
      DO 250 K = 2,NPTS                                                 MM   402
      L = K-1                                                           MM   403
      DELT = T(K)-T(L)                                                  MM   404
      VX = V(L)*COS(TH(L))                                              MM   405
      VY = V(L)*SIN(TH(L))                                              MM   406
      CALL PLACIT(DELT,L,VX,VY,P(1,K),ID(1,K))                          MM   407
      TSAVE = 0.                                                        MM   408
      TXMAX = 9999.                                                     MM   409
      IF (VX.NE.0.) TXMAX = ABS(DELX/VX)                                MM   410
      TYMAX = 9999.                                                     MM   411
      IF (VY.NE.0.) TYMAX = ABS(DELY/VY)                                MM   412
      RX = P(1,L)-(ID(1,L)-1)*DELX                                      MM   413
      RY = P(2,L)-(ID(2,L)-1)*DELY                                      MM   414
110   IF (VX) 120,130,140                                               MM   415
120   TX(1) = -RX/VX                                                    MM   416
      GO TO 150                                                         MM   417
130   TX(1) = 9999.                                                     MM   418
      GO TO 150                                                         MM   419
```

```
140   TX(1) = (DELX-RX)/VX                                              MM  420
150   CONTINUE                                                          MM  421
      IF (VY) 160,170,180                                               MM  422
160   TY = -RY/VY                                                       MM  423
      GO TO 190                                                         MM  424
170   TY = 9999.                                                        MM  425
      GO TO 190                                                         MM  426
180   TY = (DELY-RY)/VY                                                 MM  427
190   CONTINUE                                                          MM  428
      TS = AMIN1(TX(1),TY) + TSAVE                                      MM  429
      TB = AMAX1(TX(1),TY) + TSAVE                                      MM  430
      IF (TS.GT.DELT) GO TO 240                                         MM  431
      TNXT = TS                                                         MM  432
      TSNX = AMIN1(TXMAX,TYMAX)                                         MM  433
200   CONTINUE                                                          MM  434
      NC = NC+1                                                         MM  435
      TC(NC) = T(L)+TNXT                                                MM  436
      CALL PLACIT(TNXT,L,VX,VY,PC(1,NC),IDC(1,NC))                      MM  437
      IF (NC.EQ.NCMAX) GO TO 260                                        MM  438
      TSAVE = TNXT                                                      MM  439
      TNXT = TNXT+TSNX                                                  MM  440
      IF (TNXT-DELT) 210,210,220                                        MM  441
                                                                        MM  442
210   IF (TNXT-TB) 200,200,230                                          MM  443
220   IF (DELT-TB) 240,230,230                                          MM  444
230   CONTINUE                                                          MM  445
      NC = NC+1                                                         MM  446
      TC(NC) = T(L)+TB                                                  MM  447
      CALL PLACIT(TB,L,VX,VY,PC(1,NC),IDC(1,NC))                        MM  448
      IF (NC.EQ.NCMAX) GO TO 260                                        MM  449
      TSAVE = TB                                                        MM  450
      RX = PC(1,NC)-(IDC(1,NC)-1)*DELX                                  MM  451
      RY = PC(2,NC)-(IDC(2,NC)-1)*DELY                                  MM  452
      GO TO 110                                                         MM  453
240   CONTINUE                                                          MM  454
      KK = K                                                            MM  455
      IF (K.GT.26) KK = MOD(K,26)                                       MM  456
      IF (K.EQ.52) KK = 26                                              MM  457
      KSYM(K) = ICHAR(KK)                                               MM  458
      CALL PLTPNT(P(1,K),P(2,K),KSYM(K))                                MM  459
250   CONTINUE                                                          MM  460
      GO TO 270                                                         MM  461
260   KSTOP = 2                                                         MM  462
      NPTS = K                                                          MM  463
270   CONTINUE                                                          MM  464
      K = NPTS                                                          MM  465
C                                                                       MM  466
C         PRINT GRID LOCATIONS AND WIND VECTORS AT TRAJECTORY NODES     MM  467
C                                                                       MM  468
      STRT = TIMIN(TSTART)                                              MM  469
      CALL NEWPAG(TITLE,0,JDATE)                                        MM  470
      WRITE(LOUT,1)                                                     MM  471
      DO 290 L = 1,K                                                    MM  472
      IF (MOD(L,25).NE.0) GO TO 280                                     MM  473
      CALL NEWPAG(TITLE,0,JDATE)                                        MM  474
```

```
      WRITE(LOUT,1)                                                     MM   475
280   CONTINUE                                                          MM   476
      VV = V(L)*60.                                                     MM   477
      ANG = TH(L)*SRD                                                   MM   478
      T(L) = T(L) + STRT                                                MM   479
      WRITE(LOUT,2) KSYM(L),T(L),ID(1,L),ID(2,L),P(1,L),P(2,L),VV,ANG   MM   480
290   CONTINUE                                                          MM   481
C                                                                       MM   482
      CALL PLTOUT(LOUT)                                                 MM   483
C                                                                       MM   484
      CALL NEWPAG(TITLE,0,JDATE)                                        MM   485
      WRITE(LOUT,3)                                                     MM   486
      P1 = P(1,1) + UTMXOR                                              MM   487
      P2 = P(2,1) + UTMYOR                                              MM   488
      WRITE(LOUT,4)  STRT, P1, P2, ID(1,1), ID(2,1)                     MM   489
      IF(ITGRID.NE.KYES) WRITE(LPUNCH,9)  STRT,P1,P2,ID(1,1),ID(2,1)    MM   490
      DO 310 L = 1,NC                                                   MM   491
      IF (MOD(L,25).NE.0) GO TO 300                                     MM   492
      CALL NEWPAG(TITLE,0,JDATE)                                        MM   493
      WRITE(LOUT,3)                                                     MM   494
300   CONTINUE                                                          MM   495
      TC(L) = TC(L) + STRT                                              MM   496
      PC(1,L) = PC(1,L) + UTMXOR                                        MM   497
      PC(2,L) = PC(2,L) + UTMYOR                                        MM   498
      WRITE(LOUT,4) TC(L),PC(1,L),PC(2,L),IDC(1,L),IDC(2,L)             MM   499
      IF(ITGRID.EQ.KYES)  GO TO 310                                     MM   500
      IF(KPWIND.EQ.KYES)                                                MM   501
     1  WRITE (LPUNCH,9) TC(L),PC(1,L),PC(2,L),IDC(1,L),IDC(2,L)        MM   502
310   CONTINUE                                                          MM   503
      A = -10.                                                          MM   504
                                                                        MM   505
      IF(ITGRID.NE.KYES) WRITE (7,9) A                                  MM   506
C                                                                       MM   507
      IF (KSTOP.EQ.2) WRITE(LOUT,6)                                     MM   508
      KSTOP = 0                                                         MM   509
C                                                                       MM   510
C          GENERATE GRID SQUARE ID HISTORY FOR VARIABLE SIZE GRID       MM   511
C                                                                       MM   512
      IF(ITGRID.NE.KYES)  GO TO 360                                     MM   513
 320  CONTINUE                                                          MM   514
      I = IO(1,1)                                                       MM   515
      J = IO(2,1)                                                       MM   516
      IF(I.GE.1  .AND. I.LE.NX .AND. J.GE.1 .AND. J.LE.NY)  GO TO 322   MM   517
      IDVG(1) = IDFALT                                                  MM   518
      GO TO 323                                                         MM   519
322   IDVG(1) = IVG(I,J)                                                MM   520
323   IOK = KYES                                                        MM   521
      DO 330 K=1,NC                                                     MM   522
      I = IOC(1,K)                                                      MM   523
      J = IOC(2,K)                                                      MM   524
      IOK = KYES                                                        MM   525
      IF(I.LT.1)  IOK = NONO                                            MM   526
      IF(I.GT.NX) IOK = NONO                                            MM   527
      IF(J.LT.1)  IOK = NONO                                            MM   528
      IF(J.GT.NY) IOK = NONO                                            MM   529
```

```
      IF(IOK.EQ.KYES)  GO TO 325                                          MM   530
      IDVG(K+1) = IDFALT                                                  MM   531
      GO TO 330                                                           MM   532
  325 CONTINUE                                                            MM   533
      IDVG(K+1) = IVG(I,J)                                                MM   534
  330 CONTINUE                                                            MM   535
      CALL NEWPAG(TITLE,0,JDATE)                                          MM   536
      WRITE(LOUT,12)                                                      MM   537
      WRITE(LOUT,13) STRT,P1,P2,IDVG(1)                                   MM   538
      IF(KPWIND.EQ.KYES)  WRITE (LPUNCH,10)  TITLE                        MM   539
      IF(KPWIND.EQ.KYES)  WRITE(LPUNCH,14)   STRT, P1, P2, IDVG(1)        MM   540
      N = 1                                                               MM   541
      NLST = N                                                            MM   542
      DO 350 K = 1,NC                                                     MM   543
      IF(MOD(N,25).NE.0)  GO TO 340                                       MM   544
      IF(N.EQ.NLST)  GO TO 340                                            MM   545
      CALL NEWPAG(TITLE,0,JDATE)                                          MM   546
      WRITE(LOUT,12)                                                      MM   547
      NLST = N                                                            MM   548
  340 CONTINUE                                                            MM   549
      IF(IDVG(K+1) .EQ. IDVG(K))  GO TO 350                               MM   550
      N = N + 1                                                           MM   551
      WRITE(LOUT,13)  TC(K), PC(1,K), PC(2,K), IDVG(K+1)                  MM   552
      IF(KPWIND.EQ.KYES) WRITE(LPUNCH,14)TC(K),PC(1,K),PC(2,K),IDVG(K+1)MM   553
  350 CONTINUE                                                            MM   554
      IF(KPWIND.EQ.KYES)  WRITE(LPUNCH,14) A                              MM   555
  360 CONTINUE                                                            MM   556
      RETURN                                                              MM   557
C                                                                         MM   558
C                                                                         MM   559
1     FORMAT (1H0,15HTRAJECTORY DATA,/                                    MM   560
     1  1H0,48H  SYMBOL  T(MIN)      I      J      X(KM)      Y(KM),      MM   561
     2  3X,17HV(KM/HR)     THETA  )                                       MM   562
2     FORMAT (/6X,A1,F10.2,2I6,4F10.2)                                    MM   563
3     FORMAT (1H0,14X,3HUTM,7X,3HUTM,/,                                   MM   564
     1            3X,7HTC(MIN),4X,6HXC(KM),4X,6HYC(KM),4X,2HIC,4X,2HJC ) MM   565
4     FORMAT (/3F10.2,2I6)                                                MM   566
5     FORMAT (1H0,63HSUBROUTINE CROSIT HAS RECEIVED LESS THAN  2  TRAJECMM   567
                                                                          MM   568
     1TORY POINTS. / 17H CASE TERMINATED. //)                             MM   569
6     FORMAT (1H0,70HSUBROUTINE CROSIT TERMINATED TRAJECTORY EARLY TO AVMM   570
     1OID ARRAY OVERFLOW. //)                                             MM   571
  9   FORMAT (10X,3F10.2,2I10)                                            MM   572
  10  FORMAT (20A4)                                                       MM   573
  12  FORMAT (1H0,2X,10HTIME (MIN) ,5X,20H   UTM-X      UTM-Y    ,       MM   574
     1  5X,15HGRID SQUARE ID    )                                        MM   575
  13  FORMAT (1H0,3X,F7.2,6X,2F10.2,5X,I10)                               MM   576
  14  FORMAT (10X,3F10.2, I10)                                            MM   577
      END                                                                 MM   578
```

CORRECTIONS

1) Before MM 554 insert

```
ITLAST = IDVG (k+1)
TTLAST = TC (K)
```

2) After MM 554 insert

```
IF(TTLAST.EQ.T(NPTS) GO TO 355
XX = P(1,NPTS) + UTMXOR
YY= P(2.NPTS) + UTMYOR
TI= T(NPTS)
WRITE(LOUT,13) TI,XX,YY,ITLAST
IF(KPWIND.EQ.KYES) WRITE(LPUNCH,14) TI,XX,YY,ITLAST
355 CONTINUE
```

```
      SUBROUTINE DATE(IOLD,MOVE,IWAY,NEW)                                MM   579
C                                                                        MM   580
C     THIS SUBROUTINE CHANGES THE TIME AND DATE                          MM   581
C                                                                        MM   582
      DIMENSION IOLD(2),NEW(2)                                           MM   583
C                                                                        MM   584
C     IOLD(1)  OLD TIME 24 HOUR CLOCK                                    MM   585
C     IOLD(2)  OLD DATE YEAR-MONTH-DAY                                   MM   586
C     MOVE     TIME INCREMENT                                            MM   587
C     IWAY     POSITIVE OR NEGATIVE APPLICATION OF THE INCREMENT         MM   588
C     NEW(1)   NEW TIME 24 HOUR CLOCK                                    MM   589
C     NEW(2)   NEW DATE YEAR-MONTH-DAY                                   MM   590
C                                                                        MM   591
      I1 = IOLD(1)/100                                                   MM   592
      I2 = IOLD(1)-100*I1                                                MM   593
      J1 = IOLD(2)/10000                                                 MM   594
      J2 = (IOLD(2)-10000*J1)/100                                        MM   595
      J3 = IOLD(2)-10000*J1-100*J2                                       MM   596
      NEW(2) = IOLD(2)                                                   MM   597
      K22 = I1                                                           MM   598
      IF (IWAY.LT.0) GO TO 150                                           MM   599
C                                                                        MM   600
C     INCREMENT THE TIME                                                 MM   601
      K2 = I2+MOVE                                                       MM   602
      IF (K2.LT.60) GO TO 140                                            MM   603
      K2 = K2-60                                                         MM   604
      K22 = I1+1                                                         MM   605
      IF (K22.LE.24) GO TO 140                                           MM   606
      K22 = K22-24                                                       MM   607
      NEW(1) = 100*K22+K2                                                MM   608
      K22 = J2                                                           MM   609
      K2 = J3+1                                                          MM   610
      IF (J2.EQ.2) GO TO 120                                             MM   611
      IF (J2.EQ.4) GO TO 130                                             MM   612
      IF (J2.EQ.6) GO TO 130                                             MM   613
      IF (J2.EQ.9) GO TO 130                                             MM   614
      IF (J2.EQ.11) GO TO 130                                            MM   615
C     31 DAY MONTHS                                                      MM   616
      IF (K2.LE.31) GO TO 110                                            MM   617
      K2 = K2-31                                                         MM   618
100   CONTINUE                                                           MM   619
      K22 = J2+1                                                         MM   620
      IF (K22.LE.12) GO TO 110                                           MM   621
      K22 = K22-12                                                       MM   622
      J1 = J1+1                                                          MM   623
110   CONTINUE                                                           MM   624
      NEW(2) = 10000*J1+100*K22+K2                                       MM   625
      RETURN                                                             MM   626
120   CONTINUE                                                           MM   627
      IF (K2.LE.28) GO TO 110                                            MM   628
      K2 = K2-28                                                         MM   629
      GO TO 100                                                          MM   630
                                                                         MM   631
130   CONTINUE                                                           MM   632
      IF (K2.LE.30) GO TO 110                                            MM   633
```

```
      K2 = K2+30                                                        MM   634
      GO TO 100                                                         MM   635
140   CONTINUE                                                          MM   636
      NEW(1) = 100*K22+K2                                               MM   637
      RETURN                                                            MM   638
C                                                                       MM   639
C     DECREMENT TIME                                                    MM   640
150   CONTINUE                                                          MM   641
      K2 = I2-MOVE                                                      MM   642
      IF (K2.GE.0) GO TO 140                                            MM   643
      K2 = K2+60                                                        MM   644
      K22 = I1-1                                                        MM   645
      IF (K22.GE.0) GO TO 140                                           MM   646
      K22 = K22+24                                                      MM   647
      NEW(1) = 100*K22+K2                                               MM   648
      K22 = J2                                                          MM   649
      K2 = J3-1                                                         MM   650
      IF (K2.GT.0) GO TO 110                                            MM   651
      K22 = J2-1                                                        MM   652
      IF (J2.EQ.12) GO TO 170                                           MM   653
      IF (J2.EQ.10) GO TO 170                                           MM   654
      IF (J2.EQ.7) GO TO 170                                            MM   655
      IF (J2.EQ.5) GO TO 170                                            MM   656
      IF (J2.EQ.3) GO TO 180                                            MM   657
      K2 = 31                                                           MM   658
160   CONTINUE                                                          MM   659
      IF (K22.GT.0) GO TO 110                                           MM   660
      K22 = 12                                                          MM   661
      J1 = J1-1                                                         MM   662
      GO TO 110                                                         MM   663
170   CONTINUE                                                          MM   664
      K2 = 30                                                           MM   665
      GO TO 160                                                         MM   666
180   CONTINUE                                                          MM   667
      K2 = 28                                                           MM   668
      GO TO 160                                                         MM   669
C                                                                       MM   670
      END                                                               MM   671
```

```
C                                                                       MM   672
      SUBROUTINE DIFFUS(STAB,ZHT,AKZ,USTAR)                             MM   673
C                                                                       MM   674
C-----THIS SUBROUTINE CALCULATES THE VALUES OF KZ IN THE SURFACE        MM   675
C     LAYER USING THE FORMULATIONS OF BUSINGER,ET AL(1971).             MM   676
C                                                                       MM   677
      AKN=.35*USTAR*ZHT                                                 MM   678
   66 IF(STAB) 67,68,69                                                 MM   679
   67 AKZ=AKN*((1.-15.*ZHT/STAB)**0.25)                                 MM   680
      GO TO 70                                                          MM   681
   69 AKZ=AKN/(1.+4.7*ZHT/STAB)                                         MM   682
      GO TO 70                                                          MM   683
   68 CALL EXIT                                                         MM   684
   70 RETURN                                                            MM   685
```

```
      END                                                              MM   686


      FUNCTION DOTTY(A,B)                                              MM   687
C                                                                      MM   688
C RETURNS THE DOT PRODUCT OF VECTOR A AND VECTOR B                     MM   689
C                                                                      MM   690
      DIMENSION A(3),B(3)                                              MM   691
C                                                                      MM   692
      DOTTY = 0.0                                                      MM   693
                                                                       MM   694
      DO 100 I = 1,3                                                   MM   695
      DOTTY = A(I)*B(I)+DOTTY                                          MM   696
100   CONTINUE                                                         MM   697
      RETURN                                                           MM   698
      END                                                              MM   699


      FUNCTION DTIME(A)                                                MM   700
C-----THIS FUNCTION CONVERTS TIME ON FROM HOURS AND MINUTES            MM   701
C     TO HOURS AND DECIMAL FRACTIONS OF AN HOUR.  FOR EXAMPLE,         MM   702
C     1130 IS CONVERTED TO 11.5.                                       MM   703
C                                                                      MM   704
      TIMEX=A*0.01                                                     MM   705
      DTIME=(TIMEX-AINT(TIMEX))/0.6+AINT(TIMEX)                        MM   706
      RETURN                                                           MM   707
      END                                                              MM   708


      SUBROUTINE EDDY  (KPWIND)                                        MM   709
C                                                                      MM   710
C     THIS SUBROUTINE IS THE DRIVER FOR THE VERTICAL EDDY DIFFUSIVITY  MM   711
C     SUB-MODULE,                                                      MM   712
C     D. GODDEN  1.25.78                                               MM   713
C                                                                      MM   714
      DIMENSION   STORE(6), ZTIME(15),TEMP(201),TTMP(201),IKLAS(15)    MM   715
      COMMON /INPUTS/ TITLE(20),        JDATE(10),          NCURV      MM   716
      COMMON /KZINPT/ NEEDKZ,           KZPRTX,             NUMSD,     MM   717
     1                TIMESD(3),        TEMPSD(50,3),       ZELVSD(50,3)MM   718
     2    ,           TEMPSF(24),       NPTSD(3),           NKZDAT     MM   719
      COMMON/CELECT/CELHTS(5,1),NOHTS,NMES                             MM   720
      COMMON/TEMPHT/Z(50),ZZ(50),T(50),TT(50)                          MM   721
      COMMON/DIFDAT/DELTAT(30),STAB(30),ZMIX(30),USTAR(30),            MM   722
     +              UZZ(201),AKZ(201)                                  MM   723
      COMMON/WIND/TIME(100),V(100),TH(100),NPTS                        MM   724
      COMMON/ORIGIN/UTMXOR,UTMYOR                                      MM   725
      COMMON/TRAJ/TSTART,P(2,100),ID(2,100)                            MM   726
      COMMON/PEVE/PE(2,24),VE(24)                                      MM   727
      COMMON/CORIOL/F                                                  MM   728
      COMMON/WHERE/STALAT,RLONG,TMZONE                                 MM   729
      DATA IYES/4HYES /                                                MM   730
      DATA LOUT,LPUNCH  /6,1/                                          MM   731
```

```
      DATA ADVECT/0./                                                   MM   732
      DATA ZW/30./                                                      MM   733
      SUPER = -0.98                                                     MM   734
      ZSUP = 0.0                                                        MM   735
      SUP = 0.0                                                         MM   736
      CALL KZDATA(IERR)                                                 MM   737
      IF(IERR.LT.0) GO TO 2000                                          MM   738
      IOBUG = IERR                                                      MM   739
      IKKL = 1                                                          MM   740
      CALL UXYPOS                                                       MM   741
      DO 1000 ISD=1,NUMSD                                               MM   742
      ISDP1 = ISD + 1                                                   MM   743
      IF(ISD.EQ.NUMSD) GO TO 10                                         MM   744
      SUP=(TEMPSD(2,ISDP1)-TEMPSD(1,ISDP1))/(ZELVSD(2,ISDP1)-ZELVSD(1,  MM   745
     1ISDP1))*100.                                                      MM   746
      IF(SUP.GE.SUPER) GO TO 10                                         MM   747
      SUPER=SUP                                                         MM   748
      ZSUP=ZELVSD(2,ISDP1)-ZELVSD(1,ISDP1)                              MM   749
   10 IT=1                                                              MM   750
      ZTIME(IT)=TIMESD(ISD)                                             MM   751
      IF(STALAT.EQ.0.) STALAT=35.                                       MM   752
      RADLAT=STALAT*0.017453                                            MM   753
      F=1.4584E-04*SIN(RADLAT)                                          MM   754
      DH=0.                                                             MM   755
      AKMIN=6.                                                          MM   756
                                                                        MM   757
      ZTIME(IT)=DTIME(ZTIME(IT))                                        MM   758
      IN=IFIX(ZTIME(IT))+1                                              MM   759
      IF(ZW.EQ.0.) ZW=10.                                               MM   760
    2 CALL XMIT(NPTSD(ISD),ZELVSD(1,ISD),Z(1))                          MM   761
      CALL XMIT(NPTSD(ISD),TEMPSD(1,ISD),T(1))                          MM   762
      LMAX=NPTSD(ISD)                                                   MM   763
      TEMP(1)=T(1)                                                      MM   764
      T1ST=TEMP(1)                                                      MM   765
      UZZ(1)=0.                                                         MM   766
      TMAX=TEMP(1)                                                      MM   767
      ZMAX=0.                                                           MM   768
      STAELV=Z(1)                                                       MM   769
      DO 5 I=1,LMAX                                                     MM   770
      Z(I)=Z(I)-STAELV                                                  MM   771
    5 CONTINUE                                                          MM   772
      LMAXM1=LMAX-1                                                     MM   773
    6 CALL XMIT(LMAXM1,Z(2),ZZ(1))                                      MM   774
      CALL XMIT(LMAXM1,T(2),TT(1))                                      MM   775
      ZHIGH=ZZ(LMAXM1)                                                  MM   776
      TLAPS=(TT(1)-T(1))*100./ZZ(1)                                     MM   777
   20 DH=DH+10.                                                         MM   778
      NMAX=IFIX(ZHIGH/DH+0.5)+1                                         MM   779
      IF(NMAX.GT.201) GO TO 20                                          MM   780
      DH01=DH*0.01                                                      MM   781
      L=1                                                               MM   782
      M=1                                                               MM   783
    9 IF(L.EQ.LMAX) GO TO 24                                            MM   784
      ZZ(M)=AINT(ZZ(L)/DH+0.5)*DH                                       MM   785
      IF(ZZ(M).EQ.Z(M)) GO TO 16                                        MM   786
```

```
      Z(M+1)=ZZ(M)                                                    MM   787
      T(M+1)=TT(L)                                                    MM   788
      L=L+1                                                           MM   789
      M=M+1                                                           MM   790
      GO TO 9                                                         MM   791
   16 L=L+1                                                           MM   792
      GO TO 9                                                         MM   793
   24 LMAX=M                                                          MM   794
      LMAXM1=LMAX-1                                                   MM   795
      CALL XMIT(LMAXM1,Z(2),ZZ(1))                                    MM   796
      CALL XMIT(LMAXM1,T(2),TT(1))                                    MM   797
      IF(ZSUP.EQ.0.) ZSUP=Z(2)                                        MM   798
      CALL SMOOTH(LMAX)                                               MM   799
      IF(LMAX.GT.31)  LMAX = 31                                       MM   800
      ZHIGH = Z(LMAX)                                                 MM   801
      IF(ZNITE.EQ.0.) ZNITE=Z(2)                                      MM   802
      ZNITE=AMAX1(100.,ZNITE)                                         MM   803
    7 DO 12 L=2,LMAX                                                  MM   804
      TPOT=TT(L-1)+.0098*ZZ(L-1)                                      MM   805
      IF(TPOT.LT.TMAX) GO TO 12                                       MM   806
      TMAX=TPOT                                                       MM   807
      ZMAX=Z(L)                                                       MM   808
   12 CONTINUE                                                        MM   809
      IF(TLAPS.GE.-.98.OR.ZMAX.GT.0..OR.LMAX.LE.2) GO TO 14           MM   810
      TMAX=T(2)                                                       MM   811
      ZMAX=Z(2)                                                       MM   812
      GO TO 7                                                         MM   813
   14 N=1                                                             MM   814
      AKZ(N)=AKMIN                                                    MM   815
      J=1                                                             MM   816
    1 DELTAT(J)=(TT(J)-T(J))*100./(ZZ(J)-Z(J))                        MM   817
  107 ZHT=Z(J)                                                        MM   818
  108 ZHT=ZHT+DH                                                      MM   819
                                                                      MM   820
      N=N+1                                                           MM   821
      TEMP(N)=TEMP(N-1)+DELTAT(J)*DH01                                MM   822
      IF(ZHT.GE.ZZ(J)) GO TO 8                                        MM   823
      GO TO 108                                                       MM   824
    8 IF(ZZ(J).GE.ZHIGH) GO TO 21                                     MM   825
      J=J+1                                                           MM   826
      GO TO 1                                                         MM   827
   21 NMAX=N                                                          MM   828
      IF(TLAPS.GT.-.98) GO TO 33                                      MM   829
      N=1                                                             MM   830
      TTMP(N)=TEMP(N)                                                 MM   831
      GO TO 330                                                       MM   832
C-----UPDATE SURFACE TEMPERATURE AND WIND SPEED                       MM   833
   33 N=1                                                             MM   834
      J=1                                                             MM   835
      ZHT=0.                                                          MM   836
      IT=IT+1                                                         MM   837
      TSTOP=AINT(DTIME(TIME(NPTS)))                                   MM   838
      IF(ISD.EQ.NUMSD) GO TO 30                                       MM   839
      TSTOP=AINT(DTIME(TIMESD(ISDP1)))                                MM   840
   30 ZTIME(IT)=(AINT(TIME(1)*0.01)+FLOAT(IT-2))                      MM   841
```

```
      IF(ISD.GT.1) ZTIME(IT)=AINT(TIMESD(ISD)*0.01)+FLOAT(IT-1)          MM   842
      IS=IFIX(ZTIME(IT))+1                                               MM   843
      TTMP(1)=TEMPSF(IS)-TEMPSF(IN)+T1ST                                 MM   844
      USFC=VE(IS)                                                        MM   845
      IF(USFC.EQ.0.) USFC=0.5                                            MM   846
      TIMEX=ZTIME(IT)-ZTIME(1)                                           MM   847
      TTMP(1)=TTMP(1)-(ADVECT*TIMEX)                                     MM   848
      CALL ROUGNES(Z0,IS)                                                MM   849
      IF(KZPRTX.NE.IYES) GO TO 102                                       MM   850
      CALL NEWPAG(TITLE,0,IDATE)                                         MM   851
      WRITE(LOUT,39) ZTIME(IT)                                           MM   852
      IF(IDBUG.EQ.IYES)  WRITE(LOUT,26)                                  MM   853
  102 AKZ(1)=AKMIN                                                       MM   854
      UZZ(1)=0.                                                          MM   855
      ODZ=0.                                                             MM   856
      IF(TTMP(1).LT.T(1)) GO TO 40                                       MM   857
      IF(TTMP(1).EQ.T(1)) GO TO 35                                       MM   858
C-----NEW SFC TEMP GREATER THAN LAST SFC TEMP                            MM   859
  330 DELT=-0.98                                                         MM   860
   32 N=N+1                                                              MM   861
      TTMP(N)=TTMP(N-1)+DELT*DH01                                        MM   862
      IF(TTMP(N).LT.TEMP(N)) GO TO 34                                    MM   863
      IF(N.EQ.NMAX) GO TO 37                                             MM   864
      GO TO 32                                                           MM   865
   34 IF(TTMP(1).LT.TEMP(1)) GO TO 471                                   MM   866
      NTRSCT=N-1                                                         MM   867
      GO TO 36                                                           MM   868
   37 NTRSCT=NMAX                                                        MM   869
   36 IF(TTMP(1).LE.TMAX.OR.NTRSCT.LT.NMAX) GO TO 230                    MM   870
      TMAX=TTMP(1)                                                       MM   871
      ZMAX=0.                                                            MM   872
  230 ZNVRSN=(NTRSCT-1)*DH                                               MM   873
      T(1)=TTMP(1)                                                       MM   874
      IF(ZNVRSN.EQ.0.) GO TO 249                                         MM   875
      IF(ZNVRSN-Z(2)) 231,249,231                                        MM   876
  231 DO 239 J=2,LMAX                                                    MM   877
      IF(J.EQ.2) GO TO 235                                               MM   878
  232 IF(ZZ(J-1).LE.ZNVRSN) GO TO 239                                    MM   879
      L=L+1                                                              MM   880
      Z(L)=ZZ(J-1)                                                       MM   881
      T(L)=TT(J-1)                                                       MM   882
                                                                         MM   883
      GO TO 239                                                          MM   884
  235 L=J                                                                MM   885
  236 Z(L)=ZNVRSN                                                        MM   886
      T(L)=TTMP(NTRSCT)                                                  MM   887
      IF(Z(L).LT.ZZ(1)) GO TO 232                                        MM   888
  239 CONTINUE                                                           MM   889
      LMAX=L                                                             MM   890
      DELT=TLAPS                                                         MM   891
      IF(SUPER.LT.DELT) DELT=SUPER                                       MM   892
  249 LMAXM1=LMAX-1                                                      MM   893
      DO 250 J=1,LMAXM1                                                  MM   894
      TT(J)=T(J+1)                                                       MM   895
      ZZ(J)=Z(J+1)                                                       MM   896
```

```
      DELTAT(J)=(TT(J)-T(J))*100./(ZZ(J)-Z(J))                          MM   897
  250 CONTINUE                                                          MM   898
      IF(DELTAT(1).GT.DELT) DELTAT(1)=DELT                              MM   899
      IF(IT-2) 33,22,23                                                 MM   900
   22 N1=NTRSCT+1                                                       MM   901
      NO=NMAX-NTRSCT                                                    MM   902
      CALL XMIT(NO,TEMP(N1),TTMP(N1))                                   MM   903
   23 CALL STADIF(ZW,ZO,USFC,ZHIGH,DH,AKMIN,ISKLAS)                     MM   904
      GO TO 29                                                          MM   905
C-----NEW SFC TEMP EQUAL TO LAST SFC TEMP                               MM   906
   35 IF(IT.GT.2) GO TO 38                                              MM   907
      CALL XMIT(NMAX,TEMP(1),TTMP(1))                                   MM   908
   38 CALL STADIF(ZW,ZO,USFC,ZHIGH,DH,AKMIN,ISKLAS)                     MM   909
      GO TO 29                                                          MM   910
C-----NEW SFC TEMP LESS THAN LAST SFC TEMP                              MM   911
   40 IF(DELTAT(1).LT.-0.98.AND.IT.GT.1) GO TO 470                      MM   912
      T(1)=TTMP(1)                                                      MM   913
      DELTAT(1)=(T(2)-T(1))*100./Z(2)                                   MM   914
      IF(DELTAT(1)+0.98) 45,45,44                                       MM   915
   44 IF(Z(2)-ZNITE) 440,45,450                                         MM   916
   45 N1=IFIX(ZZ(1)/DH)+1                                               MM   917
      DO 400 N=2,N1                                                     MM   918
      TTMP(N)=TTMP(N-1)+DELTAT(1)*DH01                                  MM   919
  400 CONTINUE                                                          MM   920
      IF(IT.GT.2) GO TO 48                                              MM   921
      N11=N1+1                                                          MM   922
      NO=NMAX-N1                                                        MM   923
      CALL XMIT(NO,TEMP(N11),TTMP(N11))                                 MM   924
   48 CALL STADIF(ZW,ZO,USFC,ZHIGH,DH,AKMIN,ISKLAS)                     MM   925
      GO TO 29                                                          MM   926
  440 DO 445 J=3,LMAX                                                   MM   927
      IF(Z(J)-ZNITE) 445,441,442                                        MM   928
  441 JSTART=2                                                          MM   929
      GO TO 443                                                         MM   930
  442 NITE=IFIX(ZNITE/DH)+1                                             MM   931
      T(2)=TEMP(NITE)                                                   MM   932
      Z(2)=ZNITE                                                        MM   933
      JSTART=3                                                          MM   934
  443 NO=LMAX-J+1                                                       MM   935
      CALL XMIT(NO,ZZ(J-1),Z(JSTART))                                   MM   936
      CALL XMIT(NO,TT(J-1),T(JSTART))                                   MM   937
      NO=J-JSTART                                                       MM   938
      JMAX=LMAX-NO+1                                                    MM   939
      CALL XMIT(-NO,0.,Z(JMAX))                                         MM   940
      CALL XMIT(-NO,0.,T(JMAX))                                         MM   941
      NO=LMAX-1                                                         MM   942
      CALL XMIT(NO,Z(2),ZZ(1))                                          MM   943
      CALL XMIT(NO,T(2),TT(1))                                          MM   944
      LMAX=JMAX-1                                                       MM   945
                                                                        MM   946
      GO TO 446                                                         MM   947
  445 CONTINUE                                                          MM   948
  446 JMAX=LMAX-1                                                       MM   949
      DO 449 I=1,JMAX                                                   MM   950
      DELTAT(I)=(TT(I)-T(I))*100./(ZZ(I)-Z(I))                          MM   951
```

```
  449 CONTINUE                                                          MM   952
      GO TO 45                                                          MM   953
  450 T(2)=T(2)+.0098*(Z(2)-ZNITE)                                      MM   954
      Z(2)=ZNITE                                                        MM   955
      DELTAT(1)=(T(2)-T(1))*100./(Z(2)-Z(1))                            MM   956
      DELT=DELTAT(2)                                                    MM   957
      DELTAT(2)=-.98                                                    MM   958
      DO 455 M=2,LMAX                                                   MM   959
      T(M+1)=TT(M+1)                                                    MM   960
      Z(M+1)=ZZ(M+1)                                                    MM   961
      STORE=DELTAT(M+1)                                                 MM   962
      DELTAT(M+1)=DELT                                                  MM   963
  455 DELT=STORE                                                        MM   964
      CALL XMIT(LMAX,T(2),TT(1))                                        MM   965
      CALL XMIT(LMAX,Z(2),ZZ(1))                                        MM   966
      LMAX=LMAX+1                                                       MM   967
      GO TO 45                                                          MM   968
  470 T2=T(1)+DELTAT(1)*ZSUP*0.01                                       MM   969
  471 DELTAT(1)=(T2-TTMP(1))*100./ZSUP                                  MM   970
      IF(DELTAT(1).GT.-0.98) GO TO 40                                   MM   971
      T(1)=TTMP(1)                                                      MM   972
      CALL STADIF(ZW,ZO,USFC,ZHIGH,DH,AKMIN,ISKLAS)                     MM   973
      GO TO 130                                                         MM   974
   29 CONTINUE                                                          MM   975
      CALL XMIT(NMAX,TTMP(1),TEMP(1))                                   MM   976
  130 CALL KEKLAY(LMAX,ZMAX,ZHIGH,AKMIN,DH)                             MM   977
  290 IF(KZPRTX.NE.IYES) GO TO 301                                      MM   978
      WRITE(LOUT,25) DH                                                 MM   979
      NOPTS=NMAX-1                                                      MM   980
      WRITE(LOUT,1260)                                                  MM   981
      NL = NOPTS/5                                                      MM   982
      NL2 = 2*NL                                                        MM   983
      NL3 = 3*NL                                                        MM   984
      NL4 = 4*NL                                                        MM   985
      NTRUNC = MOD(NOPTS,5)                                             MM   986
      DO 100 I=1,NL                                                     MM   987
      I2 = I+NL                                                         MM   988
      I3 = I+NL2                                                        MM   989
      I4 = I+NL3                                                        MM   990
      I5 = I+NL4                                                        MM   991
      ZI = DH*FLOAT(I)                                                  MM   992
      ZI2 = DH*FLOAT(I2)                                                MM   993
      ZI3 = DH*FLOAT(I3)                                                MM   994
      ZI4 = DH*FLOAT(I4)                                                MM   995
      ZI5 = DH*FLOAT(I5)                                                MM   996
      WRITE(LOUT,28) ZI,AKZ(I+1),ZI2,AKZ(I2+1),ZI3,AKZ(I3+1),          MM   997
     1                            ZI4,AKZ(I4+1),ZI5,AKZ(I5+1)           MM   998
100   CONTINUE                                                          MM   999
      IF(NTRUNC.EQ.0)  GO TO 95                                         MM  1000
      I5 = I5 + 1                                                       MM  1001
      DO 94 I =I5,NOPTS                                                 MM  1002
      ZI = DH*FLOAT(I5)                                                 MM  1003
      I5 = I5 + 1                                                       MM  1004
94    WRITE(LOUT,1250) ZI, AKZ(I+1)                                     MM  1005
95    CONTINUE                                                          MM  1006
```

```
      NOHP1 = NOHTS + 1                                                  MM  1007
      WRITE(LOUT,1190)                                                   MM  1008
      WRITE(LOUT,1210) (IG,IG=1,NOHP1)                                   MM  1009
      WRITE(LOUT,1220)                                                   MM  1010
  301 IF(IKKL.NE.1)  GO TO 302                                           MM  1011
      CALL NEWPAG(TITLE,0,IDATE)                                         MM  1012
      NOHP1 = NOHTS + 1                                                  MM  1013
      WRITE(LOUT,1190)                                                   MM  1014
      WRITE(LOUT,1200)    (CELHTS(IG,1),IG=1,NOHTS)                      MM  1015
      WRITE(LOUT,1210)    (IG,IG=1,NOHP1)                                MM  1016
      WRITE(LOUT,1220)                                                   MM  1017
302   ITPP = IFIX(ZTIME(IT) + 1)                                         MM  1018
      CALL CELAVG(DH,NOHTS,CELHTS(1,1),AKZ,NMAX,STORE)                   MM  1019
      TJJJ = ZTIME(IT)*100.                                              MM  1020
      WRITE(LOUT,1230) TJJJ,TEMPSF(ITPP),ISKLAS,(STORE(IG),IG=1,NOHP1)   MM  1021
      TJJJ = .60*TJJJ                                                    MM  1022
      IF(KPWIND.EQ.IYES)                                                 MM  1023
     1   WRITE(LPUNCH,1240) IKKL,TJJJ,(STORE(IG),IG=1,NOHP1)             MM  1024
      IKLAS(IKKL) = ISKLAS                                               MM  1025
      IKKL = IKKL + 1                                                    MM  1026
                                                                         MM  1027
      IF(ZTIME(IT).GE.TSTOP) GO TO 1000                                  MM  1028
      GO TO 33                                                           MM  1029
 1000 CONTINUE                                                           MM  1030
      CALL NEWPAG(TITLE,0,JDATE)                                         MM  1031
      IKKL = IKKL - 1                                                    MM  1032
      TMINXX = IFIX(TIME(1))*.60                                         MM  1033
      DO 1005 KK = 1,IKKL                                                MM  1034
      IF(KPWIND.EQ.IYES) WRITE(LPUNCH,1001) TMINXX, IKLAS(KK)            MM  1035
      TMINXX = TMINXX + 60.0                                             MM  1036
 1005 CONTINUE                                                           MM  1037
      RETURN                                                             MM  1038
 2000 WRITE(LOUT,2001)                                                   MM  1039
      WRITE(LOUT,2002) IERR                                              MM  1040
      STOP                                                               MM  1041
   25  FORMAT(1H0,33H VERTICAL EDDY DIFFUSIVITY EVERY ,F3.0,8H METERS  )  MM  1042
    26 FORMAT(1H0,3X,6HHEIGHT,6X,5HDT/DZ,7X,5HDU/DZ,7X,1HA,14X,1HL,8X,    MM  1043
     15HUSTAR,4X,6HMIX HT,/22H   (METERS)   (C/100M),5X,7H(1/SEC),18X,    MM  1044
     28H(METERS),3X,17H(M/SEC)  (METERS))                                MM  1045
   28  FORMAT(1H ,5X,5(F6.0,F9.1,5X))                                     MM  1046
    39 FORMAT(1H0,15X,4HTIME,2X,F4.0,6H00 LST)                           MM  1047
 1001 FORMAT(25H TIME AND STABILITY CLASS   ,15X,F10.1,I10)              MM  1048
 1190  FORMAT(1H0,7X,70HAVERAGE DIFFUSION COEFFICIENTS FOR THE CHEMICAL/DMM  1049
     1IFFUSION MODULE                /)                                  MM  1050
 1200  FORMAT(1H0,7X,22HMESH POINT ELEVATIONS ,15X,5F10.0,5X,6HMETERS/ )  MM  1051
 1210  FORMAT(1H0,7X,4HTIME,5X,7HSURFACE,5X,9HSTABILITY,8X,6(2HKZ,I1,7X))MM  1052
 1220  FORMAT(1H ,14X,11HTEMPERATURE,5X,5HCLASS,30X,11H(M**2/MIN)  )      MM  1053
 1230  FORMAT(1H0,7X,F4.0,6X,F5.1,9X,I2,5X,6F10.0)                        MM  1054
 1240  FORMAT(14H DIFFUSIVITIES  ,I2,4X,F10.1,10X,3F10.1,/,20X,3F10.1)    MM  1055
 1250  FORMAT(1H ,85X,F6.0,F9.1)                                          MM  1056
 1260  FORMAT(1H0,6X,5(6HHEIGHT,5X,2HKZ,7X)/8X5(15H(M)   (M**2/MIN),4X)/)MM  1057
 2001 FORMAT(1H0,69HERROR IN SUBROUTINE KZDATA WHICH READS IN DATA NEEDEMM  1058
     1D TO CALCULATE KZ)                                                 MM  1059
 2002 FORMAT(1H ,5HIERR=,I4)                                             MM  1060
      END                                                                MM  1061
```

```
      SUBROUTINE EDGE(X,Y,ITF,IEDGE)                                    MM  1062
C                                                                       MM  1063
C     *EDGE* CHECKS WHETHER NEW POINT IS OUTSIDE THE MODELING REGION     MM  1064
C                                                                       MM  1065
      COMMON /GRID/ X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT            MM  1066
      DATA IONE/1/                                                      MM  1067
      DATA BUFZON /15./                                                 MM  1068
C                                                                       MM  1069
      XW = X1 - BUFZON                                                  MM  1070
      XE = X2 + BUFZON                                                  MM  1071
      YS = Y1 - BUFZON                                                  MM  1072
      YN = Y2 + BUFZON                                                  MM  1073
C                                                                       MM  1074
      IF (X.LT.XW) GO TO 100                                            MM  1075
      IF (X.GT.XE) GO TO 100                                            MM  1076
      IF (Y.LT.YS) GO TO 100                                            MM  1077
      IF (Y.GT.YN) GO TO 100                                            MM  1078
C                                                                       MM  1079
C         POINT IS OK                                                   MM  1080
C                                                                       MM  1081
      IEDGE = 0                                                         MM  1082
      GO TO 110                                                         MM  1083
100   CONTINUE                                                          MM  1084
C                                                                       MM  1085
C         POINT IS OUTSIDE GRID + BUFFER ZONE                           MM  1086
C                                                                       MM  1087
      IEDGE = ISIGN(IONE,ITF)                                           MM  1088
110   RETURN                                                            MM  1089
                                                                        MM  1090
      END                                                               MM  1091
```

```
      SUBROUTINE EXTRP(IHOUR,ARY,VAL)                                   MM  1092
C                                                                       MM  1093
C      EXTRP SELECTS A VALUE OUT OF A 24-WORD ARRAY.  IF THE VALUE      MM  1094
C      SOUGHT IS MISSING (INDICATED BY A NEGATIVE ENTRY), EXTRP         MM  1095
C      WILL ATTEMPT TO FILL THE GAP BY INTERPOLATION OR EXTRAPOLATION.  MM  1096
C                                                                       MM  1097
C     IHOUR         CLOCK TIME ON A 24 HOUR BASIS                       MM  1098
C     ARY           ARRAY OF 24 VALUES  (WIND OR VELOCITY)              MM  1099
C     VAL           VALUE SELECTED                                      MM  1100
C                                                                       MM  1101
C     MISSING DATA SITUATIONS HANDLED BY EXTRP                          MM  1102
C                                                                       MM  1103
C     CASE  I-3  I-2  I-1   I   I+1  I+2  I+3                          MM  1104
C      0                X    -    X                                   MM  1105
C      1           X    X    -    -                                   MM  1106
C      2      X    -    X    -    -                                   MM  1107
C      3      X    X    -    -                                        MM  1108
C      4                -    -    X    X                              MM  1109
C      5                -    -    X    -    X                         MM  1110
C      6                     -    -    X    X                         MM  1111
C                                                                       MM  1112
      DIMENSION ARY(24)                                                 MM  1113
```

```
C                                                                       MM  1114
      IPICK = IHOUR/100+1                                               MM  1115
      IF (IPICK.GT.24) IPICK = IPICK-24                                 MM  1116
      VAL = ARY(IPICK)                                                  MM  1117
      IF (VAL.GE.0.) RETURN                                             MM  1118
C                                                                       MM  1119
C ENTRY MISSING. MUST TRY TO GET VALUE BY INTERPOLATION OR EXTRAPOLATIONMM  1120
C IS THERE A VALUE ON BOTH SIDES                                        MM  1121
      VAL = -1.                                                         MM  1122
      IF (IPICK.EQ.1) GO TO 100                                         MM  1123
      IF (IPICK.EQ.24) GO TO 130                                        MM  1124
      IF (ARY(IPICK-1).LT.0.) GO TO 160                                 MM  1125
      IF (ARY(IPICK+1).LT.0.) GO TO 190                                 MM  1126
C                                                                       MM  1127
C CASE 0                                                                MM  1128
C                                                                       MM  1129
      VAL = 0.5*(ARY(IPICK-1)+ARY(IPICK+1))                             MM  1130
      RETURN                                                            MM  1131
C                                                                       MM  1132
C WE ARE AT THE FIRST POINT                                             MM  1133
100   CONTINUE                                                          MM  1134
      IF (ARY(2).LT.0.) GO TO 110                                       MM  1135
      IF (ARY(3).LT.0.) GO TO 120                                       MM  1136
      VAL = ARY(2)-(ARY(3)-ARY(2))                                      MM  1137
      GO TO 220                                                         MM  1138
110   CONTINUE                                                          MM  1139
      IF (ARY(3).LT.0..OR.ARY(4).LT.0.) RETURN                          MM  1140
      VAL = ARY(3)-2.*(ARY(4)-ARY(3))                                   MM  1141
      GO TO 220                                                         MM  1142
120   CONTINUE                                                          MM  1143
      IF (ARY(4).LT.0.) RETURN                                          MM  1144
      ARY(3) = 0.5*(ARY(2)+ARY(4))                                      MM  1145
      GO TO 100                                                         MM  1146
C                                                                       MM  1147
C WE ARE AT THE LAST POINT                                              MM  1148
130   CONTINUE                                                          MM  1149
      IF (ARY(23).LT.0.) GO TO 140                                      MM  1150
      IF (ARY(22).LT.0.) GO TO 150                                      MM  1151
      VAL = ARY(23)+ARY(23)-ARY(22)                                     MM  1152
                                                                        MM  1153
      GO TO 220                                                         MM  1154
140   CONTINUE                                                          MM  1155
      IF (ARY(22).LT.0..OR.ARY(21).LT.0.) RETURN                        MM  1156
      VAL = ARY(22)+2.*(ARY(22)-ARY(21))                                MM  1157
      GO TO 220                                                         MM  1158
150   CONTINUE                                                          MM  1159
      IF (ARY(21).LT.0.) RETURN                                         MM  1160
      ARY(22) = 0.5*(ARY(21)+ARY(23))                                   MM  1161
      GO TO 130                                                         MM  1162
160   CONTINUE                                                          MM  1163
      IF (ARY(IPICK+1).LT.0.) GO TO 180                                 MM  1164
      IF (IPICK+2.GT.24) GO TO 180                                      MM  1165
      IF (ARY(IPICK+2).LT.0.) GO TO 170                                 MM  1166
C                                                                       MM  1167
C CASE 4                                                                MM  1168
```

```
      VAL = ARY(IPICK+1)+(ARY(IPICK+2)-ARY(IPICK+1))                    MM  1169
      GO TO 220                                                         MM  1170
170   CONTINUE                                                          MM  1171
      IF (IPICK+3.GT.24) GO TO 180                                      MM  1172
      IF (ARY(IPICK+3).LT.0.) GO TO 180                                 MM  1173
C                                                                       MM  1174
C CASE 5                                                                MM  1175
      VAL = ARY(IPICK+1)+0.5*(ARY(IPICK+3)-ARY(IPICK+1))                MM  1176
      GO TO 220                                                         MM  1177
180   CONTINUE                                                          MM  1178
      VAL = VAL-1.                                                      MM  1179
      IF (VAL.LT.-3.) RETURN                                            MM  1180
      IF (IPICK-2.LT.1) GO TO 210                                       MM  1181
      IF (ARY(IPICK-2).LT.0.) GO TO 210                                 MM  1182
      IF (IPICK-3.LT.1) GO TO 210                                       MM  1183
      IF (ARY(IPICK-3).LT.0.) GO TO 210                                 MM  1184
C                                                                       MM  1185
C CASE 3                                                                MM  1186
      VAL = ARY(IPICK-2)+2.*(ARY(IPICK-2)-ARY(IPICK-3))                 MM  1187
      GO TO 220                                                         MM  1188
190   CONTINUE                                                          MM  1189
      IF (IPICK-2.LT.1) GO TO 210                                       MM  1190
      IF (ARY(IPICK-2).LT.0.) GO TO 200                                 MM  1191
C                                                                       MM  1192
C CASE 1                                                                MM  1193
      VAL = ARY(IPICK-1)+(ARY(IPICK-1)-ARY(IPICK-2))                    MM  1194
      GO TO 220                                                         MM  1195
200   CONTINUE                                                          MM  1196
      IF (IPICK-3.LT.1) GO TO 210                                       MM  1197
      IF (ARY(IPICK-3).LT.0.) GO TO 210                                 MM  1198
C                                                                       MM  1199
C CASE 2                                                                MM  1200
      VAL = ARY(IPICK-1)+0.5*(ARY(IPICK-1)-ARY(IPICK-3))                MM  1201
      GO TO 220                                                         MM  1202
210   CONTINUE                                                          MM  1203
      VAL = VAL-1.                                                      MM  1204
      IF (VAL.LT.-3.) RETURN                                            MM  1205
      IF (IPICK+2.GT.24) GO TO 180                                      MM  1206
      IF (ARY(IPICK+2).LT.0.) GO TO 180                                 MM  1207
      IF (IPICK+3.GT.24) GO TO 180                                      MM  1208
      IF (ARY(IPICK+3).LT.0.) GO TO 180                                 MM  1209
C                                                                       MM  1210
C CASE 6                                                                MM  1211
      VAL = ARY(IPICK+2)-2.*(ARY(IPICK+3)-ARY(IPICK+2))                 MM  1212
220   CONTINUE                                                          MM  1213
      IF (VAL.LT.0.) VAL = 0.                                           MM  1214
      RETURN                                                            MM  1215
                                                                        MM  1216
C                                                                       MM  1217
      END                                                               MM  1218
```

```
      FUNCTION FULGOL(A,Z0)                                              MM  1219
C-----THIS FUNCTION COMPUTES 1/L FROM THE FULLE PARAMETER A USING        MM  1220
C     GOLDER'S CONVERSIONS FROM STABILITY CATEGORIES TO VALUES OF        MM  1221
C     1/L.                                                               MM  1222
C                                                                        MM  1223
      IF(A.GT.8.0) GO TO 101                                             MM  1224
      SLOPE=0.0156-0.0072*(A-7.)                                         MM  1225
      XL=0.0540*(A-8.33)+SLOPE*ALOG(2.*Z0)                               MM  1226
      IF(XL.GT.-0.003) XL=-0.003                                         MM  1227
      GO TO 107                                                          MM  1228
  101 IF(A.GT.8.75) GO TO 102                                            MM  1229
      SLOPE=0.0084-0.0085*(A-8.)                                         MM  1230
      XL=0.0213*(A-8.84)+SLOPE*ALOG(2.*Z0)                               MM  1231
      IF(XL.GT.-0.0001) XL=-0.0001                                       MM  1232
      GO TO 107                                                          MM  1233
  102 IF(A.GT.9.02) GO TO 103                                            MM  1234
      SLOPE=0.0019-0.0070*(A-8.75)                                       MM  1235
      XL=0.0074*(A-9.02)+SLOPE*ALOG(2.*Z0)                               MM  1236
      IF(XL.GT.-0.00001) XL=-0.00001                                     MM  1237
      GO TO 107                                                          MM  1238
  103 IF(A.GT.9.5) GO TO 104                                             MM  1239
      SLOPE=-0.0029*(A-9.02)                                             MM  1240
      XL=0.0031*(A-9.02)+SLOPE*ALOG(2.*Z0)                               MM  1241
      IF(XL.LT.0.00001) XL=0.00001                                       MM  1242
      GO TO 107                                                          MM  1243
  104 IF(A.GT.11.25) GO TO 105                                           MM  1244
      SLOPE=-0.0014-0.0040*(A-9.5)                                       MM  1245
      XL=0.0083*(A-9.32)+SLOPE*ALOG(2.*Z0)                               MM  1246
      IF(XL.LT.0.0001) XL=0.0001                                         MM  1247
      GO TO 107                                                          MM  1248
  105 IF(A.GT.12.375) GO TO 106                                          MM  1249
      SLOPE=-0.0084-0.0069*(A-11.25)                                     MM  1250
      XL=0.0083*(A-9.32)+SLOPE*ALOG(2.*Z0)                               MM  1251
      IF(XL.LT.0.004) XL=0.004                                           MM  1252
      GO TO 107                                                          MM  1253
  106 SLOPE=-0.0162-0.0069*(A-12.375)                                    MM  1254
      XL=0.0083*(A-9.32)+SLOPE*ALOG(2.*Z0)                               MM  1255
      IF(XL.LT.0.00667) XL=0.00667                                       MM  1256
  107 FULGOL=XL                                                          MM  1257
      RETURN                                                             MM  1258
      END                                                                MM  1259


      SUBROUTINE GETAZV(KOLD,ID,XS,YS,NC,AZV,NGOOD,DDIS)                 MM  1260
C                                                                        MM  1261
C     THIS SUBROUTINE RETRIEVES MEASUREMENT RECORDS FROM                 MM  1262
C     CLOSE STATIONS.                                                    MM  1263
C                                                                        MM  1264
C     KOLD(1)  = TIME = 24 HOUR CLOCK                                    MM  1265
C     KOLD(2)  DATE = YEAR-MONTH-DAY                                     MM  1266
C     ID       STATION ID (IF ANY)                                       MM  1267
C     XS       X COORDINATE OF POSITION                                  MM  1268
C     YS       Y COORDINATE OF POSITION                                  MM  1269
C     NC       NUMBER OF CLOSEST STATIONS TO USE                         MM  1270
```

```
C      AZV(1)    AZIMUTH                                                    MM  1271
C      AZV(2)    VELOCITY                                                   MM  1272
C      AZV(3)    NUMBER OF DIGITAL WIND POINTS                              MM  1273
C      AZV(4)    NAME OF CLOSEST POINT                                      MM  1274
C      NGOOD     NUMBER OF POINTS ACTUALLY FOUND TO BE GOOD                 MM  1275
C      DDIS      DISTANCE TO POINTS FOUND                                   MM  1276
       DIMENSION KOLD(2),AZV(12)                                            MM  1277
       DIMENSION PT(24),XIMPH(24)                                           MM  1278
                                                                            MM  1279
       DIMENSION DIS(25) ,  IDIS(25)                                        MM  1280
       DIMENSION DDIS(2)                                                    MM  1281
       DIMENSION ID(2)                                                      MM  1282
       DIMENSION NAM(2)                                                     MM  1283
       DIMENSION NAME(2)                                                    MM  1284
       DIMENSION   NWDATA(52,25)                                            MM  1285
       COMMON /DATES/  IDATE(10),  ICT(10),  NFILES                         MM  1286
       COMMON  /ORIGIN/  UTMXOR, UTMYOR                                     MM  1287
       COMMON /WDATA/ NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX      MM  1288
       COMMON /WINFLD/  AWDATA(52,25) ,DISAVE(25),   IDSAVE(25)             MM  1289
       EQUIVALENCE   (NWDATA(1,1),AWDATA(1,1))                              MM  1290
       EQUIVALENCE (XID,KD)                                                 MM  1291
       EQUIVALENCE  (DIS,DISAVE), (IDIS,IDSAVE)                             MM  1292
       DATA IBLK /1H /, IFILE/11/                                           MM  1293
       DATA   INCOR/0/                                                      MM  1294
       DATA LOUT /6/                                                        MM  1295
C                                                                           MM  1296
C ZERO OUT AZV ARRAY                                                        MM  1297
       KD = IBLK                                                            MM  1298
       DO 100 I = 1,3                                                       MM  1299
       L = I-1                                                              MM  1300
       AZV(4*L+1) = 0                                                       MM  1301
       AZV(4*L+2) = 0                                                       MM  1302
       AZV(4*L+3) = XID                                                     MM  1303
       AZV(4*L+4) = XID                                                     MM  1304
100    CONTINUE                                                             MM  1305
C                                                                           MM  1306
C************************    ST. LOUIS ONLY   ****************************MM  1307
C                                                                           MM  1308
C         REMOVE THE CARDS BETWEEN THE STARED LINES FOR OTHER CITIES        MM  1309
C                                                                           MM  1310
C                                                                           MM  1311
       I = 1                                                                MM  1312
       IF = 1                                                               MM  1313
       IF(ID(1) .EQ. IBLK  )  GO TO 170                                     MM  1314
       GO TO 145                                                            MM  1315
  105  CONTINUE                                                             MM  1316
*****************************************************************************MM  1317
C      ARE WE AT A STATION                                                  MM  1318
       IF (ID(1).EQ.IBLK) GO TO 170                                         MM  1319
C                                                                           MM  1320
C      YES GET THE STATION DATA                                             MM  1321
       DO 110 I = 1,NFILES                                                  MM  1322
       IF (KOLD(2).EQ.IDATE(I)) GO TO 130                                   MM  1323
110    CONTINUE                                                             MM  1324
120    CONTINUE                                                             MM  1325
```

```
      WRITE(LOUT,1) KOLD(2)                                              MM  1326
      NGOOD = 0                                                          MM  1327
      RETURN                                                             MM  1328
130   CONTINUE                                                           MM  1329
      IC = 0                                                             MM  1330
      IF = IFILE + I                                                     MM  1331
      IF(INCOR.EQ.IDATE(I))  GO TO 145                                   MM  1332
      REWIND IF                                                          MM  1333
140   CONTINUE                                                           MM  1334
      IC = IC+1                                                          MM  1335
      IF(IC.GT.ICT(I)) GO TO 144                                         MM  1336
      READ (IF) NWDTE,NAM,APT,PT,XIMPH                                   MM  1337
      NWDATA(1,IC) = NWDTE                                               MM  1338
      NWDATA(2,IC) = NAM(1)                                              MM  1339
      NWDATA(3,IC) = NAM(2)                                              MM  1340
      AWDATA(4,IC) = APT                                                 MM  1341
                                                                         MM  1342
      CALL XMIT(24,PT,AWDATA(5,IC))                                      MM  1343
      CALL XMIT(24,XIMPH,AWDATA(29,IC))                                  MM  1344
      GO TO 140                                                          MM  1345
C                                                                        MM  1346
C                                                                        MM  1347
144   INCOR = IDATE(I)                                                   MM  1348
145   CONTINUE                                                           MM  1349
C                                                                        MM  1350
C     SEARCH FOR STATION                                                 MM  1351
C                                                                        MM  1352
      M = ICT(I)                                                         MM  1353
      DO 148 K = 1,M                                                     MM  1354
      NAM(1) = NWDATA(2,K)                                               MM  1355
      NAM(2) = NWDATA(3,K)                                               MM  1356
      IF(ID(1).EQ.NAM(1).AND.ID(2).EQ.NAM(2)) GO TO 149                  MM  1357
148   CONTINUE                                                           MM  1358
      GO TO 150                                                          MM  1359
149   CONTINUE                                                           MM  1360
      CALL XMIT(24,AWDATA(5,K),PT)                                       MM  1361
      CALL XMIT(24,AWDATA(29,K),XIMPH)                                   MM  1362
      APT = AWDATA(4,K)                                                  MM  1363
C                                                                        MM  1364
C     WE HAVE FOUND THE DATA, NOW EXTRAPOLATE IF NECESSARY               MM  1365
      CALL EXTRP(KOLD,PT,AZV(1))                                         MM  1366
      IF (AZV(1).LT.0.) GO TO 160                                        MM  1367
      CALL EXTRP(KOLD,XIMPH,AZV(2))                                      MM  1368
      IF (AZV(2).LT.0.) GO TO 160                                        MM  1369
      AZV(3) = APT                                                       MM  1370
      NGOOD = 1                                                          MM  1371
      DDIS(1) = .00000001                                                MM  1372
C        FOR ST. LOUIS USE ID(2)  NOT  ID(1)                             MM  1373
      KO = ID(2)                                                         MM  1374
      AZV(4) = XID                                                       MM  1375
      RETURN                                                             MM  1376
150   CONTINUE                                                           MM  1377
      WRITE(LOUT,2) ID,KOLD(2)                                           MM  1378
      GO TO 170                                                          MM  1379
C                                                                        MM  1380
```

```
C DATA RECORDS HAVE GAPS, TRY NEIGHBORING STATION(S)                    MM  1381
160   CONTINUE                                                          MM  1382
      WRITE(LOUT,3) ID,KOLD                                             MM  1383
C                                                                       MM  1384
C ORDER THE STATIONS AND CHECK THEIR DATA                               MM  1385
170   CONTINUE                                                          MM  1386
      CALL WSECLS(XS,YS,DIS,IDIS)                                       MM  1387
      KCT = 0                                                           MM  1388
      NGOOD = 0                                                         MM  1389
180   CONTINUE                                                          MM  1390
      IF (NGOOD.EQ.NC) GO TO 240                                        MM  1391
      KCT = KCT+1                                                       MM  1392
      IF (KCT.GT.NUMSTA) GO TO 240                                      MM  1393
      KUSE = IDIS(KCT)                                                  MM  1394
      NAME(1) = ISTANS(1,KUSE)                                          MM  1395
      NAME(2) = ISTANS(2,KUSE)                                          MM  1396
      DO 190 I=1,NFILES                                                 MM  1397
      IF (KOLD(2).EQ.IDATE(I)) GO TO 200                                MM  1398
190   CONTINUE                                                          MM  1399
      GO TO 120                                                         MM  1400
200   CONTINUE                                                          MM  1401
C                                                                       MM  1402
C         **********  FOR ST. LOUIS ONLY  ******************************MM  1403
C                                                                       MM  1404
                                                                        MM  1405
C         REMOVE CARDS BETWEEN THESE LINES FOR OTHER CITIES             MM  1406
      I = 1                                                             MM  1407
      GO TO 215                                                         MM  1408
 205  CONTINUE                                                          MM  1409
C                                                                       MM  1410
C         ************************************************************MM  1411
      IC = 0                                                            MM  1412
      IF = IFILE + I                                                    MM  1413
      IF(INCOR.EQ.IDATE(I)) GO TO 215                                   MM  1414
      REWIND IF                                                         MM  1415
210   CONTINUE                                                          MM  1416
      IC = IC + 1                                                       MM  1417
        IF(IC.GT.ICT(I)) GO TO 214                                      MM  1418
      READ (IF) NWDTE,NAM,APT,PT,XIMPH                                  MM  1419
      NWDATA(1,IC) = NWDTE                                              MM  1420
      NWDATA(2,IC) = NAM(1)                                             MM  1421
      NWDATA(3,IC) = NAM(2)                                             MM  1422
      AWDATA(4,IC) = APT                                                MM  1423
      CALL XMIT(24,PT,AWDATA(5,IC))                                     MM  1424
      CALL XMIT(24,XIMPH,AWDATA(29,IC))                                 MM  1425
      GO TO 210                                                         MM  1426
C                                                                       MM  1427
214   INCOR = IDATE(I)                                                  MM  1428
215   CONTINUE                                                          MM  1429
C                                                                       MM  1430
C     SEARCH FOR STATION                                                MM  1431
C                                                                       MM  1432
      M = ICT(I)                                                        MM  1433
      DO 217 K = 1,M                                                    MM  1434
      NAM(1) = NWDATA(2,K)                                              MM  1435
```

```
      NAM(2) = NWDATA(3,K)                                                  MM  1436
      IF(NAME(1).EQ.NAM(1).AND.NAME(2).EQ.NAM(2)) GO TO 220                 MM  1437
217   CONTINUE                                                              MM  1438
      GO TO 180                                                             MM  1439
220   CONTINUE                                                              MM  1440
      APT = AWDATA(4,K)                                                     MM  1441
      CALL XMIT(24,AWDATA(5,K),PT)                                          MM  1442
      CALL XMIT(24,AWDATA(29,K),XIMPH)                                      MM  1443
C                                                                           MM  1444
      CALL EXTRP(KOLD,PT,ZAZ)                                               MM  1445
      IF (ZAZ.LT.0.) GO TO 180                                              MM  1446
      CALL EXTRP(KOLD,XIMPH,VELL)                                           MM  1447
      IF (VELL.LT.0.) GO TO 180                                             MM  1448
      AZV(4*NGOOD+1) = ZAZ                                                  MM  1449
      AZV(4*NGOOD+2) = VELL                                                 MM  1450
      AZV(4*NGOOD+3) = APT                                                  MM  1451
      KD = NAM(2)                                                           MM  1452
      AZV(4*NGOOD+4) = XID                                                  MM  1453
      DDIS(NGOOD+1) = SQRT(DIS(KCT))                                        MM  1454
      IF (DIS(KCT).LE.RMIN) GO TO 230                                       MM  1455
      IF (DIS(KCT).GT.RMAX) GO TO 240                                       MM  1456
C                                                                           MM  1457
C PERFORM ANGLE TEST                                                        MM  1458
      CALL ANGTST(XS,YS,NGOOD,KUSE,OK)                                      MM  1459
      IF (OK.LT.0.) GO TO 180                                               MM  1460
C                                                                           MM  1461
C PERFORM THE BARRIER TEST                                                  MM  1462
      XT = SSTAN(1,KUSE)                                                    MM  1463
      YT = SSTAN(2,KUSE)                                                    MM  1464
      CALL BARIER(XT,YT,XS,YS,OK)                                           MM  1465
      IF (OK.LT.0.) GO TO 180                                               MM  1466
      NGOOD = NGOOD+1                                                       MM  1467
                                                                            MM  1468
      GO TO 180                                                             MM  1469
230   CONTINUE                                                              MM  1470
      NGOOD = 1                                                             MM  1471
240   CONTINUE                                                              MM  1472
C                                                                           MM  1473
C     CHECK WHETHER DISTANCES ARE EQUAL TO ZERO                             MM  1474
C                                                                           MM  1475
      IF(NGOOD.EQ.0) RETURN                                                 MM  1476
      IF(DDIS(1).EQ.0.) DDIS(1) = 1.0E-8                                    MM  1477
      IF(NGOOD.EQ.1) RETURN                                                 MM  1478
      DO 250 I = 2,NGOOD                                                    MM  1479
      IF(DDIS(I).NE.0.) GO TO 250                                           MM  1480
      WRITE(LOUT,4) XS,YS                                                   MM  1481
      STOP                                                                  MM  1482
250   CONTINUE                                                              MM  1483
      RETURN                                                                MM  1484
1     FORMAT (//,33H GETAZV CANT FIND DATE FILE FOR  ,I6,                   MM  1485
     1          22H  IN WIND DATA TABLES,,//)                              MM  1486
2     FORMAT (/,54H GETAZV CANT FIND WIND DATA FOR THIS STATION AND DATE    MM  1487
     1. ,2X,A4,A4,2X,I6,/,32H WILL CHECK NEAREST NEIGHBOR(S).,/)            MM  1488
3     FORMAT (/,51H INCOMPLETE DATA PREVENTS USE OF THIS STATION DATA,,     MM  1489
     1  2X,A4,A4,2X,I6,2X,I6,/,30H MUST USE NEAREST NEIGHBOR(S).,/)        MM  1490
```

```
4     FORMAT (1H0,2X,33HMULTIPLE WIND STATIONS FOUND AT (,F8.2,1H,,      MM  1491
     1        F8.2,15H). JOB ABORTED.)                                   MM  1492
      END                                                               MM  1493



      SUBROUTINE GETCON (KNEW,NC,IRF)                                   MM  1494
C                                                                       MM  1495
C       GETCON FINDS THE 'NC' CLOSEST STATIONS WITH DATA FOR EACH       MM  1496
C       VARIABLE IN THE AIR QUALITY DATA ARRAY (CON) AND                MM  1497
C       DETERMINES THE INTERPOLATED VALUES.                             MM  1498
C                                                                       MM  1499
      DIMENSION  KNEW(2), IDD(2), ISTUSE(3,10), ACON(4,10),             MM  1500
     1IWORK(25), WORK(25),  R(3),  W(3)   , DIST(3)                     MM  1501
      COMMON /AIRQAL/ NEEDAQ,   NOSPEC,   CONAM(10),   CON(24,25,10)    MM  1502
      COMMON /INPUTS/ TITLE(20),        JDATE(10),          NCURV       MM  1503
      COMMON /KZINPT/ NEEDKZ,           KZPRTX,             NUMSD,      MM  1504
     1                TIMESD(3),        TEMPSD(50,3),       ZELVSD(50,3)MM  1505
     2        ,       TEMPSF(24),       NPTSD(3)       ,    NKZDAT      MM  1506
      COMMON /REUSE/  KPSTAT, KPWDAT, CONVRT, KXTRA, KPWIND             MM  1507
      COMMON /WDATA/  NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX  MM  1508
      COMMON /WINFLD/ AWDATA(52,25) ,DISAVE(25),   IDSAVE(25)           MM  1509
      EQUIVALENCE (WORK,IWORK)                                          MM  1510
      INTEGER ATSTAT                                                    MM  1511
      DATA  LOUT /6/                                                    MM  1512
      DATA IYES  /4HYES /                                               MM  1513
      DATA NONE /4HNONE/                                                MM  1514
      N = NOSPEC                                                        MM  1515
      IHR = KNEW(1)                                                     MM  1516
      NCP = NC + 1                                                      MM  1517
C                                                                       MM  1518
      N1 = N*4                                                          MM  1519
      CALL XMIT(-N1,-1.00,ACON)                                         MM  1520
      N1 = N1- N                                                        MM  1521
      CALL XMIT(-N1,NONE,ISTUSE)                                        MM  1522
C                                                                       MM  1523
      ATSTAT = NONE                                                     MM  1524
      IF(DISAVE(1).GT. RMIN)  GO TO 90                                  MM  1525
      ATSTAT = IYES                                                     MM  1526
      KD = IDSAVE(1)                                                    MM  1527
   90 CONTINUE                                                          MM  1528
      DO 300 JS = 1,N                                                   MM  1529
      KSUC = 0                                                          MM  1530
                                                                        MM  1531
      KTRY = 0                                                          MM  1532
      IF(ATSTAT.NE.IYES)  GO TO 100                                     MM  1533
C       THE TRAJECTORY IS AT A STATION                                  MM  1534
      CALL XMIT (24,CON(1,KD,JS),WORK)                                  MM  1535
      CALL EXTRP (IHR,WORK,VAL)                                         MM  1536
      IF (VAL.LT.0.)  GO TO 100                                         MM  1537
C       INTERPOLATION OF DATA FROM OTHER STATIONS IS UNNECESSARY.       MM  1538
      ISTUSE(1,JS) = ISTANS(2,KD)                                       MM  1539
      ACON(1,JS) =    VAL                                               MM  1540
      ACON(NCP,JS) =  VAL                                               MM  1541
      GO TO 300                                                         MM  1542
```

```
C                                                                       MM  1543
  100 CONTINUE                                                          MM  1544
C     FIND AND CHECK DATA FROM CLOSEST STATIONS.                        MM  1545
  110 KTRY = KTRY + 1                                                   MM  1546
      IF(KTRY.GT.NUMSTA) GO TO 120                                      MM  1547
      IF(DISAVE(KTRY) .GT. RMAX)  GO TO 130                             MM  1548
      KT = IDSAVE(KTRY)                                                 MM  1549
      CALL XMIT(24, CON(1,KT,JS), WORK )                                MM  1550
      CALL EXTRP (IHR,WORK,VAL)                                         MM  1551
      IF(VAL.LT.0.0)  GO TO 110                                         MM  1552
      KSUC = KSUC + 1                                                   MM  1553
      ISTUSE(KSUC,JS) = ISTANS(2,KT)                                    MM  1554
      ACON(KSUC,JS) = VAL                                               MM  1555
      DIST(KSUC) = SQRT( DISAVE(KTRY) )                                 MM  1556
      IF (KSUC.LT. NC)  GO TO 110                                       MM  1557
  120 KTRY = MIN0(KTRY,NUMSTA)                                          MM  1558
  130 CONTINUE                                                          MM  1559
      IF(KSUC.LT.1)  GO TO 300                                          MM  1560
C                                                                       MM  1561
C       NOW INTERPOLATE THE VALUES FROM THE CLOSEST STATIONS            MM  1562
C                                                                       MM  1563
      SUMR = 0.0                                                        MM  1564
      DO 220  I=1,KSUC                                                  MM  1565
      IF(IRF.EQ. 2)  GO TO 200                                          MM  1566
      R(I) = 1./DIST(I)                                                 MM  1567
      GO TO 210                                                         MM  1568
  200 R(I) = 1./(DIST(I)**2)                                            MM  1569
C                                                                       MM  1570
  210 SUMR = SUMR + R(I)                                                MM  1571
  220 CONTINUE                                                          MM  1572
C                                                                       MM  1573
      SUMW = 0.0                                                        MM  1574
      DO 230 I = 1,KSUC                                                 MM  1575
      W(I) = R(I)/SUMR                                                  MM  1576
      SUMW = SUMW + ACON(I,JS)*W(I)                                     MM  1577
  230 CONTINUE                                                          MM  1578
      ACON(NCP,JS) = SUMW                                               MM  1579
C                                                                       MM  1580
  300 CONTINUE                                                          MM  1581
C                                                                       MM  1582
      IF(KXTRA.NE.IYES)   GO TO 410                                     MM  1583
      RS = SQRT(RMAX)                                                   MM  1584
      DO 350 I = 1,NUMSTA                                               MM  1585
      J = IDSAVE(I)                                                     MM  1586
      IWORK(I) = ISTANS(2,J)                                            MM  1587
      DISAVE(I) = SQRT( DISAVE(I))                                      MM  1588
      IF(DISAVE(I).GT.RS)  GO TO 360                                    MM  1589
  350 CONTINUE                                                          MM  1590
  360 KTRYM = I - 1                                                     MM  1591
C                                                                       MM  1592
C       WRITE STATION VALUES AND INTERPOLATED VALUES ON FILE.           MM  1593
                                                                        MM  1594
C                                                                       MM  1595
      WRITE(LOUT,1) KNEW(1),  ( IWORK(I),I=1,KTRYM)                     MM  1596
      WRITE(LOUT,2) KNEW(2),  (DISAVE(I),I=1,KTRYM)                     MM  1597
```

```
      WRITE(LOUT,3)   (K,K=1,NC)                                          MM  1598
      DO 400  JS = 1,N                                                    MM  1599
      WRITE(LOUT,4) CONAM(JS), (ACON(I,JS),ISTUSE(I,JS),I=1,NC),          MM  1600
     1   ACON(NCP,JS)                                                     MM  1601
  400 CONTINUE                                                            MM  1602
      GO TO 420                                                           MM  1603
  410 CONTINUE                                                            MM  1604
      WRITE(LOUT,5)  KNEW                                                 MM  1605
      WRITE(LOUT,6) (CONAM(J), ACON(NCP,J),J=1,N)                         MM  1606
  420 CONTINUE                                                            MM  1607
      N1 = 2*RMAX                                                         MM  1608
      CALL XMIT(-NUMSTA,N1,DISAVE)                                        MM  1609
      CALL XMIT(-NUMSTA, 0,IDSAVE)                                        MM  1610
      IHR = KNEW(1)/100 + 1                                               MM  1611
      TEMPSF(IHR) = ACON(NCP,1)                                           MM  1612
      RETURN                                                              MM  1613
C                                                                         MM  1614
   1  FORMAT(1H ,/,3X,6HTIME =,I6,5X,16HSTATION NUMBERS  , 4X,            MM  1615
     1 5(2X,A4),/,4(40X,2X,A4,2X,A4,2X,A4,2X,A4,2X,A4,/))                 MM  1616
   2  FORMAT(1H ,/,3X,6HDATE =,I7,4X,16HDISTANCES (KM)   ,4X,5F6.2,/,     MM  1617
     1    4(40X,5F6.2,/))                                               MM  1618
   3  FORMAT (1H0,3X,9HVARIABLES ,3(5X,1HX,I1,11H   (STAT)   ),           MM  1619
     1  14HX=INTERPOLATED  )                                             MM  1620
   4  FORMAT(1H ,4X,A4,4X,3(E9.2,2H (,A4,3H)   ),3X,E11.4 )               MM  1621
   5  FORMAT (1H0,4X,6HTIME =, I5,5X,6HDATE = , I7,//,5X,                 MM  1622
     1  31HVARIABLE     INTERPOLATED VALUE   )                           MM  1623
   6  FORMAT (1H0,7X,A8,10X,E12.4,//,9(8X,A4,10X,E12.4,//))               MM  1624
      END                                                                 MM  1625



      SUBROUTINE JULIAN (IDATE,JULDAT)                                    MM  1626
C                                                                         MM  1627
C*************************    ST. LOUIS ONLY   ****************************MM  1628
C                                                                         MM  1629
C       THIS SUBROUTINE CONVERTS A 6-INTEGER DATE (YR-MO-DAY) IN TO       MM  1630
C       A JULIAN DATE.....JULDAT(1)=DAY.........JULDAT(2)=YEAR            MM  1631
C                                                                         MM  1632
      DIMENSION MD(11) ,  JULDAT(2)                                       MM  1633
      DATA MD /31,28,31,30,31,30,31,31,30,31,30/                          MM  1634
C                                                                         MM  1635
      IY = IDATE/10000                                                    MM  1636
      IM = (IDATE - IY*10000)/100                                         MM  1637
      ID = IDATE - IY*10000 - IM*100                                      MM  1638
C                                                                         MM  1639
C       CHECK FOR LEAP YEARS                                              MM  1640
C                                                                         MM  1641
      NL = MOD(IY,4)                                                      MM  1642
      IF(NL.LT. 1)   MD(2) = 29                                           MM  1643
C                                                                         MM  1644
      ISUM = 0                                                            MM  1645
      IM = IM - 1                                                         MM  1646
      DO 10 J = 1,IM                                                      MM  1647
      ISUM = ISUM + MD(J)                                                 MM  1648
   10 CONTINUE                                                            MM  1649
```

```
      JULDAT(1) = ISUM + ID                                              MM  1650
      JULDAT(2) = 1900 + IY                                              MM  1651
C                                                                        MM  1652
      RETURN                                                             MM  1653
      END                                                                MM  1654



      SUBROUTINE KEKLAY(LMAX,ZMAX,ZHIGH,AKMIN,DH)                        MM  1655
C                                                                        MM  1656
                                                                         MM  1657
C-----THIS SUBROUTINE DETERMINES THE VALUES OF KZ IN THE EKMAN LAYER     MM  1658
C     IN THE MANNER SUGGESTED BY O'BRIEN(1970).                          MM  1659
C                                                                        MM  1660
      COMMON/DIFDAT/DELTAT(30),STAB(30),ZMIX(30),USTAR(30),              MM  1661
     +              UZZ(201),AKZ(201)                                    MM  1662
      COMMON/TEMPHT/Z(50),ZZ(50),T(50),TT(50)                            MM  1663
      DO 229 J=2,LMAX                                                    MM  1664
      ZHT=Z(J-1)                                                         MM  1665
      ZB=0.10*ZMIX(J-1)                                                  MM  1666
      DZB=ZMIX(J-1)-ZB                                                   MM  1667
      IF(STAB(J-1).GT.0.) GO TO 226                                      MM  1668
  223 ZFREE=-5.*STAB(J-1)                                                MM  1669
      IF(ZB.GT.ZFREE) GO TO 224                                          MM  1670
      AKB=.35*USTAR(J-1)*ZB*(1.-15.*ZB/STAB(J-1))**.25                   MM  1671
      DKZB=.35*USTAR(J-1)*((1.-18.75*ZB/STAB(J-1))/(1.-15.*ZB/STAB(J-1))MM  1672
     +**.75)                                                             MM  1673
      GO TO 227                                                          MM  1674
  224 ZB=ZFREE                                                           MM  1675
      AKB=.35*USTAR(J-1)*ZB*(1.-15.*ZB/STAB(J-1))**.25                   MM  1676
      DKZB=0.7*USTAR(J-1)*(-ZB/STAB(J-1))**.3333                         MM  1677
      GO TO 227                                                          MM  1678
  226 AKB=.35*USTAR(J-1)*ZB/(1.+4.7*ZB/STAB(J-1))                        MM  1679
      DKZB=.35*USTAR(J-1)/(1.+4.7*ZB/STAB(J-1))**2.                      MM  1680
  227 CONTINUE                                                           MM  1681
      AKD=AKB-AKMIN                                                      MM  1682
  228 ZHT=ZHT+DH                                                         MM  1683
      IF(ZHT.GT.Z(J)) GO TO 229                                          MM  1684
      IF(ZHT.LE.ZB) GO TO 228                                            MM  1685
      NZHT=IFIX(ZHT/DH)+1                                                MM  1686
      AKZ(NZHT)=AKMIN                                                    MM  1687
      DELZ=ZHT-ZMIX(J-1)                                                 MM  1688
      IF(DELZ.GT.0.) GO TO 228                                           MM  1689
      DEL2=DELZ*DELZ                                                     MM  1690
      AKZ(NZHT)=AKMIN+DELZ/(DZB**2.)*(AKD+(ZHT-ZB)*(DKZB+2.*AKD/DZB))    MM  1691
      GO TO 228                                                          MM  1692
  229 CONTINUE                                                           MM  1693
  260 LMAXM1=LMAX-1                                                      MM  1694
      DO 270 J=2,LMAXM1                                                  MM  1695
      IF(STAB(J-1).GT.0.) GO TO 270                                      MM  1696
      NZ=IFIX(Z(J-1)/DH)+1                                               MM  1697
      NZZ=IFIX(Z(J)/DH)+1                                                MM  1698
      AKZ(NZZ)=2.*AKZ(NZZ+1)-AKZ(NZZ+2)                                  MM  1699
      ZHT=Z(J-1)                                                         MM  1700
      IF(ZHT.GE.ZMAX) GO TO 275                                          MM  1701
```

```
  265 ZHT=ZHT+DH                                                        MM  1702
      IF(ZHT.GT.Z(J)) GO TO 270                                         MM  1703
      ZHZ=((Z(J)-ZHT)/(Z(J)-Z(J-1)))**.33333                            MM  1704
      NZHT=IFIX(ZHT/DH)+1                                               MM  1705
      ADDKZ=(ZHT-Z(J-1))*(AKZ(NZZ)-AKZ(NZ))/(Z(J)-Z(J-1))+AKZ(NZ)       MM  1706
      AKZ(NZHT)=(AKZ(NZHT)-ADDKZ)*ZHZ+ADDKZ                             MM  1707
      IF(AKZ(NZHT).LT.AKMIN) AKZ(NZHT)=AKMIN                            MM  1708
      GO TO 265                                                         MM  1709
  270 CONTINUE                                                          MM  1710
  275 IF(LMAX.LT.3) GO TO 290                                           MM  1711
      DO 280 J=3,LMAX                                                   MM  1712
      NZ=IFIX(Z(J-1)/DH)+1                                              MM  1713
      IF(AKZ(NZ+1).LE.AKZ(NZ)) GO TO 280                                MM  1714
      ZHT=Z(J-1)                                                        MM  1715
      ZHTX=Z(J)                                                         MM  1716
      IF(ZMIX(J-1).LT.Z(J)) ZHTX=ZMIX(J-1)                              MM  1717
  276 ZHT=ZHT+DH                                                        MM  1718
      IF(ZHT.GT.Z(J)) GO TO 280                                         MM  1719
                                                                        MM  1720
      IF(ZHT.GT.ZMIX(J-1)) GO TO 280                                    MM  1721
      NZHT=IFIX(ZHT/DH)+1                                               MM  1722
      ZHZ=((ZHT-Z(J-1))/(ZHTX-Z(J-1)))**.25                             MM  1723
      ADDKZ=(ZMIX(J-1)-ZHT)*(AKZ(NZ)-AKMIN)/(ZMIX(J-1)-Z(J-1))+AKMIN    MM  1724
      AKZ(NZHT)=(AKZ(NZHT)-ADDKZ)*ZHZ+ADDKZ                             MM  1725
      GO TO 276                                                         MM  1726
  280 CONTINUE                                                          MM  1727
  290 RETURN                                                            MM  1728
      END                                                               MM  1729



      FUNCTION KLASS(A)                                                 MM  1730
C                                                                       MM  1731
C     *KLASS* DETERMINES THE ATMOSPHERIC STABILITY CLASS                MM  1732
C             FROM THE PULLE STABILITY PARAMETER (A).                   MM  1733
      KLASS = 6                                                         MM  1734
      IF(A.LE.11.25) KLASS = 5                                          MM  1735
      IF(A.LE. 9.50) KLASS = 4                                          MM  1736
      IF(A.LE. 8.75) KLASS = 3                                          MM  1737
      IF(A.LE. 8.00) KLASS = 2                                          MM  1738
      IF(A.LE. 7.00) KLASS = 1                                          MM  1739
      RETURN                                                            MM  1740
      END                                                               MM  1741



      SUBROUTINE KZDATA(IERR)                                           MM  1742
C                                                                       MM  1743
C       THIS SUBROUTINE READS (UP TO 3) VERTICAL TEMPERATURE PROFILES.  MM  1744
C       THIS SUBROUTINE READS HOURLY SURFACE TEMPERATURES (DEGREES C)   MM  1745
C       WHEN  RDTEMP EQUALS YES.                                        MM  1746
C                                                                       MM  1747
C       NOTE .. FOR ST. LOUIS RDTEMP EQUALS NO SINCE THE SURFACE        MM  1748
C               TEMPERATURES ARE INTERPOLATED ALONG THE TRAJECTORY.     MM  1749
C                                                                       MM  1750
```

```
      COMMON /KZINPT/ NEEDKZ,          KZPRTX,          NUMSD,       MM  1751
     1                TIMESD(3),       TEMPSD(50,3),    ZELVSD(50,3)MM  1752
     2,               TEMPSF(24),      NPTSD(3),        NKZDAT       MM  1753
      COMMON/CELECT/CELHTS(5,1),NOHTS,NMES                            MM  1754
      DATA LIN /3/                                                    MM  1755
      DATA YES, RNEG /4HYES ,4HNO  /                                  MM  1756
      EQUIVALENCE(KYES,YES)                                           MM  1757
C                                                                     MM  1758
      READ(LIN,1) NKZDAT                                              MM  1759
      IF(NKZDAT.NE.KYES) GO TO 120                                    MM  1760
      READ(LIN,1)  KZPRTX                                             MM  1761
C       READ VERTICAL PROFILES                                        MM  1762
      DO 50  ISD=1,4                                                  MM  1763
      READ(LIN,2)  A                                                  MM  1764
      IF(A.LT.0.)  GO TO 60                                           MM  1765
      IF(ISD.GE.4)  GO TO 131                                         MM  1766
      TIMESD(ISD) = A                                                 MM  1767
      DO 30  J=1,51                                                   MM  1768
      IF(J.GE.51)  GO TO 132                                          MM  1769
   10 READ(LIN,2)  B, C                                               MM  1770
      IF(B.LT.0.0)   GO TO 40                                         MM  1771
C**********************NOTE TO USER**********************            MM  1772
C                                                                     MM  1773
C     CARDS BETWEEN STARRED LINES SHOULD BE REMOVED                   MM  1774
C     FOR CITIES OTHER THAN ST. LOUIS.  THESE CARDS                   MM  1775
C     REMOVE FROM THE TEMPERATURE PROFILES THE POINTS                 MM  1776
C     HAVING HEIGHTS WHICH ARE MULTIPLES OF 100.0 M.                  MM  1777
C     IT IS OFTEN STANDARD PROCEDURE TO RECORD THESE                  MM  1778
C     HEIGHTS EVENTHOUGH THEY ARE NOT IMPORTANT TO                    MM  1779
C     THE SHAPE OF THE SOUNDING.                                      MM  1780
      IF(J.EQ.1) GO TO 20                                             MM  1781
      IB=IFIX(B*1000.+.1)                                             MM  1782
      IF(MOD(IB,100000).EQ.0) GO TO 10                                MM  1783
C*********************************************************            MM  1784
   20 ZELVSD(J,ISD) = B                                               MM  1785
                                                                      MM  1786
      TEMPSD(J,ISD) = C                                               MM  1787
   30 CONTINUE                                                        MM  1788
      GO TO 131                                                       MM  1789
   40 NPTSD(ISD) = J - 1                                              MM  1790
   50 CONTINUE                                                        MM  1791
      GO TO 132                                                       MM  1792
   60 CONTINUE                                                        MM  1793
      NUMSD=ISD-1                                                     MM  1794
      READ(LIN,1)  RDTEMP                                             MM  1795
      IF(RDTEMP.NE.YES)  GO TO 90                                     MM  1796
C       READ SURFACE TEMPERATURES                                     MM  1797
      DO 80 I= 1,25                                                   MM  1798
      READ(LIN,2) A, B                                                MM  1799
      IF(A.LT.0.0)  GO TO 90                                          MM  1800
      IS = IFIX((A+1.E-3)/100.)+1                                     MM  1801
      IF(IS.GE.25)  GO TO 133                                         MM  1802
      TEMPSF(IS) = B                                                  MM  1803
   80 CONTINUE                                                        MM  1804
      GO TO 133                                                       MM  1805
```

```
C                                                                       MM  1806
   90 CONTINUE                                                          MM  1807
      READ(LIN,4) NOHTS                                                 MM  1808
      READ(LIN,5)   (CELHTS(I,1),I=1,NOHTS)                             MM  1809
      NMES = 1                                                          MM  1810
C                                                                       MM  1811
  120 IERR=1                                                            MM  1812
      RETURN                                                            MM  1813
C                                                                       MM  1814
  131 IERR=-1                                                           MM  1815
      RETURN                                                            MM  1816
  132 IERR=-2                                                           MM  1817
      RETURN                                                            MM  1818
  133 IERR=-3                                                           MM  1819
      RETURN                                                            MM  1820
  134 IERR=-4                                                           MM  1821
      RETURN                                                            MM  1822
C                                                                       MM  1823
    1 FORMAT (40X,A4)                                                   MM  1824
    2 FORMAT (40X,2F10.0)                                               MM  1825
    4 FORMAT(40X,I2)                                                    MM  1826
    5 FORMAT(40X,F10.0)                                                 MM  1827
      END                                                               MM  1828



      SUBROUTINE METIN (IDT,LTAPE)                                      MM  1829
C                                                                       MM  1830
C        THIS IS THE SHORT RECORD  VERSION OF METIN                     MM  1831
C                                                                       MM  1832
C        THIS SUBROUTINE READS WIND SPEEDS AND DIRECTIONS, SURFACE      MM  1833
C        TEMPERATURES, AIR QUALITY DATA, AND SOLAR RADIATION DATA       MM  1834
C        FROM ST. LOUIS RAMS TAPES. .                                   MM  1835
C                                                                       MM  1836
C        THE WIND DATA IS STORED IN AWDATA.                             MM  1837
C        THE TEMPERATURE DATA IS STORED IN CON(I,J,1) AND TEMPSF(I).    MM  1838
C        THE AIR QUALITY DATA IS STORED IN CON(I,J,2 THRU 10).          MM  1839
C        THE RADIATION DATA IS REDUCED INTO UV(I) AND PRINTED ON LOUT.  MM  1840
C                                                                       MM  1841
C        NOTE......ONLY ONE DAYS WORTH OF DATA IS STORED AT A TIME.     MM  1842
C        THIS SUBROUTINE  FINDS THE DATA FOR THE DATE (IDT) SPECIFIED.  MM  1843
C                                                                       MM  1844
C                                                                       MM  1845
      DIMENSION AIMAGE(662) , WORK(550),  X(25,22), UV(24), JULDAT(2),  MM  1846
     1  NWDATA(52,25) , DTEMP(24) , ASHR(24) , LWDSTA(9), CLOUDY(24)    MM  1847
C                                                                       MM  1848
                                                                        MM  1849
      COMMON /AIRQAL/ NEEDAQ,   NOSPEC,   CONAM(10),   CON(24,25,10)    MM  1850
      COMMON /DATES/  IDATE(10),  ICT(10),  NFILES                      MM  1851
      COMMON /INPUTS/ TITLE(20),  JDATE(10),  NQURV                     MM  1852
      COMMON /KZINPT/ NEEDKZ,           KZPRTX,             NUMSD,      MM  1853
     1                TIMESD(3),        TEMPSD(50,3),       ZELVSD(50,3)MM  1854
     2         ,      TEMPSF(24),       NPTSD(3)        ,   NKZDAT      MM  1855
      COMMON /REUSE/  KPSTAT, KPWDAT, CONVRT, KXTRA, KPWIND             MM  1856
      COMMON /WDATA/  NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX  MM  1857
```

```
      COMMON /WHERE/  RLAT,   RLONG,     TMZONE                          MM  1858
      COMMON /WINFLD/ AWDATA(52,25) ,DISAVE(25),   IDSAVE(25)            MM  1859
C                                                                         MM  1860
      EQUIVALENCE (YES,IYES)                                              MM  1861
      EQUIVALENCE (X,WORK)                                                MM  1862
      EQUIVALENCE (NWDATA,AWDATA)                                         MM  1863
      DATA IYES, NO /4HYES ,4HNO  /                                       MM  1864
      DATA ZERO /0.0000000/                                               MM  1865
      DATA LIN,LOUT,LPUNCH /3,6,1/                                        MM  1866
C                                                                         MM  1867
C        ***********DATA SPECIFICALLY FOR ST. LOUIS**********************MM  1868
      DATA NODAYS /1/                                                     MM  1869
      DATA NRAM /4HRAMS/                                                  MM  1870
      DATA APT / 90./                                                     MM  1871
      DATA HIVAL / 500. /                                                 MM  1872
      DATA LWDSTA  / 108,110,114,115,116,117,118,121,0/                   MM  1873
      DATA NSTMAX , NSPMAX /25, 10/                                       MM  1874
      DATA CONAM /4HTEMP, 4HO3  , 4HCO  , 4HCH4 , 4HTHC , 4HNO  , 4HNO2 MM  1875
     1  ,     4HNOX , 4HTS  , 4HSO2  /                                    MM  1876
    6 NOSPEC = 10                                                         MM  1877
      DEBUG = NO                                                          MM  1878
C       ***************************************************************MM  1879
      LU = LTAPE                                                          MM  1880
      KNTER = 0                                                           MM  1881
      NFILE = 0                                                           MM  1882
      IMIN = 1                                                            MM  1883
      IMAX = 24                                                           MM  1884
C                                                                         MM  1885
      IDATE(1) = IDT                                                      MM  1886
      ICT(1) = NUMSTA                                                     MM  1887
      NFILES = 1                                                          MM  1888
C                                                                         MM  1889
C                                                                         MM  1890
      CALL JULIAN(IDT,JULDAT)                                             MM  1891
C                                                                         MM  1892
C        INITIALIZE ARRAYS                                                MM  1893
C                                                                         MM  1894
      IF(IMIN.EQ.1.AND.IMAX.EQ.24)  GO TO 95                              MM  1895
C                                                                         MM  1896
      CALL XMIT (24,-1.,UV)                                               MM  1897
      N = 52*NSTMAX                                                       MM  1898
      CALL XMIT(-N,-1.,AWDATA)                                            MM  1899
      N = 24*NSPMAX*NSTMAX                                                MM  1900
      CALL XMIT(-N,-1.,CON)                                               MM  1901
C                                                                         MM  1902
   95 CONTINUE                                                            MM  1903
C                                                                         MM  1904
      DO 100 J = 1,NUMSTA                                                 MM  1905
      NWDATA(1,J) = IDT                                                   MM  1906
      NWDATA(2,J) = ISTANS(1,J)                                           MM  1907
      NWDATA(3,J) = ISTANS(2,J)                                           MM  1908
      AWDATA(4,J) = APT                                                   MM  1909
  100 CONTINUE                                                            MM  1910
C                                                                         MM  1911
                                                                          MM  1912
```

```
C        RETRIEVE ONE HOURS WORTH OF DATA FROM TAPE                          MM  1913
C                                                                            MM  1914
 110  CONTINUE                                                               MM  1915
C                                                                            MM  1916
C         ON UNIVAC ELIMINATE CARDS BETWEEN STARED CARDS                     MM  1917
C         ELIMINATE AIMAGE FROM DIMENSION STATEMENT ABOVE                    MM  1918
C         ***************************************************************MM  1919
      BUFFER IN (LU,0) (AIMAGE(1),AIMAGE(2))                                 MM  1920
 120  IF(UNIT,LU)  120, 130, 460, 470                                        MM  1921
C           NOT READY  ....READY...EOF...PARITY                              MM  1922
 130  CONTINUE                                                               MM  1923
      DECODE(12,20,AIMAGE(1))  IYR,  IDAY,  IHR                              MM  1924
C         ***************************************************************MM  1925
C         ON UNIVAC INCLUDE THE FOLLOWING CARD                               MM  1926
C     READ (LU,20,ERR=470,END=460)  IYR, IDAY, IHR                           MM  1927
      IF(IDAY.EQ.JULDAT(1))  GO TO 140                                       MM  1928
C      SKIP FILE LU                                                          MM  1929
      NFILE = NFILE + 1                                                      MM  1930
      IF(NFILE.GE.NODAYS)  GO TO 450                                         MM  1931
      GO TO 110                                                              MM  1932
 140  CONTINUE                                                               MM  1933
C         ON UNIVAC ELIMINATE CARDS BETWEEN STARED CARDS                     MM  1934
C         ***************************************************************MM  1935
      I1 = 3                                                                 MM  1936
      I2 = I1 + 29                                                           MM  1937
      DO 145 I = 1,22                                                        MM  1938
      BUFFER IN (LU,0)  (AIMAGE(I1),AIMAGE(I2))                              MM  1939
 143  IF(UNIT,LU)  143,  144,  460,  470                                     MM  1940
 144  I1 = I2 + 1                                                            MM  1941
      I2 = I1 + 29                                                           MM  1942
 145  CONTINUE                                                               MM  1943
C         ***************************************************************MM  1944
C         ON UNIVAC INCLUDE THE FOLLOWING CARDS                              MM  1945
C     DO 150 K = 1,22                                                        MM  1946
C     READ (LU,40,ERR=470,END=460)  (X(J,K),J=1,NUMSTA)                      MM  1947
C150  CONTINUE                                                               MM  1948
      I1 = IMIN - 1                                                          MM  1949
      IF(IHR.LT,I1)  GO TO 110                                               MM  1950
C         ON UNIVAC ELIMINATE CARDS BETWEEN STARED CARDS                     MM  1951
C         ***************************************************************MM  1952
      I1 = 3                                                                 MM  1953
      J1 = 1                                                                 MM  1954
      J2 = 10                                                                MM  1955
      DO 150 K = 1,55                                                        MM  1956
      DECODE(120,40,AIMAGE(I1))  (WORK(JS),JS=J1,J2)                         MM  1957
      J1 = J1 + 10                                                           MM  1958
      J2 = J2 + 10                                                           MM  1959
      I1 = I1 + 12                                                           MM  1960
 150  CONTINUE                                                               MM  1961
C         ***************************************************************MM  1962
      KNTER = KNTER + 1                                                      MM  1963
C                                                                            MM  1964
      IF(DEBUG.NE. YES)  GO TO 170                                           MM  1965
      WRITE(LOUT,30) IYR, IDAY, IHR                                          MM  1966
      DO 160 K = 1,22                                                        MM  1967
```

```
      WRITE(LOUT,50)  K, (X(J,K),J=1,NUMSTA)                              MM  1968
  160 CONTINUE                                                            MM  1969
  170 CONTINUE                                                            MM  1970
C                                                                         MM  1971
  300 INDEX = IHR + 1                                                     MM  1972
C                                                                         MM  1973
C                                                                         MM  1974
                                                                          MM  1975
C        XMIT WIND DATA                                                   MM  1976
C                                                                         MM  1977
      I1 = INDEX + 4                                                      MM  1978
      I2 = INDEX + 28                                                     MM  1979
      DO 310 J = 1,NUMSTA                                                 MM  1980
      AWDATA(I1,J) = X(J,2)                                               MM  1981
      AWDATA(I2,J) = X(J,1)                                               MM  1982
  310 CONTINUE                                                            MM  1983
C        XMIT TEMPERATURE DATA                                            MM  1984
      DO 320 J = 1,NUMSTA                                                 MM  1985
      CON(INDEX,J,1) = X(J,3)                                             MM  1986
  320 CONTINUE                                                            MM  1987
      IF(NEEDAQ.NE.IYES)  GO TO 350                                       MM  1988
C                                                                         MM  1989
C        REDUCE SOLAR RADIATION DATA                                      MM  1990
C        CURRENTLY ALL SOLAR RADIATION DATA BELOW .05 LANGLEYS/MIN        MM  1991
C        AND ABOVE 2.0 LANGLEYS/MIN IS EXCLUDED.                          MM  1992
C                                                                         MM  1993
      UV(INDEX) = 0.0                                                     MM  1994
      SUM1 = 0.0                                                          MM  1995
      SUM2 = 0.0                                                          MM  1996
      KS = 0                                                              MM  1997
      DO 330 J= 1,NUMSTA                                                  MM  1998
      IF( X(J,7).LT. 0.05)  GO TO 330                                     MM  1999
      IF( X(J,7).GT. 2.00)  GO TO 330                                     MM  2000
      IF( X(J,8).LT. 0.05)  GO TO 330                                     MM  2001
      IF( X(J,8).GT. 2.00)  GO TO 330                                     MM  2002
      SUM1 = SUM1 + X(J,7)                                                MM  2003
      SUM2 = SUM2 + X(J,8)                                                MM  2004
      KS = KS + 1                                                         MM  2005
  330 CONTINUE                                                            MM  2006
      IF(KS.GT.0)  UV(INDEX) = (SUM1 + SUM2)/KS                           MM  2007
C                                                                         MM  2008
C        XMIT TEMPERATURE AND ATMOSPHERIC CONCENTRATION DATA              MM  2009
C                                                                         MM  2010
      DO 340  I = 2,10                                                    MM  2011
      K = I + 9                                                           MM  2012
      IF(I.GE.10)  K = I + 10                                             MM  2013
      DO 340 J = 1,NUMSTA                                                 MM  2014
      CON(INDEX,J,I) = X(J,K)                                             MM  2015
  340 CONTINUE                                                            MM  2016
  350 CONTINUE                                                            MM  2017
C                                                                         MM  2018
C          DETERMINE HOURLY AVERAGE NEAR-SURFACE TEMPERATURE GRADIENTS    MM  2019
C          AND WIND SHEAR CORRECTION FOR 10M ELEVATION WIND SPEEDS.       MM  2020
C                                                                         MM  2021
C          NEUTRAL CONDITIONS   (DEFAULT CONDITION)                       MM  2022
```

```
      ASHR(INDEX) = 1.219                                                MM  2023
      SUM1 = 0.0                                                         MM  2024
      KS = 0                                                             MM  2025
      DO 360 J = 1,NUMSTA                                                MM  2026
      IF(X(J,6) .GT. 10.  )   GO TO  360                                 MM  2027
      IF(X(J,6) .LT.-10.  )   GO TO 360                                  MM  2028
      SUM1 = SUM1 + X(J,6)                                               MM  2029
      KS = KS + 1                                                        MM  2030
  360 CONTINUE                                                           MM  2031
      IF(KS.LT.1)  GO TO 370                                             MM  2032
      DTEMP(INDEX) = SUM1/25./KS                                         MM  2033
C         UNSTABLE CONDITIONS                                            MM  2034
      IF(DTEMP(INDEX).LT. -.015)  ASHR(INDEX) = 1.166                    MM  2035
C         STABLE CONDITIONS                                              MM  2036
      IF(DTEMP(INDEX) .GT. -.005)  ASHR(INDEX) = 1.437                   MM  2037
                                                                         MM  2038
C                                                                        MM  2039
 370  CONTINUE                                                           MM  2040
C                                                                        MM  2041
      IF(KNTER.LT. IMAX)   GO TO 110                                     MM  2042
C                                                                        MM  2043
C        YOU ARE FINISHED READING THE DATA FROM THE TAPE.                MM  2044
C                                                                        MM  2045
C        EDIT WIND DATA AND CONVERT THE WIND DIRECTIONS FROM THE          MM  2046
C        RAMS 360 POINT SYSTEM TO A 90 POINT SYSTEM                      MM  2047
C                                                                        MM  2048
 380  CONTINUE                                                           MM  2049
      KS = 1                                                             MM  2050
      A = APT/360.0                                                      MM  2051
      AINV = 1./A                                                        MM  2052
      DO 390 J = 1,NUMSTA                                                MM  2053
      K = LWDSTA(KS) - 100                                               MM  2054
      DO 390 I = IMIN , IMAX                                             MM  2055
      I1 = I + 4                                                         MM  2056
      I2 = I + 28                                                        MM  2057
      IF(AWDATA(I1,J).GT.HIVAL) AWDATA(I1,J) = - AINV                    MM  2058
      AWDATA(I1,J) = AWDATA(I1,J)*A                                      MM  2059
      IF(AWDATA(I2,J).GT.HIVAL)   AWDATA(I2,J) = -1.0                    MM  2060
C         AND ADJUST 10M WIND SPEEDS TO THE 30M STANDARD                 MM  2061
      IF(J.NE.K)   GO TO 390                                             MM  2062
      AWDATA(I2,J) = AWDATA(I2,J)*ASHR(I)                                MM  2063
      KS = KS + 1                                                        MM  2064
 390  CONTINUE                                                           MM  2065
      IF(NEEDAQ.NE.IYES)   GO TO  440                                    MM  2066
C                                                                        MM  2067
C        EDIT ATMOSHPERIC CONCENTRATION DATA                             MM  2068
C                                                                        MM  2069
      DO 410 K = 2, NOSPEC                                               MM  2070
      DO 410 J = 1, NUMSTA                                               MM  2071
      DO 400 I = IMIN, IMAX                                              MM  2072
      IF(CON(I,J,K).GT.HIVAL)  CON(I,J,K) = -1.0                         MM  2073
 400  CONTINUE                                                           MM  2074
 410  CONTINUE                                                           MM  2075
C                                                                        MM  2076
C        EDIT AND REDUCE TEMPERATURE DATA                                MM  2077
```

```
C                                                                       MM  2078
      DO 430 I = IMIN, IMAX                                             MM  2079
      KS = 0                                                            MM  2080
      SUM1 = 0.                                                         MM  2081
      DO 420 J = 1, NUMSTA                                              MM  2082
      IF(CON(I,J,1).GT.HIVAL)   CON(I,J,1) = -1.0                       MM  2083
      IF(CON(I,J,1).LT.ZERO)  GO TO 420                                 MM  2084
      SUM1 = SUM1 + CON(I,J,1)                                          MM  2085
      KS = KS + 1                                                       MM  2086
  420 CONTINUE                                                          MM  2087
      IF(KS.GT.0)  TEMPSF(I) = SUM1/KS                                  MM  2088
  430 CONTINUE                                                          MM  2089
  440 CONTINUE                                                          MM  2090
C                                                                       MM  2091
C          GENERATE SKY CLEARNESS RATIOS FROM UV DATA                   MM  2092
C                                                                       MM  2093
      CALL SKY(UV,CLOUDY,IDATE)                                         MM  2094
C                                                                       MM  2095
      REWIND LTAPE                                                      MM  2096
C                                                                       MM  2097
      GO TO 480                                                         MM  2098
  450 WRITE(LOUT,60)  NFILE,IYR,IDAY,IHR                                MM  2099
      GO TO 500                                                         MM  2100
                                                                        MM  2101
  460 WRITE(LOUT,70)  IYR,  IDAY,  IHR                                  MM  2102
      GO TO 500                                                         MM  2103
  470 WRITE(LOUT,80)     IYR,  IDAY,  IHR                               MM  2104
      GO TO 500                                                         MM  2105
C                                                                       MM  2106
  480 CONTINUE                                                          MM  2107
C        PRINT WIND DATA IF KPWDAT = YES                                MM  2108
C        PUNCH WIND DATA IF SPCHWD = YES                                MM  2109
      SPCHWD = NO                                                       MM  2110
      IF (KPWDAT.NE.IYES)  GO TO 500.                                   MM  2111
      CALL NEWPAG(TITLE,0,JDATE)                                        MM  2112
      WRITE(LOUT,77)                                                    MM  2113
      DO 490 J = 1,NUMSTA                                               MM  2114
      WRITE(LOUT,78) (NWDATA(I),I=1,28)                                 MM  2115
      WRITE(LOUT,79) (NWDATA(I),I=1,3) , (NWDATA(I),I=29,52)            MM  2116
      IF(SPCHWD.NE.IYES)  GO TO 490                                     MM  2117
      WRITE(LPUNCH,73) (NWDATA(I),I=1,28)                               MM  2118
      WRITE(LPUNCH,74) (NWDATA(I),I=1,3), (NWDATA(I),I=29,52)           MM  2119
  490 CONTINUE                                                          MM  2120
  500 CONTINUE                                                          MM  2121
C                                                                       MM  2122
      IF(DEBUG.NE.YES)  GO TO 530                                       MM  2123
C                                                                       MM  2124
      WRITE(LOUT,82)                                                    MM  2125
      DO 510 J = 1,NUMSTA                                               MM  2126
      WRITE(LOUT,85) J, (NWDATA(I,J),I=1,3), AWDATA(4,J)                MM  2127
      WRITE(LOUT,90)     (AWDATA(I,J), I=5,52)                          MM  2128
  510 CONTINUE                                                          MM  2129
C                                                                       MM  2130
      WRITE(LOUT,84)                                                    MM  2131
      DO 520 K = 1,NOSPEC                                               MM  2132
```

```
      WRITE(LOUT,86)   CONAM(K),  CONAM(K),  CONAM(K), CONAM(K)          MM  2133
      DO 520 J = 1,NUMSTA                                                MM  2134
      WRITE(LOUT,85) J, (NWDATA(I,J),I=1,3)                              MM  2135
      WRITE(LOUT,90) (CON(I,J,K), I=1,24)                                MM  2136
  520 CONTINUE                                                           MM  2137
C                                                                        MM  2138
  530 CONTINUE                                                           MM  2139
      ICASE = 1                                                          MM  2140
      RETURN                                                             MM  2141
C                                                                        MM  2142
      ENTRY EMETIN                                                       MM  2143
      IF(NEEDAQ.NE.IYES) RETURN                                          MM  2144
      IF(ICASE.GT.1)  GO TO 555                                          MM  2145
      CALL NEWPAG(TITLE,0,JDATE)                                         MM  2146
      WRITE(LOUT,89)  IYR,IDAY                                           MM  2147
      WRITE(LOUT,87)                                                     MM  2148
      DO 540 I=1,24                                                      MM  2149
      I1 = (I-1)*100                                                     MM  2150
      I2 = I1 + 59                                                       MM  2151
      WRITE(LOUT,88) I1, I2, UV(I), TEMPSF(I), DTEMP(I), CLOUDY(I)       MM  2152
  540 CONTINUE                                                           MM  2153
C                                                                        MM  2154
C         PUNCH CLEARNESS RATIOS AND TEMPERATURES FOR INPUT TO KEMOD     MM  2155
      IF(KPWIND.NE.IYES) RETURN                                          MM  2156
      A = -10.0                                                          MM  2157
      WRITE(LPUNCH,81)  TITLE                                            MM  2158
      DO 550 I = 6,20                                                    MM  2159
      TM = 60.*FLOAT(I-1)                                                MM  2160
      WRITE(LPUNCH,75)  TM, CLOUDY(I)                                    MM  2161
  550 CONTINUE                                                           MM  2162
      WRITE(LPUNCH,75)  A                                                MM  2163
                                                                         MM  2164
  555 CONTINUE                                                           MM  2165
      DO 560 I =6,20                                                     MM  2166
      TM = 60.*FLOAT(I-1)                                                MM  2167
      WRITE(LPUNCH,76)  TM, TEMPSF(I)                                    MM  2168
  560 CONTINUE                                                           MM  2169
      WRITE(LPUNCH,76)  A                                                MM  2170
      ICASE = ICASE + 1                                                  MM  2171
  580 RETURN                                                             MM  2172
C                                                                        MM  2173
   20 FORMAT (3I4)                                                       MM  2174
   30 FORMAT (10X,4I10)                                                  MM  2175
   40 FORMAT (10E12.6)                                                   MM  2176
   50 FORMAT (1H0,2HV=,I3,4X,10E11.3,/,10X,10E11.3,/,10X, 5E11.3)        MM  2177
   60 FORMAT (1H0, 38HTHE DATE REQUESTED IS NOT IN THE FILE       ,4I8)  MM  2178
   70 FORMAT (1H0, 38HAN UNEXPECTED EOF  --  METIN STOPPED        ,4I8)  MM  2179
   73 FORMAT (I6,1X,2A4,13X,I2,2HPT,24I2)                                MM  2180
   74 FORMAT (I6,1X,2A4,14X,3HM/S/,24I2)                                 MM  2181
   75 FORMAT (20H CLEARNESS RATIOS         ,20X,2F10.3)                  MM  2182
   76 FORMAT (20H TEMPERATURE (DEG C)      ,20X,2F10.3)                  MM  2183
   77 FORMAT (25H0 EDITED WIND DATA                    )                 MM  2184
   78 FORMAT (1H ,I6,1X,2A4,13X,I2,2HPT,24I3)                            MM  2185
   79 FORMAT (1H ,I6,1X,2A4,14X,3HM/S,24I3)                              MM  2186
   80 FORMAT (1H0, 38HPARITY ERROR ENCOUNTERED  -- METIN          ,4I8)  MM  2187
```

```
   81  FORMAT (20A4)                                                          MM  2188
   85  FORMAT (1H0,I2,10X,I6,4X,A4,I4,F10.1 )                                 MM  2189
   82  FORMAT (1H1,22HWIND DATA FROM AWDATA    )                              MM  2190
   84  FORMAT(1H1,47H TEMPERATURE AND ATMOSHPERIC CONCENTRATIONS       )      MM  2191
   86  FORMAT(1H0,4(A4,20X)  )                                                MM  2192
   87  FORMAT (1H0,10X,26HREGIONAL HOURLY AVERAGES     ,// 11X,4HHOUR,       MM  2193
      1  10X,22HULTRAVIOLET RADIATION  ,4X,11HTEMPERATURE  ,4X,              MM  2194
      2  20HTEMPERATURE GRADIENT  ,4X,15HCLEARNESS RATIO  ,                  MM  2195
      3  /,25X,22H   (LANGLEYS/MIN)       ,                                 MM  2196
      5  4X, 11H   (DEG C)     , 9X,14H(DEG C/METER)            )            MM  2197
   88  FORMAT (1H0,8X,I4,3H - ,I4,5X,E15.3,F20.2,F20.4,F20.3 )                MM  2198
   89  FORMAT(1H0,10X,21HST. LOUIS     YEAR =  ,I5,  5X,6HDAY =   ,I3 )      MM  2199
   90  FORMAT (1H ,5X,12E10.2)                                                MM  2200
C                                                                             MM  2201
      END                                                                     MM  2202
```

```
      SUBROUTINE PLACIT(DT,L,VX,VY,POS,NDX)                                   MM  2203
C                                                                             MM  2204
C         STARTING FROM THE POSITION DEFINED BY LOCATION  L  IN THE  P        MM  2205
C         ARRAY,  PLACIT  COMPUTES THE POSITION  POS  AND GRID SQUARE         MM  2206
C         INDICES  NDX  AT TIME  T(L) + DT.  VX  AND  VY  SHOULD BE           MM  2207
C         THE  X  AND  Y  COMPONENTS OF  V(L).                                MM  2208
C                                                                             MM  2209
      DIMENSION POS(2),NDX(2)                                                 MM  2210
      COMMON /WIND/ T(100), V(100), TH(100), NPTS                             MM  2211
      COMMON /TRAJ/ TSTART, P(2,100), ID(2,100)                               MM  2212
      COMMON /GRID/ X1,X2,Y1,Y2,NX,NY,DELX,DELY,DELT                          MM  2213
      DATA SMALL /1.E-4/                                                      MM  2214
      POS(1) = P(1,L)+VX*DT                                                   MM  2215
      POS(2) = P(2,L)+VY*DT                                                   MM  2216
      IF(V(L).EQ.0.) GO TO 10                                                 MM  2217
      YPLUS=0.0
      YPLUS=0.0
      GO to 20
 10   XPLUS=(VX/V(L))*SMALL
      YPLUS=(VY/V(L))*SMALL
 20   X=(POS(1)+ XPLUS)IDELX
      IADD=1
      IF(X.LT.O.)IADD=0
      NDX(1)=IFIX(X)+IADD
      X=(POS(2)+YPLUS)/DELY
      IADD=1
      IF(X.LT.0.)IADD=0
      NDX(2)=IFIX(X)+IADD
      RETURN
      END
```

```
      SUBROUTINE RUFNES(Z0,J)                                                 MM  2225
C                                                                             MM  2226
                                                                              MM  2227
C-----THIS SUBROUTINE CHECKS IF THE LOCATION ALONG THE TRAJECTORY             MM  2228
C     AT EACH HOUR IS WITHIN A PRESPECIFIED DISTANCE FROM THE                 MM  2229
C     CENTER OF THE URBAN AREA OF THE REGION.  IF SO, A SPECIAL               MM  2230
C     URBAN ROUGHNESS LENGTH (Z0) IS USED RATHER THAN A TYPICAL               MM  2231
C     RURAL ROUGHNESS LENGTH.                                                 MM  2232
C                                                                             MM  2233
C     NOTE.........THE UTM X AND Y COORDINATES OF THE CENTER OF               MM  2234
C     THE URBAN AREA (IN KM) AND AN APPROXIMATE RADIAL DISTANCE               MM  2235
C     FROM THIS CENTER MUST BE INPUT IN THE DATA STATEMENT BELOW.             MM  2236
```

```
C                                                                       MM  2237
      COMMON/PEVE/PE(2,24),VE(24)                                       MM  2238
      COMMON/ORIGIN/UTMXOR,UTMYOR                                       MM  2239
C                                                                       MM  2240
      DATA UTMXUR,  UTMYUR  / 739.5, 4280.5/                            MM  2241
      DATA RADURB, RADSUB  /5.,10./                                     MM  2242
      DATA ZOURBN, ZORURL  /1.,.25/                                     MM  2243
C                                                                       MM  2244
      PE(1,J)=PE(1,J)+UTMXOR                                            MM  2245
      PE(2,J)=PE(2,J)+UTMYOR                                            MM  2246
      A=(PE(1,J)-UTMXUR)**2                                             MM  2247
      B=(PE(2,J)-UTMYUR)**2                                             MM  2248
      A = SQRT(A+B)                                                     MM  2249
      IF(A.GE.RADSUB)  GO TO 40                                         MM  2250
      IF(A.LE.RADURB)  GO TO 30                                         MM  2251
      PORT = (A - RADURB)/(RADSUB - RADURB)                             MM  2252
      ZO = ZOURBN + PORT*(ZORURL-ZOURBN)                                MM  2253
      GO TO 50                                                          MM  2254
   30 ZO = ZOURBN                                                       MM  2255
      GO TO 50                                                          MM  2256
   40 ZO = ZORURL                                                       MM  2257
   50 RETURN                                                            MM  2258
      END                                                               MM  2259



      SUBROUTINE SETIN                                                  MM  2260
C                                                                       MM  2261
C       SETIN FOR ST. LOUIS                                             MM  2262
C                                                                       MM  2263
C     SETIN INITIALIZES WIND STATION NAMES AND COORDINATES.             MM  2264
C     SETIN INITIALIZES LATITUDE, LONGITUDE, AND TIME ZONE.             MM  2265
C                                                                       MM  2266
C     ISTANS IS AN ARRAY CONTAINING THE NAMES OF THE WIND STATIONS      MM  2267
C       SSTAN IS AN ARRAY CONTAINING THE UTM X-Y COORDINATES OF THE     MM  2268
C       METEROLOGICAL STATIONS                                          MM  2269
C                                                                       MM  2270
C     NUMSTA IS THE NUMBER OF STATIONS                                  MM  2271
C     RLAT IS THE LATITUDE                                              MM  2272
C     RLONG IS THE LONGITUDE                                            MM  2273
C     TMZONE IS THE TIME ZONE (IE                                       MM  2274
C                                    6.   FOR ST. LOUIS                 MM  2275
C                                    8.   FOR LOS ANGELES )             MM  2276
C                                                                       MM  2277
C                                                                       MM  2278
      COMMON /WDATA/ NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX   MM  2279
      COMMON  /GRID/  X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT          MM  2280
      COMMON  /ORIGIN/  UTMXOR, UTMYOR                                  MM  2281
      COMMON /WHERE/  RLAT,    RLONG,     TMZONE                       MM  2282
C                                                                       MM  2283
C         ------------------------------------------------------------  MM  2284
C                                                                       MM  2285
C            USER MUST DEFINE THE FOLLOWING INFORMATION                 MM  2286
C                                                                       MM  2287
      DATA RLAT /38.6/                                                  MM  2288
```

```
      DATA RLONG /90.2/                                                 MM  2289
                                                                        MM  2290
      DATA TMZONE /6.0/                                                 MM  2291
      DATA NUMSTA /25/                                                  MM  2292
C                                                                       MM  2293
      DATA ISTANS  /4HRAMS,   4H 101,                                   MM  2294
     1              4HRAMS,   4H 102,   4HRAMS,   4H 103,              MM  2295
     2              4HRAMS,   4H 104,   4HRAMS,   4H 105,              MM  2296
     3              4HRAMS,   4H 106,   4HRAMS,   4H 107,              MM  2297
     4              4HRAMS,   4H 108,   4HRAMS,   4H 109,              MM  2298
     5              4HRAMS,   4H 110,   4HRAMS,   4H 111,              MM  2299
     6              4HRAMS,   4H 112,   4HRAMS,   4H 113,              MM  2300
     7              4HRAMS,   4H 114,   4HRAMS,   4H 115,              MM  2301
     8              4HRAMS,   4H 116,   4HRAMS,   4H 117,              MM  2302
     9              4HRAMS,   4H 118,   4HRAMS,   4H 119,              MM  2303
     *              4HRAMS,   4H 120,   4HRAMS,   4H 121,              MM  2304
     1              4HRAMS,   4H 122,   4HRAMS,   4H 123,              MM  2305
     2              4HRAMS,   4H 124,   4HRAMS,   4H 125  /            MM  2306
C                                                                       MM  2307
C            STAT     UTM-X      UTM-Y                                 MM  2308
C                                                                       MM  2309
      SSTAN(1, 1) =  744.183                                            MM  2310
      SSTAN(2, 1) =            4279.862                                 MM  2311
      SSTAN(1, 2) =  742.518                                            MM  2312
      SSTAN(2, 2) =            4286.045                                 MM  2313
      SSTAN(1, 3) =  747.588                                            MM  2314
      SSTAN(2, 3) =            4282.467                                 MM  2315
      SSTAN(1, 4) =  747.312                                            MM  2316
      SSTAN(2, 4) =            4277.304                                 MM  2317
      SSTAN(1, 5) =  743.706                                            MM  2318
      SSTAN(2, 5) =            4276.453                                 MM  2319
      SSTAN(1, 6) =  738.660                                            MM  2320
      SSTAN(2, 6) =            4277.566                                 MM  2321
      SSTAN(1, 7) =  740.179                                            MM  2322
      SSTAN(2, 7) =            4282.610                                 MM  2323
      SSTAN(1, 8) =  748.407                                            MM  2324
      SSTAN(2, 8) =            4291.102                                 MM  2325
      SSTAN(1, 9) =  755.802                                            MM  2326
      SSTAN(2, 9) =            4279.886                                 MM  2327
      SSTAN(1,10) =  747.209                                            MM  2328
      SSTAN(2,10) =            4272.826                                 MM  2329
      SSTAN(1,11) =  738.812                                            MM  2330
      SSTAN(2,11) =            4272.479                                 MM  2331
      SSTAN(1,12) =  733.938                                            MM  2332
      SSTAN(2,12) =            4280.913                                 MM  2333
      SSTAN(1,13) =  737.738                                            MM  2334
      SSTAN(2,13) =            4289.820                                 MM  2335
      SSTAN(1,14) =  744.320                                            MM  2336
      SSTAN(2,14) =            4297.456                                 MM  2337
      SSTAN(1,15) =  757.111                                            MM  2338
      SSTAN(2,15) =            4297.799                                 MM  2339
      SSTAN(1,16) =  762.777                                            MM  2340
      SSTAN(2,16) =            4290.083                                 MM  2341
      SSTAN(1,17) =  760.560                                            MM  2342
      SSTAN(2,17) =            4272.818                                 MM  2343
```

```
      SSTAN(1,18) =  743.065                                                MM  2344
      SSTAN(2,18) =           4263.256                                      MM  2345
      SSTAN(1,19) =  729.759                                                MM  2346
      SSTAN(2,19) =           4270.547                                      MM  2347
      SSTAN(1,20) =  723.079                                                MM  2348
      SSTAN(2,20) =           4285.909                                      MM  2349
      SSTAN(1,21) =  732.414                                                MM  2350
      SSTAN(2,21) =           4302.376                                      MM  2351
      SSTAN(1,22) =  741.631                                                MM  2352
                                                                            MM  2353
      SSTAN(2,22) =           4329.223                                      MM  2354
      SSTAN(1,23) =  777.320                                                MM  2355
      SSTAN(2,23) =           4286.378                                      MM  2356
      SSTAN(1,24) =  749.275                                                MM  2357
      SSTAN(2,24) =           4236.537                                      MM  2358
      SSTAN(1,25) =  697.445                                                MM  2359
      SSTAN(2,25) =           4282.240                                      MM  2360
C                                                                           MM  2361
C         ------------------------------------------------------------      MM  2362
C                                                                           MM  2363
      DO 40 I = 1,NUMSTA                                                    MM  2364
      SSTAN(1,I) = SSTAN(1,I) - UTMXOR                                      MM  2365
      SSTAN(2,I) = SSTAN(2,I) - UTMYOR                                      MM  2366
C                                                                           MM  2367
40    CONTINUE                                                              MM  2368
      X1 = 0.0                                                              MM  2369
      X2 = NX*DELX                                                          MM  2370
      Y1 = 0.0                                                              MM  2371
      Y2 = NY*DELY                                                          MM  2372
      RETURN                                                                MM  2373
      END                                                                   MM  2374



      SUBROUTINE SETTUP                                                     MM  2375
C                                                                           MM  2376
C         INITIALIZATION SUBROUTINE FOR GENERATION OF                       MM  2377
C         TRAJECTORY GRID SQUARE HISTORIES.                                 MM  2378
C                                                                           MM  2379
      DIMENSION        UTM(2,5)                                             MM  2380
      COMMON /CNTROL/  KSTOP,  TSUN                                         MM  2381
      COMMON /GRID/    X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT             MM  2382
      COMMON /INPUTS/  TITLE(20),   JDATE(10),    NCURV                    MM  2383
      COMMON /TRAJ/    TSTART,  P(2,100),   ID(2,100)                       MM  2384
      COMMON /WDATA/   NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX    MM  2385
      COMMON /WIND/    T(100),  V(100),  TH(100),  NPTS                     MM  2386
      COMMON /ORIGIN/  UTMXOR,    UTMYOR                                    MM  2387
      DATA LIN, LOUT /3,6/                                                  MM  2388
C                                                                           MM  2389
C         CONVERT WIND VELOCITIES TO KM/MIN AND TIMES TO MINUTES            MM  2390
C                                                                           MM  2391
      V(1) = V(1)/60.                                                       MM  2392
      TSTART = T(1) + TSUN                                                  MM  2393
      TSHIFT = TIMIN(T(1))                                                  MM  2394
      T(1) = 0.000                                                          MM  2395
```

```
      DO 100 K = 2,NPTS                                                 MM  2396
      V(K) = V(K)/60.0                                                  MM  2397
      T(K) = TIMIN(T(K)) - TSHIFT                                       MM  2398
      IF(T(K).LT.T(K-1))  T(K) = T(K) + 1440.0                          MM  2399
  100 CONTINUE                                                          MM  2400
C                                                                       MM  2401
C         SET UP GRID AND LOCATE TRAJECTORY START POINT                 MM  2402
      X1 = 0.0                                                          MM  2403
      X2 = NX*DELX                                                      MM  2404
      Y1 = 0.0                                                          MM  2405
      Y2 = NY*DELY                                                      MM  2406
C         NORTH-WEST CORNER                                             MM  2407
      UTM(1,1) = UTMXOR                                                 MM  2408
      UTM(2,1) = UTMYOR + Y2                                            MM  2409
C         NORTH-EAST CORNER                                             MM  2410
      UTM(1,2) = UTMXOR + X2                                            MM  2411
      UTM(2,2) = UTMYOR + Y2                                            MM  2412
C         SOUTH-EAST CORNER                                             MM  2413
      UTM(1,3) = UTMXOR + X2                                            MM  2414
      UTM(2,3) = UTMYOR                                                 MM  2415
                                                                        MM  2416
C         SOUTH-WEST CORNER                                             MM  2417
      UTM(1,4) = UTMXOR                                                 MM  2418
      UTM(2,4) = UTMYOR                                                 MM  2419
C         START POINT                                                   MM  2420
      UTM(1,5) = UTMXOR + P(1,1)                                        MM  2421
      UTM(2,5) = UTMYOR + P(2,1)                                        MM  2422
      CALL PLACIT (0.,1,0.,0.,UTM(1,5),JDATE(9))
      ID(1,1) = JDATE(9)
      ID(2,1) = JDATE(10)
                                                                        MM  2425
C         PRINT GRID CORNER AND START POINT COORDINATES                 MM  2426
C                                                                       MM  2427
      WRITE(LOUT,3)                                                     MM  2428
      WRITE(LOUT,4)       UTM(1,1), UTM(2,1), X1, Y2                    MM  2429
      WRITE(LOUT,5)       UTM(1,2), UTM(2,2), X2, Y2                    MM  2430
      WRITE(LOUT,6)       UTM(1,3), UTM(2,3), X2, Y1                    MM  2431
      WRITE(LOUT,7)       UTM(1,4), UTM(2,4), X1, Y1                    MM  2432
      WRITE(LOUT,8)       UTM(1,5), UTM(2,5), P(1,1), P(2,1)            MM  2433
C                                                                       MM  2434
C         INITIALIZE PRINTER-PLOT                                       MM  2435
      READ(LIN,1) XL ,XR, YB, YT                                        MM  2436
      CALL SETPLT(XL,XR,YB,YT)                                          MM  2437
      RETURN                                                            MM  2438
C                                                                       MM  2439
C         FORMATS                                                       MM  2440
1     FORMAT(30X,4F10.0)                                                MM  2441
3     FORMAT  (1H0,/////,21H  GRID REFERENCE DATA  ,//, 5X,             MM  2442
     1  12H(KILOMETERS)  ,15X,15HUTM COORDINATES ,8X,17HLOCAL COORDINATEMM  2443
     2S  ,/,34X,1HX,9X,1HY,13X,1HX,9X,1HY )                             MM  2444
4     FORMAT (1H0,4X,17HNORTH WEST CORNER  ,6X,2F10.2,4X,2F10.2 )       MM  2445
5     FORMAT (1H0,4X,17HNORTH EAST CORNER  ,6X,2F10.2,4X,2F10.2 )       MM  2446
6     FORMAT (1H0,4X,17HSOUTH EAST CORNER  ,6X,2F10.2,4X,2F10.2 )       MM  2447
7     FORMAT (1H0,4X,17HSOUTH WEST CORNER  ,6X,2F10.2,4X,2F10.2 )       MM  2448
8     FORMAT (1H0,4X,17HSTART POINT        ,6X,2F10.2,4X,2F10.2 )       MM  2449
      END                                                               MM  2450
```

```
      SUBROUTINE SKY (UV,CLOUDY,IDATE)                                  MM  2451
C                                                                       MM  2452
C     GENERATE SKY CLEARNESS RATIOS                                     MM  2453
C     ST. LOUIS VERSION   12.22.77                                      MM  2454
C     CONVRT IS SET TO CONVERT WATTS/METER**2 TO LANGLEYS/MIN.          MM  2455
C                                                                       MM  2456
      DIMENSION        UV(1),  CLOUDY(1),  IDATE(1)                     MM  2457
      COMMON /WHERE/   RLAT,  RLONG,  TMZONE                            MM  2458
C                                                                       MM  2459
C                                                                       MM  2460
      IHALF = 30                                                        MM  2461
      CONVT = 1./697.333334                                             MM  2462
      IY = IDATE(1)/10000                                               MM  2463
      IM = IDATE(1)/100 - IY*100                                        MM  2464
      ID = IDATE(1) - IY*10000  - IM*100                                MM  2465
      IY = IY + 1900                                                    MM  2466
      DO 10 I =1,24                                                     MM  2467
      IF(UV(I).GT. 1.E-4)  GO TO 15                                     MM  2468
      CLOUDY(I) = 1.0                                                   MM  2469
 10   CONTINUE                                                          MM  2470
      GO TO 50                                                          MM  2471
 15   CONTINUE                                                          MM  2472
      IHR = (I-1)*100 + IHALF                                           MM  2473
      TIME = FLOAT(IHR)                                                 MM  2474
      DO 40 J = I,24                                                    MM  2475
      IF(UV(J).LT. 1.E-4)  GO TO 30                                     MM  2476
      CALL SOLAR(RLAT,RLONG,TMZONE,IY,IM,ID,TIME,D,5)                   MM  2477
      Z = 90.0 - D                                                      MM  2478
                                                                        MM  2479
      IF(Z.GT.30.)  GO TO 20                                            MM  2480
      CLEAR = (-.0146667*Z*Z + .05833334*Z + 66)*CONVT                  MM  2481
      GO TO 25                                                          MM  2482
 20   CLEAR = (-.992727*Z + 84.4)*CONVT                                 MM  2483
 25   CLOUDY(J) = UV(J)/CLEAR                                           MM  2484
      IF(CLOUDY(J).GT. 1.)  CLOUDY(J) = 1.0000                          MM  2485
      TIME = TIME + 100                                                 MM  2486
      GO TO 40                                                          MM  2487
 30   CONTINUE                                                          MM  2488
      CLOUDY(J) = 1.000                                                 MM  2489
 40   CONTINUE                                                          MM  2490
 50   RETURN                                                            MM  2491
      END                                                               MM  2492
```

```
      SUBROUTINE SMOOTH(LMAX)                                           MM  2493
C                                                                       MM  2494
C-----THIS SUBROUTINE SMOOTHS THE TEMPERATURE SOUNDING BY DIVIDING THE  MM  2495
C     SOUNDING INTO LAYERS WITH LAPSE RATES WHICH ARE GT 0.98, LT 0.98  MM  2496
C     BUT GT 0.0, AND LE 0.0, AND THEN DETERMINING THE AVERAGE LAPSE    MM  2497
C     RATE OVER EACH LAYER.                                             MM  2498
C                                                                       MM  2499
      COMMON/TEMPHT/Z(50),ZZ(50),T(50),TT(50)                           MM  2500
      EQUIVALENCE(JMAXM1,LMAXP1,LMAXM2)                                 MM  2501
      DIMENSION DELTAT(49)                                              MM  2502
```

```
      LMAX=LMAX-1                                                       MM  2503
      JMAX=LMAX                                                         MM  2504
      M=1                                                               MM  2505
      DO 1050 L=1,LMAX                                                  MM  2506
      DELTAT(L)=(TT(L)-T(L))*100./(ZZ(L)-Z(L))+.98                      MM  2507
      IF (L.EQ.1) GO TO 1050                                            MM  2508
      IF(DELTAT(L))     1025,1025,1030                                  MM  2509
 1020 M=M+1                                                             MM  2510
      GO TO 1035                                                        MM  2511
 1025 IF(DELTAT(L-1).GT.0.)   GO TO 1020                                MM  2512
      GO TO 1035                                                        MM  2513
 1030 IF(DELTAT(L-1).LE.0.) GO TO 1020                                  MM  2514
      IF((DELTAT(L).GE..98.AND.DELTAT(L-1).LT..98).OR.                  MM  2515
     +(DELTAT(L).LT..98.AND.DELTAT(L-1).GE..98)) GO TO 1020             MM  2516
 1035 TT(M)=TT(L)                                                       MM  2517
      ZZ(M)=ZZ(L)                                                       MM  2518
 1050 CONTINUE                                                          MM  2519
      LMAX=M+1                                                          MM  2520
      NO=JMAX-M                                                         MM  2521
      CALL XMIT(-NO,0.,TT(LMAX))                                        MM  2522
      CALL XMIT(-NO,0.,ZZ(LMAX))                                        MM  2523
      JMAXM1=JMAX-1                                                     MM  2524
      CALL XMIT(JMAXM1,TT(1),T(2))                                      MM  2525
      CALL XMIT(JMAXM1,ZZ(1),Z(2))                                      MM  2526
      JMAX=LMAX                                                         MM  2527
      M=3                                                               MM  2528
      LMAXM2=LMAX-2                                                     MM  2529
      IF(LMAXM2.LT.3)       RETURN                                      MM  2530
      DO 1100 L=3,LMAXM2                                                MM  2531
      DELTAT(L)=(TT(L)-TT(L-1))*100./(ZZ(L)-ZZ(L-1))+.98                MM  2532
      IF(DELTAT(L).LT.0..AND.(ZZ(L)-ZZ(L-1)).LT.100.) GO TO 1080        MM  2533
      M=M+1                                                             MM  2534
      GO TO 1100                                                        MM  2535
 1080 NO=LMAX-1-L                                                       MM  2536
      CALL XMIT(NO,TT(L+1),T(M+1))                                      MM  2537
      CALL XMIT(NO,ZZ(L+1),Z(M+1))                                      MM  2538
 1100 CONTINUE                                                          MM  2539
      LMAX=M+1                                                          MM  2540
      NO=JMAX-LMAX                                                      MM  2541
                                                                        MM  2542
      LMAXP1=LMAX+1                                                     MM  2543
      CALL XMIT(-NO,0.,T(LMAXP1))                                       MM  2544
      CALL XMIT(-NO,0.,T(LMAXP1))                                       MM  2545
      JMAXM1=JMAX-1                                                     MM  2546
      CALL XMIT(JMAXM1,T(2),TT(1))                                      MM  2547
      CALL XMIT(JMAXM1,Z(2),ZZ(1))                                      MM  2548
      RETURN                                                            MM  2549
      END                                                               MM  2550
```

```
      SUBROUTINE STABIL(STAB,J,A,ZTOP,DUDZ,DELT,U,ZHTMIX,USTAR,ZW,Z0)   MM  2551
C-----THIS SUBROUTINE CALCULATES THE VALUE OF THE STABILITY PARAMETER A MM  2552
C     DEVELOPED BY FULLE(1975), AND THEN DETERMINES THE CORRESPONDING   MM  2553
C     VALUE OF THE MONIN-OBUKHOV LENGTH, L.  FIRST, AN APPROPRIATE      MM  2554
C     STABILITY CLASS IS SELECTED ACCORDING TO THE VALUE OF A, AND THEN MM  2555
C     A VALUE OF L IS DETERMINED AS SUGGESTED BY GOLDER(1972).          MM  2556
C     IN ADDITION, THE HEIGHT OF THE BOUNDARY LAYER IS DETERMINED FOR   MM  2557
C     STABLE, NEUTRAL, AND UNSTABLE CONDITIONS AS A FUNCTION OF U*/F.   MM  2558
C                                                                       MM  2559
      COMMON/CORIOL/F                                                   MM  2560
      REAL LASTAB                                                       MM  2561
      DUMB=DUDZ                                                         MM  2562
      U60=U*60.                                                         MM  2563
      IF(J.EQ.1) DUDZ=U/(ZW-Z0)                                         MM  2564
  100 LASTAB=STAB                                                       MM  2565
      DTDZ=DELT                                                         MM  2566
      IF(DTDZ.LT.-5.) DTDZ=-5.                                          MM  2567
      A=1./(10.+DTDZ)**2+.0025*DUDZ                                     MM  2568
      A=1./SQRT(A)                                                      MM  2569
  107 STAB=1./FULGOL(A,Z0)                                              MM  2570
      IF(J.EQ.1) GO TO 108                                              MM  2571
      CHANST=ABS((LASTAB-STAB)/STAB)                                    MM  2572
      IF(CHANST.LE..01) GO TO 108                                       MM  2573
      ZOL=ZTOP/STAB                                                     MM  2574
      IF(STAB.LT.0.) DUDZ=DUMB*(1.-15.*ZOL)**-.25                       MM  2575
      IF(STAB.GT.0.) DUDZ=DUMB*(1.+4.7*ZOL)                             MM  2576
      GO TO 100                                                         MM  2577
  108 IF(STAB.LT.0.) GO TO 110                                          MM  2578
      USTAR=0.35/(ALOG(ZW/Z0)+4.7/STAB*(ZW-Z0))*U60                     MM  2579
      USTOR=USTAR/60.                                                   MM  2580
      ZHTMIX=0.5*SQRT(USTOR*STAB/F)                                     MM  2581
      ZHTMAX=0.5*USTOR/F                                                MM  2582
      IF(ZHTMIX.GT.ZHTMAX) ZHTMIX=ZHTMAX                                MM  2583
      GO TO 115                                                         MM  2584
  110 ZWPHI=(1.-15.*ZW/STAB)**.25                                       MM  2585
      Z0PHI=(1.-15.*Z0/STAB)**.25                                       MM  2586
      USTAR=0.35/(ALOG(ZW/Z0)+ALOG(((Z0PHI**2+1.)*(Z0PHI+1.)**2)/       MM  2587
     1((ZWPHI**2+1.)*(ZWPHI+1.)**2))+2.*(ATAN(ZWPHI)-ATAN(Z0PHI)))*U60  MM  2588
      USTOR=USTAR/60.                                                   MM  2589
      ZHTMIN=0.5*USTOR/F                                                MM  2590
      ZHTMIX=0.5*(USTOR/F)**1.5/SQRT(-STAB)                             MM  2591
      IF(ZHTMIX.GT.4000.) ZHTMIX=4000.                                  MM  2592
      IF((ZHTMIX.LT.ZHTMIN).OR.(J.GT.1)) ZHTMIX=ZHTMIN                  MM  2593
  115 RETURN                                                            MM  2594
      END                                                               MM  2595
```

```
      SUBROUTINE STADIF(ZW,Z0,USFC,ZHIGH,DH,AKMIN,ISKLAS)                MM  2596
C                                                                        MM  2597
C-----THIS SUBROUTINE CONTROLS THE CALCULATION OF THE STABILITY          MM  2598
C     AND THE DIFFUSIVITIES FOR EACH HOURLY UPDATE                       MM  2599
C                                                                        MM  2600
      COMMON/DIFDAT/DELTAT(30),STAB(30),ZMIX(30),USTAR(30),              MM  2601
     +             UZZ(201),AKZ(201)                                     MM  2602
      COMMON/TEMPHT/Z(50),ZZ(50),T(50),TT(50)                            MM  2603
      DATA LOUT, YES, YNO /6,4HYES ,4HNO  /                              MM  2604
                                                                         MM  2605
      DBUG=YNO                                                           MM  2606
      J=0                                                                MM  2607
      N=1                                                                MM  2608
  202 J=J+1                                                              MM  2609
      STAB(J)=-1.E6                                                      MM  2610
      DELT=DELTAT(J)                                                     MM  2611
      UZZ(J+1)=USFC*ALOG(ZZ(J)/Z0)/ALOG(ZW/Z0)                           MM  2612
      DDZ=ABS(UZZ(J+1)-UZZ(J))/(ZZ(J)-Z(J))                              MM  2613
      ZTOP=ZZ(J)                                                         MM  2614
      CALL STABIL(STAB(J),J,A,ZTOP,DDZ,DELT,USFC,ZMIX(J),USTAR(J),ZW,Z0)MM  2615
      IF(J.EQ.1) ISKLAS = KLASS(A)                                       MM  2616
      USTORE=USTAR(J)/60.                                                MM  2617
  207 ZHT=Z(J)                                                           MM  2618
  208 ZHT=ZHT+DH                                                         MM  2619
      N=N+1                                                              MM  2620
      CALL DIFFUS(STAB(J),ZHT,AKZ(N),USTAR(J))                           MM  2621
      IF(AKZ(N).LT.AKMIN) AKZ(N)=AKMIN                                   MM  2622
      IF(ZHT.LT.ZZ(J)) GO TO 208                                         MM  2623
      IF(DBUG.EQ.YES) WRITE(LOUT,9) Z(J),ZZ(J),DELT,DDZ,A,STAB(J),       MM  2624
     1    USTORE,ZMIX(J)                                                 MM  2625
      IF(ZZ(J).EQ.ZHIGH) GO TO 29                                        MM  2626
      GO TO 202                                                          MM  2627
   29 RETURN                                                             MM  2628
    9 FORMAT(1H ,F4.0,1H-,F4.0,F10.3,3X,F10.7,F10.3,F15.1,F10.2,F10.1)   MM  2629
      END                                                                MM  2630



      FUNCTION TIMIN(T)                                                  MM  2631
C                                                                        MM  2632
C         TIMIN CONVERTS MILITARY CLOCK TIME TO MINUTES                  MM  2633
C                                                                        MM  2634
      XMIN = AMOD(T,100.)                                                MM  2635
      HRS = (T-XMIN)/100.                                                MM  2636
      TIMIN = 60.*HRS+XMIN                                               MM  2637
      RETURN                                                             MM  2638
      END                                                                MM  2639
```

```
      FUNCTION UNIDOT(A,B)                                                  MM  2640
C                                                                           MM  2641
C         UNIDOT RETURNS THE DOT PRODUCT OF A UNIT VECTOR PARALLEL          MM  2642
C         TO  A  WITH A UNIT VECTOR PARALLEL TO  B,  I.E. THE COSINE        MM  2643
C         OF THE ACUTE ANGLE SEPARATING  A  AND  B.                         MM  2644
C                                                                           MM  2645
      DIMENSION A(3), B(3)                                                  MM  2646
      DENOM = SQRT((A(1)**2+A(2)**2+A(3)**2)*(B(1)**2+B(2)**2+B(3)**2))     MM  2647
      UNIDOT = 0.                                                           MM  2648
      IF (DENOM.EQ.0.) RETURN                                               MM  2649
      DO 100 I = 1,3                                                        MM  2650
      UNIDOT = UNIDOT+A(I)*B(I)                                             MM  2651
100   CONTINUE                                                              MM  2652
      UNIDOT = UNIDOT/DENOM                                                 MM  2653
      RETURN                                                                MM  2654
      END                                                                   MM  2655



      SUBROUTINE UNITV(A,B)                                                 MM  2656
C                                                                           MM  2657
C RETURNS B THE UNIT VECTOR PARALLEL TO VECTOR A                            MM  2658
C                                                                           MM  2659
      DIMENSION A(3),B(3)                                                   MM  2660
C                                                                           MM  2661
      X = SQRT(A(1)**2+A(2)**2+A(3)**2)                                     MM  2662
      DO 100 I = 1,3                                                        MM  2663
      B(I) = A(I)/X                                                         MM  2664
100   CONTINUE                                                              MM  2665
      RETURN                                                                MM  2666
      END                                                                   MM  2667



                                                                            MM  2668
      SUBROUTINE UXYPOS                                                     MM  2669
C                                                                           MM  2670
C-----THIS SUBROUTINE DETERMINES THE WIND SPEED AND X,Y COORDINATES         MM  2671
C     (LOCAL ORIGIN) AT EACH HOUR FROM THE TRAJECTORY VECTOR.               MM  2672
C                                                                           MM  2673
      COMMON/WIND/TIME(100),V(100),TH(100),NPTS                             MM  2674
      COMMON/TRAJ/TSTART,P(2,100),ID(2,100)                                 MM  2675
      COMMON/PEVE/PE(2,24),VE(24)                                           MM  2676
C                                                                           MM  2677
      TE=AINT(TIME(1)*0.01)+1.                                              MM  2678
      J=IFIX(TE)                                                            MM  2679
      PE(1,J)=P(1,1)                                                        MM  2680
      PE(2,J)=P(2,1)                                                        MM  2681
      VE(J)=V(1)                                                            MM  2682
C                                                                           MM  2683
      DO 50 I=2,NPTS                                                        MM  2684
      TI=DTIME(TIME(I))                                                     MM  2685
    5 IF(TI-TE) 50,20,30                                                    MM  2686
   20 J=J+1                                                                 MM  2687
      IF(J.GT.24) RETURN                                                    MM  2688
```

```
      PE(1,J)=P(1,I)                                                    MM  2689
      PE(2,J)=P(2,I)                                                    MM  2690
      VE(J)=V(I)                                                        MM  2691
      TE=TE+1.                                                          MM  2692
      GO TO 50                                                          MM  2693
   30 J=J+1                                                             MM  2694
      IF(J.GT.24) RETURN                                                MM  2695
      TK=DTIME(TIME(I-1))                                               MM  2696
      TFACTR=(TI-TE)/(TI-TK)                                            MM  2697
      PE(1,J)=P(1,I)-(P(1,I)-P(1,I-1))*TFACTR                           MM  2698
      PE(2,J)=P(2,I)-(P(2,I)-P(2,I-1))*TFACTR                           MM  2699
      VE(J)=V(I)-(V(I)-V(I-1))*TFACTR                                   MM  2700
      TE=TE+1.                                                          MM  2701
      GO TO 5                                                           MM  2702
   50 CONTINUE                                                          MM  2703
      RETURN                                                            MM  2704
      END                                                               MM  2705



      SUBROUTINE WINDRD(LUWND,KPWDAT,TITLE,JDATE)                       MM  2706
C                                                                       MM  2707
C     THIS SUBROUTINE READS WIND DATA AND PUTS IT ONTO AS MANY AS  NFILEMM  2708
C     SEPARATE FILES BY DATE.  THE DATES ARE NOT ORDERED.               MM  2709
C                                                                       MM  2710
      DIMENSION TITLE(20), JDATE(2)                                     MM  2711
      DIMENSION PT(24),XIMPH(24),IQ(24),JQ(24)                          MM  2712
      DIMENSION NAM(2)                                                  MM  2713
      DIMENSION IDATE(10), ICT(10)                                      MM  2714
      COMMON /DATES/ IDATE, ICT, NFILES                                 MM  2715
      DATA IBLNK/2H  /,NONO/4HNO  /                                     MM  2716
      DATA LOUT /6/                                                     MM  2717
C                                                                       MM  2718
      NFILES = 10                                                       MM  2719
C************************    ST. LOUIS ONLY   ****************************MM  2720
C                                                                       MM  2721
C     GET THE ST. LOUIS WIND DATA IN SUBROUTINE *** METIN ***           MM  2722
C                                                                       MM  2723
C     ALL CARDS BETWEEN THE LINES SHOULD BE REMOVED FOR OTHER CITIES.   MM  2724
C                                                                       MM  2725
      RETURN                                                            MM  2726
C                                                                       MM  2727
C************************    ST. LOUIS ONLY   ****************************MM  2728
C                                                                       MM  2729
90    ISET = 0                                                          MM  2730
                                                                        MM  2731
      IOLDD = 0                                                         MM  2732
      IFILE = 11                                                        MM  2733
      ICOLD = 0                                                         MM  2734
      IF (KPWDAT.EQ.NONO) GO TO 100                                     MM  2735
      CALL NEWPAG(TITLE,0,JDATE)                                        MM  2736
      WRITE (LOUT,1)                                                    MM  2737
      L = 0                                                             MM  2738
C                                                                       MM  2739
100   CONTINUE                                                          MM  2740
```

```
      READ (LUWND,2) NWDTE,NAM,APT,PT                                   MM  2741
      IF (NWDTE.EQ.999999) GO TO 200                                    MM  2742
      BACKSPACE LUWND                                                   MM  2743
      READ (LUWND,3) IQ                                                 MM  2744
      READ (LUWND,4) XIMPH                                              MM  2745
      BACKSPACE LUWND                                                   MM  2746
      READ (LUWND,3) JQ                                                 MM  2747
      IF (KPWDAT.EQ.NONO) GO TO 120                                     MM  2748
      L = L+1                                                           MM  2749
      IF (MOD(L,51).NE.0) GO TO 110                                     MM  2750
      CALL NEWPAG(TITLE,0,JDATE)                                        MM  2751
      WRITE (LOUT,1)                                                    MM  2752
110   CONTINUE                                                          MM  2753
      WRITE (LOUT,5) NWDTE,NAM,APT,PT                                   MM  2754
      WRITE (LOUT,6) XIMPH                                              MM  2755
120   CONTINUE                                                          MM  2756
C                                                                       MM  2757
C         REPLACE MISSING DATA WITH NEGATIVE INPUTS                     MM  2758
C                                                                       MM  2759
      DO 130 K = 1,24                                                   MM  2760
      IF (IQ(K).EQ.IBLNK) PT(K) = -1.                                   MM  2761
      IF (JQ(K).EQ.IBLNK) XIMPH(K) = -1.                                MM  2762
130   CONTINUE                                                          MM  2763
C                                                                       MM  2764
      IF (NWDTE.EQ.IOLDD) GO TO 170                                     MM  2765
      IOLDD = NWDTE                                                     MM  2766
      IF (ICOLD.EQ.0) GO TO 150                                         MM  2767
      DO 140 I = 1,ICOLD                                                MM  2768
      IF (IDATE(I).EQ.NWDTE) GO TO 160                                  MM  2769
140   CONTINUE                                                          MM  2770
      IF (ICOLD.EQ.NFILES) GO TO 180                                    MM  2771
150   CONTINUE                                                          MM  2772
      ICOLD = ICOLD+1                                                   MM  2773
      IDATE(ICOLD) = NWDTE                                              MM  2774
      I = ICOLD                                                         MM  2775
      ICT(ICOLD) = 0                                                    MM  2776
      IFILED = IFILE + I                                                MM  2777
      REWIND IFILED                                                     MM  2778
      GO TO 170                                                         MM  2779
C                                                                       MM  2780
160   CONTINUE                                                          MM  2781
      IFILED = IFILE+I                                                  MM  2782
170   CONTINUE                                                          MM  2783
      WRITE (IFILED) NWDTE,NAM,APT,PT,XIMPH                             MM  2784
      ICT(IFILED-IFILE) = ICT(IFILED-IFILE) + 1                         MM  2785
      GO TO 100                                                         MM  2786
C                                                                       MM  2787
180   CONTINUE                                                          MM  2788
      IF (ISET.GT.0) GO TO 190                                          MM  2789
      WRITE (LOUT,7) NFILES                                             MM  2790
      ISET = 1                                                          MM  2791
190   WRITE (LOUT,5) NWDTE,NAM,APT,PT                                   MM  2792
      WRITE (LOUT,6) XIMPH                                              MM  2793
                                                                        MM  2794
      GO TO 100                                                         MM  2795
```

```
C                                                                       MM  2796
200   CONTINUE                                                          MM  2797
      DO 210 I = 1,ICOLD                                                MM  2798
      IF = IFILE + I                                                    MM  2799
      END FILE IF                                                       MM  2800



      REWIND IF                                                         MM  2801
210   CONTINUE                                                          MM  2802
C                                                                       MM  2803
      RETURN                                                            MM  2804
1     FORMAT (//1H ,50X,9HWIND DATA,//)                                 MM  2805
2     FORMAT (I6,1X,A4,A4,13X,F2.0,2X,24F2.0)                           MM  2806
3     FORMAT (32X,24A2)                                                 MM  2807
4     FORMAT (32X,24F2.0)                                               MM  2808
5     FORMAT (1H ,I6,1X,A4,A4,13X,F3.0,2HPT,24F3.0)                     MM  2809
6     FORMAT (1H ,32X,24F3.0)                                           MM  2810
7     FORMAT (//11H MORE THAN ,I3,40H DATES HAVE BEEN FOUND IN THE WIND MM  2811
     1DATA.,/,45H THE FOLLOWING ENTRIES COULD NOT BE ACCEPTED.,//)      MM  2812
      END                                                               MM  2813



      SUBROUTINE WINDY(IWAS,LTAPE)                                      MM  2814
C                                                                       MM  2815
C         WINDY IS THE SECONDARY DRIVER FOR THE AIR TRAJECTORY  MODULE  MM  2816
C                                                                       MM  2817
C     MODIFIED BY F.W. LURMANN      (12.9.77)                           MM  2818
C                                                                       MM  2819
      DIMENSION KOLD(2),KNEW(2)                                         MM  2820
      DIMENSION DDD(3)                                                  MM  2821
      DIMENSION AZV(12)                                                 MM  2822
      DIMENSION ID(2),IDD(2)                                            MM  2823
      COMMON/INPUTS/  TITLE(20),          JDATE(10),          NCURV     MM  2824
      COMMON /AIRQAL/ NEEDAQ,   NOSPEC,   CONAM(10),   CON(24,25,10)    MM  2825
      COMMON /CNTROL/ KSTOP, TSUN                                       MM  2826
      COMMON /DATES/  IDATE(10),  ICT(10),  NFILES                      MM  2827
      COMMON  /GRID/  X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT          MM  2828
      COMMON /KZINPT/ NEEDKZ,           KZPRTX,             NUMSD,      MM  2829
     1                TIMESD(3),        TEMPSD(50,3),       ZELVSD(50,3)MM  2830
     2         ,      TEMPSF(24),       NPTSD(3)       ,    NKZDAT      MM  2831
      COMMON /ORIGIN/ UTMXOR, UTMYOR                                    MM  2832
      COMMON /REUSE/  KPSTAT,  KPWDAT,  CONVRT,  KXTRA, KPWIND          MM  2833
      COMMON /TRAJ/   TSTART, P(2,100), IDXX(2,100)                     MM  2834
      COMMON /WDATA/  NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX  MM  2835
      COMMON /WHERE/  RLAT,          RLONG,              TMZONE         MM  2836
      COMMON /WIND/   T(100), V(100), TH(100), NPTS                     MM  2837
      COMMON /WINFLD/ AWDATA(52,25) ,DISAVE(25),   IDSAVE(25)           MM  2838
      INTEGER ENTRAJ                                                    MM  2839
C                                                                       MM  2840
      DATA DALITE/4HDAYL/,IBLNK/1H /,NONO/4HNO  /                       MM  2841
      DATA NIT/3/                                                       MM  2842
      DATA RTD/57.29578/, NEG/2HNO/                                     MM  2843
      DATA IDATE  /10*0/                                                MM  2844
```

```
      DATA LIN,LOUT,LPUNCH /3,6,1/                                      MM  2845
      DATA KYES/4HYES /                                                 MM  2846
      ENTRAJ = NONO                                                     MM  2847
      IF (IWAS.GT.0) GO TO 130                                          MM  2848
C                                                                       MM  2849
   95 CONTINUE                                                          MM  2850
C                                                                       MM  2851
      READ(LIN,9) KPSTAT,KPWDAT                                         MM  2852
      READ(LIN,9) KXTRA                                                 MM  2853
      READ(LIN,11) CONVRT                                               MM  2854
      IF (CONVRT.LE.0.) CONVRT = 1.0                                    MM  2855
      READ(LIN,11)  UTMXOR,  UTMYOR                                     MM  2856
                                                                        MM  2857
      READ(LIN,11)  DELX,  DELY                                         MM  2858
      READ(LIN,31)  NX, NY                                              MM  2859
      READ(LIN,11)  RMIN,  RMAX                                         MM  2860
      RMIN = RMIN**2                                                    MM  2861
      RMAX = RMAX**2                                                    MM  2862
      CALL SETIN                                                        MM  2863
      IF (KPSTAT.EQ.NONO) GO TO 120                                     MM  2864
      CALL NEWPAG(TITLE,0,JDATE)                                        MM  2865
      WRITE(LOUT,1)                                                     MM  2866
      DO 110 I = 1,NUMSTA                                               MM  2867
      IF (MOD(I,52).NE.0) GO TO 100                                     MM  2868
      CALL NEWPAG(TITLE,0,JDATE)                                        MM  2869
      WRITE(LOUT,1)                                                     MM  2870
  100 CONTINUE                                                          MM  2871
      WRITE(LOUT,2) ISTANS(1,I),ISTANS(2,I),SSTAN(1,I),SSTAN(2,I)       MM  2872
  110 CONTINUE                                                          MM  2873
  120 CONTINUE                                                          MM  2874
      IF(ENTRAJ.EQ.KYES)  GO TO 333                                     MM  2875
C                                                                       MM  2876
C     READ WIND DATA                                                    MM  2877
      CALL WINDRD(LTAPE,KPWDAT,TITLE,JDATE)                             MM  2878
      IWAS = 1                                                          MM  2879
C                                                                       MM  2880
  130 CONTINUE                                                          MM  2881
C                                                                       MM  2882
C     READ CASE DATA                                                    MM  2883
C                                                                       MM  2884
      ICASES = 1                                                        MM  2885
      IK = 0                                                            MM  2886
      READ(LIN,8) IDT,XLCL                                              MM  2887
      TSUN = 0.                                                         MM  2888
      READ(LIN,9) STIME                                                 MM  2889
      IF (STIME.EQ.DALITE) TSUN = 0100.                                 MM  2890
      ILCL = IFIX(XLCL)                                                 MM  2891
      IST = IFIX(XLCL-TSUN)                                             MM  2892
      READ(LIN,10) ID,XS,YS                                             MM  2893
      XS = XS - UTMXOR   <-IF(ID(1).NE.IBLK)GOTO 132                    MM  2894
      YS = YS - UTMYOR                                                  MM  2895
  132 READ(LIN,11) TTOTAL                                               MM  2896
      READ(LIN,11) DTSEG                                                MM  2897
      ITN = IFIX(60.*DTSEG)                                             MM  2898
      NN = IFIX((TTOTAL+0.9*DTSEG)/DTSEG)                               MM  2899
```

```
      READ(LIN,11) AZ,VEL                                                MM  2900
      READ(LIN,12) ITF                                                   MM  2901
      READ(LIN,12) IRF                                                   MM  2902
      READ(LIN,12) NC                                                    MM  2903
C                                                                        MM  2904
      IF(IDT.EQ.IDATE(1))  GO TO 134                                     MM  2905
C       GET THE METEORELOGICAL AND AIR QUALITY DATA FOR ST. LOUIS.       MM  2906
      CALL METIN (IDT,LTAPE)                                             MM  2907
  134 CONTINUE                                                           MM  2908
C                                                                        MM  2909
C     IDT     DATE YEAR-MONTH-DAY                                        MM  2910
C     ID      STATION IDENTIFICATION (OR BLANK)                          MM  2911
C     XS      X COORDINATE OF START                                      MM  2912
C     YS      Y COORDINATE OF START                                      MM  2913
C     ILCL    STARTING TIME OF DAY - 24 HOUR CLOCK - LOCAL TIME          MM  2914
C     IST     STARTING TIME OF DAY - 24 HOUR CLOCK - STANDARD TIME       MM  2915
C     ITN     TIME BETWEEN MOVES                                         MM  2916
C     NN      NUMBER OF MOVES TO BE MADE                                 MM  2917
C     AZ      STARTING AZIMUTH(IF ANY)                                   MM  2918
C     VEL     STARTING VELOCITY (IF ANY)                                 MM  2919
                                                                         MM  2920
C     ITF     IF POS MOVE FORWARD IN TIME, IF NEG MOVE BACKWARD          MM  2921
C     IRF     IF 1 CALCULATE INVERSE DISTANCE, IF 2 CALCULATE INVERSE SQUMM  2922
C     NC      NUMBER OF CLOSEST STATIONS TO TRY TO USE                   MM  2923
C                                                                        MM  2924
C                                                                        MM  2925
C     IF STATION ID IS BLANK DO COORDINATES CORRESPOND TO A KNOWN STATIOMM  2926
C                                                                        MM  2927
      IF (ID(1).NE.IBLNK) GO TO 150                                      MM  2928
      DO 140 I = 1,NUMSTA                                                MM  2929
      IF (XS.NE.SSTAN(1,I)) GO TO 140                                    MM  2930
      IF (YS.NE.SSTAN(2,I)) GO TO 140                                    MM  2931
      ID(1) = ISTANS(1,I)                                                MM  2932
      ID(2) = ISTANS(2,I)                                                MM  2933
      GO TO 150                                                          MM  2934
140   CONTINUE                                                           MM  2935
150   CONTINUE                                                           MM  2936
C                                                                        MM  2937
C IF COORDINATES ARE ZERO LOOK UP STATION AND SUPPLY COORDINATES         MM  2938
      IF (XS.NE.0..OR.YS.NE.0.) GO TO 170                                MM  2939
      DO 160 I = 1,NUMSTA                                                MM  2940
      IF (ID(1).NE.ISTANS(1,I)) GO TO 160                                MM  2941
      IF (ID(2).NE.ISTANS(2,I)) GO TO 160                                MM  2942
      XS = SSTAN(1,I)                                                    MM  2943
      YS = SSTAN(2,I)                                                    MM  2944
      GO TO 170                                                          MM  2945
160   CONTINUE                                                           MM  2946
170   CONTINUE                                                           MM  2947
C                                                                        MM  2948
C     PRINT THE INITIAL INPUTS                                           MM  2949
C                                                                        MM  2950
      CALL NEWPAG(TITLE,0,JDATE)                                         MM  2951
      WRITE(LOUT,13) IDT,ILCL                                            MM  2952
      IF (TSUN.EQ.0.) WRITE(LOUT,14)                                     MM  2953
      IF (TSUN.EQ.0100.) WRITE(LOUT,15)                                  MM  2954
```

```
      WRITE(LOUT,16) XS,YS                                              MM  2955
      IF (ID(1).EQ.IBLNK) WRITE(LOUT,17)                                MM  2956
      IF (ID(1).NE.IBLNK) WRITE(LOUT,18) ID                             MM  2957
      WRITE(LOUT,19)                                                    MM  2958
      IF (VEL.GT.0.) WRITE(LOUT,20) AZ,VEL                              MM  2959
      WRITE(LOUT,21)                                                    MM  2960
      IF (ITF.GE.0) WRITE(LOUT,22)                                      MM  2961
      IF (ITF.LT.0) WRITE(LOUT,23)                                      MM  2962
      WRITE(LOUT,24) TTOTAL                                             MM  2963
      WRITE(LOUT,25) DTSEG                                              MM  2964
      WRITE(LOUT,26) NC                                                 MM  2965
      IF (IRF.EQ.1) WRITE(LOUT,27)                                      MM  2966
      IF (IRF.EQ.2) WRITE(LOUT,28)                                      MM  2967
      IF(KXTRA.EQ.KYES)  WRITE(LOUT,33)                                 MM  2968
C                                                                       MM  2969
      KOLD(1) = IST                                                     MM  2970
      KOLD(2) = IDT                                                     MM  2971
      KNEW(1) = IST                                                     MM  2972
      KNEW(2) = IDT                                                     MM  2973
      IOD(1) = ID(1)                                                    MM  2974
      IOD(2) = ID(2)                                                    MM  2975
      IF (ITF.LT.0) CALL DATE(KOLD,ITN,ITF,KNEW)                        MM  2976
      IF (VEL.GT.0.) GO TO 190                                          MM  2977
      CALL GETAZV(KNEW,IOD,XS,YS,NC,AZV,NGD,DDD)                        MM  2978
      IF (NGD.LE.0) GO TO 180                                           MM  2979
      CALL AZVDIS(NGD,AZV,DDD,AZ,VEL,IRF)                               MM  2980
      VEL = VEL*CONVRT                                                  MM  2981
      GO TO 190                                                         MM  2982
                                                                        MM  2983
180   CONTINUE                                                          MM  2984
      IK = IK-1                                                         MM  2985
      WRITE(LOUT,3)                                                     MM  2986
      GO TO 280                                                         MM  2987
C                                                                       MM  2988
190   CONTINUE                                                          MM  2989
      IK = 1                                                            MM  2990
      P(1,1) = XS                                                       MM  2991
      P(2,1) = YS                                                       MM  2992
      NTM = 0                                                           MM  2993
      XMUL = DTSEG                                                      MM  2994
      IF (ITF.GT.0) GO TO 200                                           MM  2995
      XMUL = -DTSEG                                                     MM  2996
      T(1) = FLOAT(KOLD(1))                                             MM  2997
      IK = 2                                                            MM  2998
      GO TO 250                                                         MM  2999
200   CONTINUE                                                          MM  3000
      IF (KXTRA.EQ.KYES) WRITE(LOUT,4) KOLD,XS,YS                       MM  3001
      IF (ITF.GT.0) GO TO 210                                           MM  3002
      KOLD(1) = KNEW(1)                                                 MM  3003
      KOLD(2) = KNEW(2)                                                 MM  3004
210   CONTINUE                                                          MM  3005
C                                                                       MM  3006
C     APPLY MOVE NOW TO GET NEW POINT                                   MM  3007
      ANG = 90.-AZ                                                      MM  3008
      IF (ANG.LT.0.) ANG = ANG+360.                                     MM  3009
```

```
      ANGUSE = ANG/57.29577951                                          MM  3010
      T(IN) = FLOAT(KOLD(1))                                            MM  3011
      V(IN) = VEL                                                       MM  3012
      TH(IN) = ANGUSE                                                   MM  3013
      IF (EXTRA.EQ.NYES) WRITE(LOUT,5) KOLD,VEL,ANG,(AZV(I),I=4,12,4)   MM  3014
      XNEW = XS+XMUL*COS(ANGUSE)*VEL                                    MM  3015
      YNEW = YS+XMUL*SIN(ANGUSE)*VEL                                    MM  3016
      CALL BARIER(XNEW,YNEW,XS,YS,OK)                                   MM  3017
      IF (OK.LT.0.) GO TO 320                                           MM  3018
      CALL EDGE(XNEW,YNEW,ITF,IEDGE)                                    MM  3019
      IF (IEDGE) 330,230,220                                            MM  3020
220   CONTINUE                                                          MM  3021
      IF (NTM.GE.MM) GO TO 280                                          MM  3022
      NN = NTM+1                                                        MM  3023
230   CONTINUE                                                          MM  3024
      XS = XNEW                                                         MM  3025
      YS = YNEW                                                         MM  3026
C                                                                       MM  3027
C     LAST MOVE CHECK                                                   MM  3028
      IF (NTM.GE.MM) GO TO 280                                          MM  3029
C                                                                       MM  3030
      IN = IN+1                                                         MM  3031
C                                                                       MM  3032
C     CHANGE DATE                                                       MM  3033
      CALL DATE(KOLD,ITN,ITF,KNEW)                                      MM  3034
      IF (ITF.LT.0) GO TO 240                                           MM  3035
      KOLD(1) = KNEW(1)                                                 MM  3036
      KOLD(2) = KNEW(2)                                                 MM  3037
240   CONTINUE                                                          MM  3038
      IDD(1) = IBLNK                                                    MM  3039
      IDD(2) = IBLNK                                                    MM  3040
      CALL GETAZV(KNEW,IDD,XS,YS,NC,AZV,NGD,DDD)                        MM  3041
      IF (NGD.LE.0) GO TO 180                                           MM  3042
      CALL AZVOIS(NGD,AZV,DDD,AZ,VEL,IRF)                               MM  3043
      VEL = VEL*CONVRT                                                  MM  3044
C                                                                       MM  3045
                                                                        MM  3046
C     WE JUST GOT A NEW AZIMUTH AND VELOCITY                            MM  3047
C     IF WE ARE MOVING BACKWARD IN TIME, THERE FOLLOW  NIT  ITERATIONS  MM  3048
C     TO REFINE THE AZIMUTH AND VELOCITY.                               MM  3049
C     (NIT  IS SET IN A DATA STATEMENT ABOVE.)                          MM  3050
C                                                                       MM  3051
      IF(ITF.GT.0)  GO TO 270                                           MM  3052
C                                                                       MM  3053
C                                                                       MM  3054
250   CONTINUE                                                          MM  3055
      IF (NIT.EQ.0) GO TO 270                                           MM  3056
      DO 260 IT = 1,NIT                                                 MM  3057
      ANG1 = 90.-AZ                                                     MM  3058
      IF (ANG1.LT.0.) ANG1 = ANG1+360.                                  MM  3059
      ANGYU = ANG1/57.29577951                                          MM  3060
      XNUT = XS+XMUL*COS(ANGYU)*VEL                                     MM  3061
      YNUT = YS+XMUL*SIN(ANGYU)*VEL                                     MM  3062
      CALL BARIER(XNUT,YNUT,XS,YS,OK)                                   MM  3063
      IF (OK.LT.0) GO TO 310                                            MM  3064
```

```
      IDD(1) = IBLNK                                                    MM  3065
      IDD(2) = IBLNK                                                    MM  3066
      CALL GETAZV(KNEW,IDD,XNUT,YNUT,NC,AZV,NGD,DDD)                    MM  3067
      IF (NGD.LE.0) GO TO 180                                           MM  3068
      CALL AZVDIS(NGD,AZV,DDD,AZ,VEL,IRF)                               MM  3069
      VEL = VEL*CONVRT                                                  MM  3070
260   CONTINUE                                                          MM  3071
270   CONTINUE                                                          MM  3072
      NTM = NTM+1                                                       MM  3073
      GO TO 200                                                         MM  3074
C                                                                       MM  3075
280   CONTINUE                                                          MM  3076
      NPTS = IK                                                         MM  3077
C                                                                       MM  3078
      IF (ITF.GT.0) GO TO 300                                           MM  3079
C                                                                       MM  3080
C     REVERSE THE ORDER OF THE TIME, VELOCITY, AND ANGLE LISTS          MM  3081
C                                                                       MM  3082
      NXY = (IK-MOD(IK,2))/2                                            MM  3083
      DO 290 K = 1,NXY                                                  MM  3084
      J = IK-K+1                                                        MM  3085
      TT = T(K)                                                         MM  3086
      T(K) = T(J)                                                       MM  3087
      T(J) = TT                                                         MM  3088
      VV = V(K)                                                         MM  3089
      V(K) = V(J)                                                       MM  3090
      V(J) = VV                                                         MM  3091
      AA = TH(K)                                                        MM  3092
      TH(K) = TH(J)                                                     MM  3093
      TH(J) = AA                                                        MM  3094
290   CONTINUE                                                          MM  3095
      V(IK) = V(IK-1)                                                   MM  3096
      TH(IK) = TH(IK-1)                                                 MM  3097
      P(1,1) = XS                                                       MM  3098
      P(2,1) = YS                                                       MM  3099
300   CONTINUE                                                          MM  3100
      IF (KPWIND.EQ.NEG) GO TO 308                                      MM  3101
      WRITE(LPUNCH,30)  TITLE                                           MM  3102
      A = P(1,1) + UTMXOR                                               MM  3103
      B = P(2,1) + UTMYOR                                               MM  3104
      WRITE(LPUNCH,47)  A, B                                            MM  3105
      IF(TSUN.EQ.0.)  WRITE(LPUNCH,48)                                  MM  3106
      IF(TSUN.GT.0.)  WRITE(LPUNCH,49)                                  MM  3107
      DO 305 I = 1,NPTS                                                 MM  3108
                                                                        MM  3109
      A = TH(I)*RTD                                                     MM  3110
      B = TIMIN(T(I))                                                   MM  3111
      WRITE(LPUNCH,32)  B,  V(I),  A                                    MM  3112
  305 CONTINUE                                                          MM  3113
      A = -100.                                                         MM  3114
      WRITE(LPUNCH,32)  A                                               MM  3115
308   CONTINUE                                                          MM  3116
C                                                                       MM  3117
      GO TO 500                                                         MM  3118
C PRINT MESSAGE                                                         MM  3119
```

```
310   CONTINUE                                                          MM  3120
      XNEW = XNUT                                                       MM  3121
      YNEW = YNUT                                                       MM  3122
320   CONTINUE                                                          MM  3123
      IF(ITF.LT.0) IK = IK + 1                                          MM  3124
      WRITE(LOUT,6) XS,YS,XNEW,YNEW                                     MM  3125
      GO TO 280                                                         MM  3126
330   CONTINUE                                                          MM  3127
      IK = IK+1                                                         MM  3128
      WRITE(LOUT,7)                                                     MM  3129
      GO TO 280                                                         MM  3130
 332  CONTINUE                                                          MM  3131
C                                                                       MM  3132
C                                                                       MM  3133
C        FOR THE UNIVAC (OR IBM) REPLACE   *ENTRY TRAJEC*  WITH         MM  3134
C        *ENTRY TRAJEC(IWAS,KPWIND,LTAPE)*                              MM  3135
C                                                                       MM  3136
      ENTRY TRAJEC                                                      MM  3137
C                                                                       MM  3138
      IF(IWAS.GT. 0)  GO TO 335                                         MM  3139
      ENTRAJ = KYES                                                     MM  3140
      GO TO 95                                                          MM  3141
  333 CONTINUE                                                          MM  3142
      IWAS = 1                                                          MM  3143
  335 CONTINUE                                                          MM  3144
C                                                                       MM  3145
C        READ INPUTS FOR INDIVIDUAL CASE                                MM  3146
C                                                                       MM  3147
      READ(LIN,8)  IDT                                                  MM  3148
      READ(LIN,12) IRF                                                  MM  3149
      READ(LIN,12) NC                                                   MM  3150
C                                                                       MM  3151
      P(1,1) = P(1,1) + UTMXOR                                          MM  3152
      P(2,1) = P(2,1) + UTMYOR                                          MM  3153
      IF(IDT.EQ.IDATE(1))  GO TO 350                                    MM  3154
C                                                                       MM  3155
C       GET THE METEORELOGICAL AND AIR QUALITY DATA FOR ST. LOUIS.      MM  3156
      IF(NEEDAQ.NE.KYES)  GO TO 350                                     MM  3157
      CALL METIN (IDT,LTAPE)                                            MM  3158
  350 CONTINUE                                                          MM  3159
C                                                                       MM  3160
      ILCL = IFIX(T(1))                                                 MM  3161
C     PRINT THE INITIAL INPUT                                           MM  3162
      CALL NEWPAG(TITLE,0,JDATE)                                        MM  3163
      WRITE(LOUT,13) IDT,ILCL                                           MM  3164
      IF (TSUN.EQ.0.) WRITE(LOUT,14)                                    MM  3165
      IF (TSUN.EQ.0100.) WRITE(LOUT,15)                                 MM  3166
      WRITE(LOUT,16)  P(1,1), P(2,1)                                    MM  3167
      WRITE(LOUT,26) NC                                                 MM  3168
      IF (IRF.EQ.1) WRITE(LOUT,27)                                      MM  3169
      IF (IRF.EQ.2) WRITE(LOUT,28)                                      MM  3170
      DO 360 J = 1,NPTS                                                 MM  3171
                                                                        MM  3172
      V(J) = V(J)*CONVRT                                                MM  3173
 360  CONTINUE                                                          MM  3174
```

```
C                                                                       MM  3175
 500  CONTINUE                                                          MM  3176
C                                                                       MM  3177
      CALL CROSIT                                                       MM  3178
      IF(KSTOP.GT.0)  GO TO 540                                         MM  3179
C                                                                       MM  3180
      CALL EMETIN                                                       MM  3181
C                                                                       MM  3182
C     CONVERT TIME FROM MINUTES TO 2400 - HOUR CLOCK AND CONVERT        MM  3183
C     WIND SPEEDS FROM KILOMETERS/MINUTE TO METERS/SECOND               MM  3184
      DT = T(2) - T(1)  + .001                                          MM  3185
      T(1) = TSTART                                                     MM  3186
      V(1) = V(1)*16.666667                                             MM  3187
      DO 510 J = 2,NPTS                                                 MM  3188
      IF(J.GT.100)  GO TO 515                                           MM  3189
      DTNXT = T(J+1) - T(J) + .001                                      MM  3190
      T(J) = CLOCKT( T(J-1) , DT)                                       MM  3191
      DT = DTNXT                                                        MM  3192
      V(J) = V(J)*16.666667                                             MM  3193
 510  CONTINUE                                                          MM  3194
 515  CONTINUE                                                          MM  3195
C                                                                       MM  3196
      IF(NEEDAQ.NE.KYES)  GO TO 530                                     MM  3197
C                                                                       MM  3198
      CALL NEWPAG(TITLE,0,JDATE)                                        MM  3199
      KNEW(2) = IDT                                                     MM  3200
      DO 520 J = 1,NPTS                                                 MM  3201
      KNEW(1) = IFIX(T(J))                                              MM  3202
      CALL WSECLS (P(1,J),P(2,J),DISAVE,IDSAVE)                         MM  3203
      CALL GETCON (KNEW,NC,IRF)                                         MM  3204
      IF(J.EQ.NPTS)  GO TO 520                                          MM  3205
      IF(MOD(J,2).EQ. 0)  CALL NEWPAG(TITLE,0,JDATE)                    MM  3206
 520  CONTINUE                                                          MM  3207
 530  CONTINUE                                                          MM  3208
C                                                                       MM  3209
      IF(NEEDKZ.NE.KYES)  GO TO 540                                     MM  3210
      CALL EDDY(KPWIND)                                                 MM  3211
 540  CONTINUE                                                          MM  3212
      RETURN                                                            MM  3213
C                                                                       MM  3214
C                                                                       MM  3215
1     FORMAT (//10X,25H STATIONS AND COORDINATES,/)                     MM  3216
2     FORMAT (1H ,10X,2A4,2F10.3)                                       MM  3217
3     FORMAT (26H0THERE ARE NO CLOSE POINTS)                            MM  3218
4     FORMAT (1H ,I6,4X,I6,4X,2F6.2)                                    MM  3219
5     FORMAT (1H+,40X,I6,4X,I6,4X,2F8.2,2X,3(2X,A4))                    MM  3220
6     FORMAT (27H0POINT MOVED ACROSS BARRIER,2(4X,2F10.2)//)            MM  3221
7     FORMAT (/42H0TRAJECTORY BACKED MORE THAN 15 KM OF GRID    //)     MM  3222
8     FORMAT (40X,I6,4X,F4.0)                                           MM  3223
9     FORMAT (40X,A4,6X,A4)                                             MM  3224
10    FORMAT (40X,2A4,2X,2F10.0)                                        MM  3225
11    FORMAT (40X,2F10.0)                                               MM  3226
12    FORMAT (40X,I2)                                                   MM  3227
13    FORMAT (///,33H  AIR TRAJECTORY CALCULATION DATA ,///             MM  3228
     1  47H     START DATE AND LOCAL TIME AT INITIAL POINT,13X,I6,4X,I4)MM  3229
```

```
14    FORMAT (1H+,73X,20H HRS (STANDARD TIME))                                MM  3230
15    FORMAT (1H+,73X,20H HRS (DAYLIGHT TIME))                                MM  3231
16    FORMAT ( /,30H      LOCATION OF INITIAL POINT,30X,F6.2,4X,F6.2,3X)      MM  3232
17    FORMAT (1H+,78X,30H(NOT AT A MEASUREMENT STATION))                      MM  3233
18    FORMAT (1H+,78X,1H(,A4,A4,23H  MEASUREMENT STATION)   )                 MM  3234
                                                                              MM  3235
19    FORMAT (/,                                                              MM  3236
     1  56H     PRESCRIBED STARTING AZIMUTH AND VELOCITY (OPTIONAL) )         MM  3237
20    FORMAT (1H+,59X,F6.2,4X,F6.2,  6H KM/HR  )                              MM  3238
21    FORMAT ( /,40H      DIRECTION OF TRAJECTORY DEVELOPMENT )               MM  3239
22    FORMAT (1H+,59X,7HFORWARD )                                             MM  3240
23    FORMAT (1H+,59X,8HBACKWARD )                                            MM  3241
24    FORMAT ( /,32H      DESIRED TRAJECTORY DURATION,28X,F5.1,4H HRS )       MM  3242
25    FORMAT ( /,30H      TRAJECTORY SEGMENT LENGTH,30X,F5.1,4H HRS )         MM  3243
26    FORMAT ( /,42H      MEASUREMENT INTERPOLATION SCHEME      ,18X,I1)      MM  3244
27    FORMAT (1H+,60X,39H  CLOSEST STATIONS WITH  1/R  WEIGHTING ,//)         MM  3245
28    FORMAT (1H+,60X,42H  CLOSEST STATIONS WITH  1/R**2  WEIGHTING ,//)      MM  3246
30    FORMAT (20A4)                                                           MM  3247
31    FORMAT (40X,2I10)                                                       MM  3248
32    FORMAT (40X,3F10.3)                                                     MM  3249
33    FORMAT (1H0,//,17H OLD TIME - DATE ,7X,1HX,5X,1HY,10X,                  MM  3250
     1 16HNEW TIME - DATE ,6X,16HVELOCITY AZIMUTH  ,4X,8HSTATIONS ,/)         MM  3251
47    FORMAT (23HSTART LOCATION (COORDS),17X,2F10.2)                          MM  3252
48    FORMAT (34HIS LOCAL TIME STANDARD OR DAYLIGHT,6X,8HSTANDARD)            MM  3253
49    FORMAT (34HIS LOCAL TIME STANDARD OR DAYLIGHT,6X,8HDAYLIGHT)            MM  3254
      END                                                                     MM  3255



      SUBROUTINE WSECLS(X,Y,DIS,IDIS)                                         MM  3256
C                                                                             MM  3257
C         THIS SUBROUTINE CALCULATES SQUARED DISTANCES FROM A GIVEN           MM  3258
C         POINT TO EACH OF A SET OF POINTS AND THEN ORDERS THE DISTANCES      MM  3259
C         OUT TO A MAXIMUM VALUE.                                             MM  3260
C                                                                             MM  3261
C         X           X COORDINATE OF GIVEN POINT                             MM  3262
C         Y           Y COORDINATE OF GIVEN POINT                             MM  3263
C         DIS         THE SQUARE OF THE DISTANCE FROM THE GIVEN POINT TO      MM  3264
C                     ALL THE POINTS IN THE POINT SET.  THOSE VALUES OF       MM  3265
C                     DIS LESS THAN RMAX ARE ARRANGED IN INCREASING           MM  3266
C                     ORDER AT THE TOP OF THE TABLE.                          MM  3267
C         IDIS        THE NUMBER OF THE POINT IN THE POINT SET                MM  3268
C                     CORRESPONDING TO THE DIS ARRAY                          MM  3269
C                                                                             MM  3270
      DIMENSION DIS(2), IDIS(2)                                               MM  3271
      COMMON /WDATA/ NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX         MM  3272
C                                                                             MM  3273
      DO 100 I = 1,NUMSTA                                                     MM  3274
      IDIS(I) = I                                                             MM  3275
      DIS(I) = (SSTAN(1,I)-X) **2+(SSTAN(2,I)-Y) **2                          MM  3276
100   CONTINUE                                                                MM  3277
C                                                                             MM  3278
      DO 110 N = 1,NUMSTA                                                     MM  3279
      M = NUMSTA+1-N                                                          MM  3280
      CALL FMINF(DIS(N),M,FMIN,NMIN)                                          MM  3281
```

```
      NMIN = NMIN+N-1                                                   MM  3282
      IF (FMIN.GT.RMAX) GO TO 120                                       MM  3283
      J = IDIS(N)                                                       MM  3284
      IDIS(N) = IDIS(NMIN)                                              MM  3285
      IDIS(NMIN) = J                                                    MM  3286
      XX = DIS(N)                                                       MM  3287
      DIS(N) = FMIN                                                     MM  3288
      DIS(NMIN) = XX                                                    MM  3289
110   CONTINUE                                                          MM  3290
C                                                                       MM  3291
120   RETURN                                                            MM  3292
      END                                                               MM  3293



      BLOCK DATA                                                        MM  3294
C                                                                       MM  3295
C         BLOCK DATA PROGRAM NUMBER ONE                                 MM  3296
C         VARIABLE SIZE GRID ID NUMBERS  FOR ST. LOUIS                  MM  3297
      INTEGER A1, A2, A3, A4, A5, A6, A7, A8                            MM  3298
                                                                        MM  3299
      DIMENSION   A1(341),  A2(370),  A3(439) , A4(441),               MM  3300
     1            A5(319),  A6(315),  A7(308),  A8(248)                MM  3301
      COMMON /VGRID/  ITGRID,     IVG(100,100)                          MM  3302
      EQUIVALENCE (A1(1),IVG(1,1))                                      MM  3303
      EQUIVALENCE (A2(1),IVG(42,4))                                     MM  3304
      EQUIVALENCE (A3(1),IVG(12,8))                                     MM  3305
      EQUIVALENCE (A4(1),IVG(51,12))                                    MM  3306
      EQUIVALENCE (A5(1),IVG(92,16))                                    MM  3307
      EQUIVALENCE (A6(1),IVG(11,20))                                    MM  3308
      EQUIVALENCE (A7(1),IVG(26,23))                                    MM  3309
      EQUIVALENCE (A8(1),IVG(34,26))                                    MM  3310
C                                                                       MM  3311
      DATA ITGRID /4HYES /                                              MM  3312
C                                                                       MM  3313
      DATA A1/                                                          MM  3314
     1       64, 4*  64,     72, 4*  72,     79, 4*  79,     96, 9*  96,MM  3315
     2      147, 4* 147,    181, 4* 181,    214, 4* 214,    291, 4* 291,MM  3316
     3      348,    395,    424,    467, 1* 467,    528, 4* 528,    695,MM  3317
     4   4* 695,    878, 4* 878,   1056, 4*1056,   1205, 4*1205,   1373,MM  3318
     5   4*1373,   2335, 2*2335,   2361, 1*2361,   1582, 4*1582,   1616,MM  3319
     6   4*1616,   1633, 4*1633,     64, 4*  64,     72, 4*  72,     79,MM  3320
     7   4*  79,     96, 9*  96,    147, 4* 147,    181, 4* 181,    214,MM  3321
     8   4* 214,    291, 4* 291,    349,    396,    425,    467, 1* 467,MM  3322
     9      528, 4* 528,    695, 4* 695,    878, 4* 878,   1056, 4*1056,MM  3323
     1     1205, 4*1205,   1373, 4*1373,   2335, 2*2335,   2361, 1*2361,MM  3324
     2     1582, 4*1582,   1616, 4*1616,   1633, 4*1633,     64, 4*  64,MM  3325
     3       72, 4*  72,     79, 4*  79,     96, 9*  96,    147, 4* 147,MM  3326
     4      181, 4* 181,    214, 4* 214,    291, 4* 291,    350,    397,MM  3327
     5      426,    468,    496,    528, 4* 528,    695, 4* 695,    878,MM  3328
     6   4* 878,   1056, 4*1056,   1205, 4*1205,   1373, 4*1373,   2335,MM  3329
     7   2*2335,   2362,   2368,   1582, 4*1582,   1616, 4*1616,   1633,MM  3330
     8   4*1633,     64, 4*  64,     72, 4*  72,     79, 4*  79,     96,MM  3331
     9   9*  96,    147, 4* 147,    181, 4* 181,    214, 4* 214,    291/MM  3332
      DATA A2/                                                          MM  3333
```

```
1    4* 291,     351,     398,     427,     469,     [illegible],     528, 4* 528,MM  3334
2       695, 4* 695,     878, 4* 878,    1056, 4*1056,    1205, 4*1205,MM  3335
3      1373, 4*1373,    2336, 1*2336,    2347,    2363, 1*2363,    1582,MM  3336
4    4*1582,    1616, 4*1616,    1633, 4*1633,       , 4*  64,      72,MM  3337
5    4*  72,      79, 4*  79,      96, 9*  96,     1 7, 4* 147,     181,MM  3338
6    4* 181,     214, 4* 214,     291, 4* 291,     352,     399,     428,MM  3339
7       470,     498,     528, 4* 528,     695, 4* 695,     878, 4* 878,MM  3340
8      1056, 4*1056,    1205, 4*1205,    1373, 4*1373,    2336, 1*2336,MM  3341
9      2348,    2363, 1*2363,    1582, 4*1582,    1616, 4*1616,    1633,MM  3342
1    4*1633,      65, 9*  65,      80, 4*  80,      97, 4*  97,     126,MM  3343
2    4* 126,     148, 4* 148,     182, 4* 182,     215, 4* 215,     292,MM  3344
3    4* 292,     353,     400,     429,     471,     499,     529, 4* 529,MM  3345
4       696, 4* 696,     879, 4* 879,    1057, 9*1057,    1374, 4*1374,MM  3346
5      1480, 9*1480,    1617, 9*1617,      65, 9*  65,      80, 4*  80,MM  3347
6        97, 4*  97,     126, 4* 126,     148, 4* 148,     182, 4* 182,MM  3348
7       215, 4* 215,     292, 4* 292,     354, 3* 354,     500,     529,MM  3349
8    4* 529,     696, 4* 696,     879, 4* 879,    1057, 9*1057,    1374,MM  3350
9    4*1374,    1480, 9*1480,    1617, 9*1617,      65, 9*  65,      80/MM  3351
 DATA A3/                                                                MM  3352
1    4*  80,      97, 4*  97,     126, 4* 126,     148, 4* 148,     182,MM  3353
2    4* 182,     215, 4* 215,     292, 4* 292,     354, 3* 354,     501,MM  3354
3       529, 4* 529,     696, 4* 696,     879, 4* 879,    1057, 9*1057,MM  3355
4      1374, 4*1374,    1480, 9*1480,    1617, 9*1617,      65, 9*  65,MM  3356
5        80, 4*  80,      97, 4*  97,     126, 4* 126,     148, 4* 148,MM  3357
6       182, 4* 182,     215, 4* 215,     292, 4* 292,     354, 3* 354,MM  3358
7       502,     529, 4* 529,     696, 4* 696,     879, 4* 879,    1057,MM  3359
8    9*1057,    1374, 4*1374,    1480, 9*1480,    1617, 9*1617,      65,MM  3360
9    9*  65,      80, 4*  80,      97, 4*  97,     126, 4* 126,     148,MM  3361
1    4* 148,     182, 4* 182,     215, 4* 215,     292, 4* 292,     354,MM  3362
2    3* 354,     503,     529, 4* 529,     696, 4* 696,     879, 4* 879,MM  3363
3      1057, 9*1057,    1374, 4*1374,    1480, 9*1480,    1617, 9*1617,MM  3364
4        65, 9*  65,      81, 4*  81,      98, 4*  98,     127, 4* 127,MM  3365
5       149, 4* 149,     183, 4* 183,     216, 4* 216,     293, 4* 293,MM  3366
6       355, 4* 355,     530, 4* 530,     697, 9* 697,    1057, 9*1057,MM  3367
7      1375, 4*1375,    1480, 9*1480,    1617, 9*1617,      65, 9*  65,MM  3368
8        81, 4*  81,      98, 4*  98,     127, 4* 127,     149, 4* 149,MM  3369
9       183, 4* 183,     216, 4* 216,     293, 4* 293,     355, 4* 355/MM  3370
 DATA A4/                                                                MM  3371
1       530, 4* 530,     697, 9* 697,    1057, 9*1057,    1375, 4*1375,MM  3372
2      1480, 9*1480,    1617, 9*1617,      65, 9*  65,      81, 4*  81,MM  3373
3        98, 4*  98,     127, 4* 127,     149, 4* 149,     183, 4* 183,MM  3374
4       216, 4* 216,     293, 4* 293,     355, 4* 355,     530, 4* 530,MM  3375
5       697, 9* 697,    1057, 9*1057,    1375, 4*1375,    1480, 9*1480,MM  3376
6      1617, 9*1617,      65, 9*  65,      81, 4*  81,      98, 4*  98,MM  3377
7       127, 4* 127,     149, 4* 149,     183, 4* 183,     216, 4* 216,MM  3378
8       293, 4* 293,     355, 4* 355,     530, 4* 530,     697, 9* 697,MM  3379
9      1057, 9*1057,    1375, 4*1375,    1480, 9*1480,    1617, 9*1617,MM  3380
1        65, 9*  65,      81, 4*  81,      98, 4*  98,     127, 4* 127,MM  3381
2       149, 4* 149,     183, 4* 183,     216, 4* 216,     293, 4* 293,MM  3382
3       355, 4* 355,     530, 4* 530,     697, 9* 697,    1057, 9*1057,MM  3383
4      1375, 4*1375,    1480, 9*1480,    1617, 9*1617,      66, 9*  66,MM  3384
5        82, 4*  82,      99, 4*  99,     128, 4* 128,     150, 4* 150,MM  3385
6       184, 4* 184,    2088, 2*2088,    2129, 2*2129,    2151, 2*2151,MM  3386
7      2188, 2*2188,    2203, 2*2203,    2232, 1*2232,    2246, 2*2246,MM  3387
8       697, 9* 697,    1058, 1*1058,    1116,    1145,    1174,    1206,MM  3388
```

```
9   4*1206,   1376, 4*1376,   1481, 4*1481,   1583, 4*1583,   1618/MM  3389
 DATA A5/                                                       MM    3390
1    4*1618,   1634, 4*1634,     66, 9*  66,     82, 4*  82,     99,MM 3391
2    4*  99,    128, 4* 128,    150, 4* 150,    184, 4* 184,   2088,MM 3392
3    2*2088,   2129, 2*2129,   2151, 2*2151,   2188, 2*2188,   2203,MM 3393
4    2*2203,   2232, 1*2232,   2246, 2*2246,    697, 9* 697,   1058,MM 3394
5    1*1058,   1117,   1146,   1175,   1206, 4*1206,   1376, 4*1376,MM 3395
6      1481, 4*1481,   1583, 4*1583,   1618, 4*1618,   1634, 4*1634,MM 3396
7        66, 9*  66,     82, 4*  82,     99, 4*  99,    128, 4* 128,MM 3397
8       150, 4* 150,    184, 4* 184,   2088, 2*2088,   2129, 2*2129,MM 3398
9      2151, 2*2151,   2188, 2*2188,   2203, 2*2203,   2233,   2235,MM 3399
1      2246, 2*2246,    697, 9* 697,   1059, 2*1059,   1147,   1176,MM 3400
2      1206, 4*1206,   1376, 4*1376,   1481, 4*1481,   1583, 4*1583,MM 3401
3      1618, 4*1618,   1634, 4*1634,     66, 9*  66,     82, 4*  82,MM 3402
4        99, 4*  99,    128, 4* 128,    150, 4* 150,    184, 4* 184,MM 3403
5      2089, 1*2089,   2111, 1*2111,   2131, 1*2131,   2152, 1*2152,MM 3404
6      2171, 1*2171,   2190, 1*2190,   2204, 1*2204,   2218, 1*2218,MM 3405
7      2236, 1*2236,   2247, 1*2247,    697, 9* 697,   1059, 2*1059,MM 3406
8      1148,   1177,   1206, 4*1206,   1376, 4*1376,   1481, 4*1481,MM 3407
9      1583, 4*1583,   1618, 4*1618,   1634, 4*1634,     66, 9*  66/MM 3408
 DATA A6/                                                       MM    3409
1        82, 4*  82,     99, 4*  99,    128, 4* 128,    150, 4* 150,MM 3410
2       184, 4* 184,   2089, 1*2089,   2111, 1*2111,   2131, 1*2131,MM 3411
3      2152, 1*2152,   2171, 1*2171,   2190, 1*2190,   2204, 1*2204,MM 3412
4      2218, 1*2218,   2236, 1*2236,   2247, 1*2247,    697, 9* 697,MM 3413
5      1059, 2*1059,   1149,   1178,   1206, 4*1206,   1376, 4*1376,MM 3414
6      1481, 4*1481,   1583, 4*1583,   1618, 4*1618,   1634, 4*1634,MM 3415
7        66, 9*  66,   2026, 4*2026,   2028, 4*2028,    129, 4* 129,MM 3416
8       151, 4* 151,    185, 4* 185,   2090, 3*2090,   2132, 1*2132,MM 3417
9      2153, 1*2153,   2172, 1*2172,   2191, 1*2191,   2205, 1*2205,MM 3418
1      2219, 1*2219,   2237, 3*2237,    698, 4* 698,    880, 4* 880,MM 3419
2      1060, 4*1060,   1207, 4*1207,   1377, 4*1377,   1482, 4*1482,MM 3420
3      1584, 9*1584,   1635, 4*1635,     66, 9*  66,   2026, 4*2026,MM 3421
4      2028, 4*2028,    129, 4* 129,    151, 4* 151,    185, 4* 185,MM 3422
5      2090, 3*2090,   2132, 1*2132,   2153, 1*2153,   2172, 1*2172,MM 3423
6      2191, 1*2191,   2205, 1*2205,   2219, 1*2219,   2237, 3*2237,MM 3424
7       698, 4* 698,    880, 4* 880,   1060, 4*1060,   1207, 4*1207,MM 3425
8      1377, 4*1377,   1482, 4*1482,   1584, 9*1584,   1635, 4*1635,MM 3426
9        66, 9*  66,   2026, 4*2026,   2028, 4*2028,    129, 4* 129/MM 3427
 DATA A7/                                                       MM    3428
1       151, 4* 151,    185, 4* 185,   2090, 3*2090,   2133, 1*2133,MM 3429
2      2154, 1*2154,   2173, 1*2173,   2192, 1*2192,   2206, 1*2206,MM 3430
3      2220, 1*2220,   2237, 3*2237,    698, 4* 698,    880, 4* 880,MM 3431
4      1060, 4*1060,   1207, 4*1207,   1377, 4*1377,   1482, 4*1482,MM 3432
5      1584, 9*1584,   1635, 4*1635,     66, 9*  66,   2026, 4*2026,MM 3433
6      2028, 4*2028,    129, 4* 129,    151, 4* 151,    185, 4* 185,MM 3434
7      2090, 3*2090,   2133, 1*2133,   2154, 1*2154,   2173, 1*2173,MM 3435
8      2192, 1*2192,   2206, 1*2206,   2220, 1*2220,   2237, 3*2237,MM 3436
9       698, 4* 698,    880, 4* 880,   1060, 4*1060,   1207, 4*1207,MM 3437
1      1377, 4*1377,   1482, 4*1482,   1584, 9*1584,   1635, 4*1635,MM 3438
2        66, 9*  66,   2026, 4*2026,   2028, 4*2028,    129, 4* 129,MM 3439
3       151, 4* 151,    185, 4* 185,   2091, 1*2091,   2112, 1*2112,MM 3440
4      2134, 1*2134,   2155, 1*2155,   2174, 1*2174,   2193, 1*2193,MM 3441
5      2207, 1*2207,   2221, 1*2221,   2238, 1*2238,   2248, 1*2248,MM 3442
6       698, 4* 698,    880, 4* 880,   1060, 4*1060,   1207, 4*1207,MM 3443
```

```
     7     1377, 4*1377,   1482, 4*1482,   1584, 9*1584,   1635, 4*1635,MM  3444
     8       67, 9*  67,   2027, 4*2027,   2029, 4*2029,    130, 4* 130,MM  3445
     9     2039, 1*2039,   2048, 1*2048,   2057, 1*2057,   2065, 1*2065/MM  3446
      DATA A8/                                                          MM  3447
     1     2075, 1*2075,   2091, 1*2091,   2112, 1*2112,   2134, 1*2134,MM  3448
     2     2155, 1*2155,   2174, 1*2174,   2193, 1*2193,   2207, 1*2207,MM  3449
     3     2221, 1*2221,   2238, 1*2238,   2248, 1*2248,    699, 4* 699,MM  3450
     4      881, 1* 881,    948,    990, 1* 990,   1061, 4*1061,   1208,MM  3451
     5  9*1208,   1483, 4*1483,   1584, 9*1584,   1636, 4*1636,     67,MM  3452
     6  9*  67,   2027, 4*2027,   2029, 4*2029,    130, 4* 130,   2039,MM  3453
     7  1*2039,   2048, 1*2048,   2057, 1*2057,   2065, 1*2065,   2075,MM  3454
     8  1*2075,   2092, 1*2092,   2113, 1*2113,   2135, 1*2135,   2156,MM  3455
     9  1*2156,   2175, 1*2175,   2194, 1*2194,   2208, 1*2208,   2222,MM  3456
     1  1*2222,   2239, 1*2239,   2249, 1*2249,    699, 4* 699,    881,MM  3457
     2  1* 881,    949,    990, 1* 990,   1061, 4*1061,   1208, 9*1208,MM  3458
     3    1483, 4*1483,   1584, 9*1584,   1636, 4*1636,     67, 9*  67,MM  3459
     4    2027, 4*2027,   2029, 4*2029,    130, 4* 130,   2040, 1*2040,MM  3460
     5    2049, 1*2049,   2058, 1*2058,   2066, 1*2066,   2076, 1*2076,MM  3461
     6    2092, 1*2092,   2113, 1*2113,   2135, 1*2135,   2156, 1*2156,MM  3462
     7    2175, 1*2175,   2194, 1*2194,   2208, 1*2208,   2222, 1*2222,MM  3463
     8    2239, 1*2239,   2249, 1*2249,    699, 4* 699,    882,    922,MM  3464
     9     950,    991,   1027,   1061, 4*1061,   1208, 9*1208,   1483/MM  3465
      END                                                               MM  3466
```

```
      BLOCK DATA                                                        MM  3467
C                                                                       MM  3468
C         VARIABLE SIZE GRID ID NUMBERS  FOR ST. LOUIS                  MM  3469
C         BLOCK DATA PROGRAM NUMBER TWO                                 MM  3470
      INTEGER A9,A10,A11,A12,A13,A14,A15,A16,A17,A18,A19                 MM  3471
      DIMENSION   A9(265),                                              MM  3472
     2            A10(254), A11(255), A12(229), A13(242), A14(211),    MM  3473
     3            A15(203), A16(202), A17(202), A18(202), A19(212)     MM  3474
      COMMON /VGRID/  ITGRID,   IVG(100,100)                            MM  3475
      EQUIVALENCE (A9(1),IVG(82,28))                                    MM  3476
      EQUIVALENCE (A10(1),IVG(47,31))                                   MM  3477
      EQUIVALENCE (A11(1),IVG(1,34))                                    MM  3478
      EQUIVALENCE (A12(1),IVG(56,36))                                   MM  3479
      EQUIVALENCE (A13(1),IVG(85,38))                                   MM  3480
      EQUIVALENCE (A14(1),IVG(27,41))                                   MM  3481
      EQUIVALENCE (A15(1),IVG(38,43))                                   MM  3482
      EQUIVALENCE (A16(1),IVG(41,45))                                   MM  3483
      EQUIVALENCE (A17(1),IVG(43,47))                                   MM  3484
      EQUIVALENCE (A18(1),IVG(45,49))                                   MM  3485
      EQUIVALENCE (A19(1),IVG(47,51))                                   MM  3486
C                                                                       MM  3487
      DATA A9/                                                          MM  3488
     1  4*1483,   1584, 9*1584,   1636, 4*1636,     67, 9*  67,   2027,MM  3489
     2  4*2027,   2029, 4*2029,    130, 4* 130,   2040, 1*2040,   2049,MM  3490
     3  1*2049,   2058, 1*2058,   2066, 1*2066,   2076, 1*2076,   2093,MM  3491
     4  1*2093,   2114, 1*2114,   2136, 1*2136,   2157, 1*2157,   2176,MM  3492
     5  1*2176,   2195, 1*2195,   2209, 1*2209,   2223, 1*2223,   2240,MM  3493
     6  1*2240,   2250, 1*2250,    699, 4* 699,    883, 1* 883,    951,MM  3494
     7     992,   1028,   1061, 4*1061,   1208, 9*1208,   1483, 4*1483,MM  3495
```

```
8      1584, 9*1584,    1636, 4*1636,      67, 9*  67,    2027, 4*2027,MM  3496
9      2029, 4*2029,     130, 4* 130,    2041, 1*2041,    2050, 1*2050,MM  3497
1      2059, 1*2059,    2067, 1*2067,    2077, 1*2077,    2093, 1*2093,MM  3498
2      2114, 1*2114,    2136, 1*2136,    2157, 1*2157,    2176, 1*2176,MM  3499
3      2195, 1*2195,    2209, 1*2209,    2223, 1*2223,    2240, 1*2240,MM  3500
4      2250, 1*2250,     699, 4* 699,     883, 1* 883,     952,     993,MM  3501
5      1029,    1061, 4*1061,    1208, 9*1208,    1483, 4*1483,    1584,MM  3502
6    9*1584,    1636, 4*1636,      67, 9*  67,      85, 4*  85,    2030,MM  3503
7    4*2030,     131, 4* 131,    2041, 1*2041,    2050, 1*2050,    2059,MM  3504
8    1*2059,    2067, 1*2067,    2077, 1*2077,    2094, 1*2094,    2115,MM  3505
9    1*2115,    2137, 1*2137,    2158, 1*2158,    2177, 1*2177,    2196/MM  3506
 DATA A10/                                                            MM  3507
1    1*2196,    2210, 1*2210,    2224, 1*2224,    2241, 1*2241,    2251,MM  3508
2    1*2251,    2256, 1*2256,    2261, 1*2261,    2262, 1*2262,    2270,MM  3509
3    1*2270,    2279, 1*2279,    1062, 4*1062,    1208, 9*1208,    1484,MM  3510
4    4*1484,    1585, 4*1585,    1619, 4*1619,    1637, 4*1637,      67,MM  3511
5    9*  67,      85, 4*  85,    2030, 4*2030,     131, 4* 131,    2042,MM  3512
6    1*2042,    2051, 1*2051,    2060, 1*2060,    2068, 3*2068,    2094,MM  3513
7    1*2094,    2115, 1*2115,    2137, 1*2137,    2158, 1*2158,    2177,MM  3514
8    1*2177,    2196, 1*2196,    2210, 1*2210,    2224, 1*2224,    2241,MM  3515
9    1*2241,    2251, 1*2251,    2256, 1*2256,    2261, 1*2261,    2262,MM  3516
1    1*2262,    2270, 1*2270,    2279, 1*2279,    1062, 4*1062,    1208,MM  3517
2    9*1208,    1484, 4*1484,    1585, 4*1585,    1619, 4*1619,    1637,MM  3518
3    4*1637,      67, 9*  67,      85, 4*  85,    2030, 4*2030,     131,MM  3519
4    4* 131,    2042, 1*2042,    2051, 1*2051,    2060, 1*2060,    2068,MM  3520
5    3*2068,    2095, 1*2095,    2116, 1*2116,    2138, 1*2138,    2159,MM  3521
6    1*2159,    2178, 1*2178,    2197, 1*2197,    2211, 1*2211,    2225,MM  3522
7    1*2225,    2242, 1*2242,    2252, 1*2252,    2257,    2260, 2*2260,MM  3523
8      2263, 2*2263,    2273, 2*2273,    1062, 4*1062,    1208, 9*1208,MM  3524
9      1484, 4*1484,    1585, 4*1585,    1619, 4*1619,    1637, 4*1637/MM  3525
 DATA A11/                                                            MM  3526
1       67, 9*  67,      85, 4*  85,    2030, 4*2030,     131, 4* 131,MM  3527
2      2043, 1*2043,    2052, 1*2052,    2061, 1*2061,    2068, 3*2068,MM  3528
3      2095, 1*2095,    2116, 1*2116,    2138, 1*2138,    2159, 1*2159,MM  3529
4      2178, 1*2178,    2197, 1*2197,    2211, 1*2211,    2225, 1*2225,MM  3530
5      2242, 1*2242,    2252, 1*2252,    2258,    2260, 2*2260,    2263,MM  3531
6    2*2263,    2273, 2*2273,    1062, 4*1062,    1208, 9*1208,    1484,MM  3532
7    4*1484,    1585, 4*1585,    1619, 4*1619,    1637, 4*1637,      67,MM  3533
8    9*  67,      85, 4*  85,    2030, 4*2030,     131, 4* 131,    2043,MM  3534
9    1*2043,    2052, 1*2052,    2061, 1*2061,    2068, 3*2068,    2096,MM  3535
1    1*2096,    2117, 1*2117,    2139, 1*2139,    2160, 1*2160,    2179,MM  3536
2    1*2179,    2198, 1*2198,    2212, 1*2212,    2226, 1*2226,    2243,MM  3537
3    1*2243,    2253, 1*2253,    2259,    2260, 2*2260,    2263, 2*2263,MM  3538
4      2273, 2*2273,    1062, 4*1062,    1208, 9*1208,    1484, 4*1484,MM  3539
5      1585, 4*1585,    1619, 4*1619,    1637, 4*1637,      68, 4*  68,MM  3540
6        73, 4*  73,      86, 4*  86,     103, 4* 103,     132, 4* 132,MM  3541
7       156, 4* 156,     189, 4* 189,    2096, 1*2096,    2117, 1*2117,MM  3542
8      2139, 1*2139,    2160, 1*2160,    2179, 1*2179,    2198, 1*2198,MM  3543
9      2212, 1*2212,    2226, 1*2226,    2243, 1*2243,    2253, 1*2253/MM  3544
 DATA A12/                                                            MM  3545
1       702,     741,     776,     813, 1* 813,     886, 2* 886,     994,MM  3546
2    1* 994,    1063, 4*1063,    2304, 2*2304,    2313, 1*2313,    1378,MM  3547
3    3*1378,    1459,    1485,    1508,    1517,    1539,    1559,    1586,MM  3548
4    4*1586,    1620, 4*1620,    1637, 4*1637,      68, 4*  68,      73,MM  3549
5    4*  73,      86, 4*  86,     103, 4* 103,     132, 4* 132,     156,MM  3550
```

```
6    4* 156,     189, 4* 189,    2097, 1*2097,    2118, 1*2118,    2140,MM  3551
7    1*2140,    2161, 3*2161,    2199, 1*2199,    2213, 1*2213,    2227,MM  3552
8    1*2227,    2244, 1*2244,    2254, 1*2254,     703,     742,     777,MM  3553
9       813, 1* 813,     886, 2* 886,     994,  * 994,    1063, 4*1063,MM  3554
1      2304, 2*2304,    2313, 1*2313,    1378,  1378,    1460,    1486,MM  3555
2      1509,    1518,    1540,    1560,    1586, 4 1586,    1620, 4*1620,MM  3556
3      1637, 4*1637,      68, 4*  68,      73, 4*  73,      86, 4*  86,MM  3557
4       103, 4* 103,     132, 4* 132,     156, 4* 156,     189, 4* 189,MM  3558
5      2097, 1*2097,    2118, 1*2118,    2140, 1*2140,    2161, 3*2161,MM  3559
6      2199, 1*2199,    2213, 1*2213,    2227, 1*2227,    2244, 1*2244,MM  3560
7      2254, 1*2254,     704,     743,     778,     814,     846,     886,MM  3561
8    2* 886,     995, 1* 995,    1063, 4*1063,    2304, 2*2304,    2314,MM  3562
9      2318,    1378, 3*1378,    1461,    1487,    1510,    1519,    1541/MM  3563
 DATA A13/                                                              MM  3564
1      1561,    1586, 4*1586,    1620, 4*1620,    1637, 4*1637,      68,MM  3565
2    4*  68,      73, 4*  73,      86, 4*  86,     103, 4* 103,     132,MM  3566
3    4* 132,     156, 4* 156,     189, 4* 189,    2098, 1*2098,    2119,MM  3567
4    1*2119,    2141, 1*2141,    2161, 3*2161,    2200, 1*2200,    2214,MM  3568
5    1*2214,    2228, 1*2228,    2245, 1*2245,    2255, 1*2255,     705,MM  3569
6       744,     779,     815,     847,     887, 2* 887,     995, 1* 995,MM  3570
7      1063, 4*1063,    2305, 1*2305,    2310, 1*2310,    2319,    1378,MM  3571
8    3*1378,    1462,    1488,    1511,    1520,    1542,    1562,    1586,MM  3572
9    4*1586,    1620, 4*1620,    1637, 4*1637,      68, 4*  68,      73,MM  3573
1    4*  73,      86, 4*  86,     103, 4* 103,     132, 4* 132,     156,MM  3574
2    4* 156,     189, 4* 189,    2098, 1*2098,    2119, 1*2119,    2141,MM  3575
3    1*2141,    2161, 3*2161,    2200, 1*2200,    2214, 1*2214,    2228,MM  3576
4    1*2228,    2245, 1*2245,    2255, 1*2255,     706,     745,     780,MM  3577
5       816,     848,     887, 2* 887,     996, 1* 996,    1063, 4*1063,MM  3578
6      2305, 1*2305,    2310, 1*2310,    2320,    1379,    1410,    1431,MM  3579
7      1449,    1463,    1489,    1512,    1521,    1543,    1563,    1586,MM  3580
8    4*1586,    1620, 4*1620,    1637, 4*1637,      69, 4*  69,      74,MM  3581
9    4*  74,      87, 4*  87,     104, 4* 104,     133, 4* 133,    2044/MM  3582
 DATA A14/                                                              MM  3583
1    4*2044,     190, 4* 190,     226, 1* 226,     251, 1* 251,     281,MM  3584
2    1* 281,     310, 1* 310,     327,     337,     366,     401,     430,MM  3585
3       472, 1* 472,     540, 1* 540,     597,     635,     665,     707,MM  3586
4       746,     781,     817,     849,     887, 2* 887,     996, 1* 996,MM  3587
5      2284, 1*2284,    2290,    2296, 1*2296,    1211, 1*1211,    1274,MM  3588
6    2*1274,    1380,    1411,    1432,    1450,    1464,    2337, 1*2337,MM  3589
7      2349, 1*2349,    2369, 1*2369,    2388, 3*2388,    1621, 4*1621,MM  3590
8      1638, 4*1638,      69, 4*  69,      74, 4*  74,      87, 4*  87,MM  3591
9       104, 4* 104,     133, 4* 133,    2044, 4*2044,     190, 4* 190,MM  3592
1       226, 1* 226,     251, 1* 251,     281, 1* 281,     310, 1* 310,MM  3593
2       328,     338,     367,     402,     431,     472, 1* 472,     540,MM  3594
3    1* 540,     598,     636,     666,     708,     747,     782,     818,MM  3595
4       850,     888,     923,     953,     997, 1* 997,    2284, 1*2284,MM  3596
5      2291,    2296, 1*2296,    1211, 1*1211,    1274, 2*1274,    1381,MM  3597
6      1412,    1433,    1451,    1465,    2337, 1*2337,    2349, 1*2349,MM  3598
7      2369, 1*2369,    2388, 3*2388,    1621, 4*1621,    1638, 4*1638,MM  3599
8        69, 4*  69,      74, 4*  74,      87, 4*  87,     104, 4* 104,MM  3600
9       133, 4* 133,    2044, 4*2044,     190, 4* 190,     227, 1* 227/MM  3601
 DATA A15/                                                              MM  3602
1       252, 1* 252,     282, 1* 282,     311, 1* 311,     329,     339,MM  3603
2       368,     403,     432,     473,     504,     541,     571,     599,MM  3604
3       637,     667,     709,     748,     783,     819,     851,     889,MM  3605
```

```
4      924,    954,    997, 1* 997,   2285,   2289,   2292,   2297,MM  3606
5     2301,   1212,   1246,   1274, 2*1274,   1382,   1413,   1434,MM  3607
6     1452,   1466,   2338, 1*2338,   2350, 1*2350,   2370, 1*2370,MM  3608
7     2388, 3*2388,   1621, 4*1621,   1638, 4*1638,     69, 4*  69,MM  3609
8       74, 4*  74,     87, 4*  87,    104, 4* 104,    133, 4* 133,MM  3610
9     2044, 4*2044,    190, 4* 190,    227, 1* 227,    252, 1* 252,MM  3611
1      282, 1* 282,    311, 1* 311,    330,    340,    369,    404,MM  3612
2      433,    474,    505,    542,    572,    600,    638,    668,MM  3613
3      710,    749,    784,    820,    852,    890,    925,    955,MM  3614
4      998, 1* 998,   2286, 1*2286,   2293,   2298, 1*2298,   1213,MM  3615
5     1247,   1275,   1308,   1340,   1383,   1414,   1435,   1453,MM  3616
6     1467,   2338, 1*2338,   2350, 1*2350,   2370, 1*2370,   2388,MM  3617
7   3*2388,   1621, 4*1621,   1638, 4*1638,     69, 4*  69,     74,MM  3618
8   4*  74,     87, 4*  87,    104, 4* 104,    133, 4* 133,   2044,MM  3619
9   4*2044,    190, 4* 190,    228, 1* 228,    253, 1* 253,    283/MM  3620
 DATA A16/                                                        MM  3621
1   1* 283,    312, 1* 312,    331,    341,    370,    405,    434,MM  3622
2      475,    506,    543,    573,    601,    639,    669,    711,MM  3623
3      750,    785,    821,    853,    891,    926,    956,    998,MM  3624
4   1* 998,   2286, 1*2286,   2294,   2298, 1*2298,   1214,   1248,MM  3625
5     1276,   1309,   1341,   1384,   1415,   1436,   1454,   1468,MM  3626
6     2339, 1*2339,   2351, 1*2351,   2371, 1*2371,   2389, 1*2389,MM  3627
7     2400, 1*2400,   1621, 4*1621,   1638, 4*1638,     60, 4*  60,MM  3628
8       75, 9*  75,    105, 4* 105,    134, 4* 134,    159, 4* 159,MM  3629
9      191, 4* 191,    228, 1* 228,    253, 1* 253,    283, 1* 283,MM  3630
1      312, 1* 312,    332,    342,    371,    406,    435,    476,MM  3631
2      507,    544,    574,    602,    640,    670,    712,    751,MM  3632
3      786,    822,    854,    892,    927,    957,    999,   1030,MM  3633
4     1066,   1095,   1118,   1150,   1179,   1215,   1249,   1277,MM  3634
5     1310,   1342,   1385,   1416,   1437,   1455, 1*1455,   2339,MM  3635
6   1*2339,   2351, 1*2351,   2371, 1*2371,   2389, 1*2389,   2400,MM  3636
7   1*2400,   1622, 4*1622,   1639, 4*1639,     60, 4*  60,     75,MM  3637
8   9*  75,    105, 4* 105,    134, 4* 134,    159, 4* 159,    191,MM  3638
9   4* 191,    229, 1* 229,    254, 1* 254,    284, 1* 284,    313/MM  3639
 DATA A17/                                                        MM  3640
1   1* 313,    333,    343,    372,    407,    436,    477,    508,MM  3641
2      545,    575,    603,    641,    671,    713,    752,    787,MM  3642
3      823,    855,    893,    928,    958,   1000,   1031,   1067,MM  3643
4     1096,   1119,   1151,   1180,   1216,   1250,   1278,   1311,MM  3644
5     1343,   1386,   1417,   1438,   1455, 1*1455,   2340, 1*2340,MM  3645
6     2352, 1*2352,   2372, 1*2372,   2390, 1*2390,   2401, 1*2401,MM  3646
7     1622, 4*1622,   1639, 4*1639,     60, 4*  60,     75, 9*  75,MM  3647
8      105, 4* 105,    134, 4* 134,    159, 4* 159,    191, 4* 191,MM  3648
9      229, 1* 229,    254, 1* 254,    284, 1* 284,    313, 1* 313,MM  3649
1      334,    344,    373,    408,    437,    478,    509,    546,MM  3650
2      576,    604,    642,    672,    714,    753,    788,    824,MM  3651
3      856,    894,    929,    959,   1001,   1032,   1068,   1097,MM  3652
4     1120,   1152,   1181,   1217,   1251,   1279,   1312,   1344,MM  3653
5     1387,   1418,   1439,   1456, 1*1456,   2340, 1*2340,   2352,MM  3654
6   1*2352,   2372, 1*2372,   2390, 1*2390,   2401, 1*2401,   1622,MM  3655
7   4*1622,   1639, 4*1639,     60, 4*  60,     75, 9*  75,    105,MM  3656
8   4* 105,    134, 4* 134,    159, 4* 159,    191, 4* 191,    230,MM  3657
9   1* 230,    255, 1* 255,    285, 1* 285,    314, 1* 314,    335/MM  3658
 DATA A18/                                                        MM  3659
1      345,    374,    409,    438,    479,    510,    547,    577,MM  3660
```

```
2      605,    643,    673,    715,    754,    789,    825,    857,MM  3661
3      895,    930,    960,   1002,   1033,   1069,   1098,   1121,MM  3662
4     1153,   1182,   1218,   1252,   1280,   1313,   1345,   1388,MM  3663
5   1*1388,   1440,   1456, 1*1456,   2341, 1*2353,   2353, 1*2353,MM  3664
6     2373, 1*2373,   2391, 1*2391,   2402, 1*2[illegible]2,   1622, 4*1622,MM  3665
7     1639, 4*1639,     60, 4*  60,     75, 9*  75,    105, 4* 105,MM  3666
8      134, 4* 134,    159, 4* 159,    191, 4* 19[illegible]    230, 1* 230,MM  3667
9      255, 1* 255,    285, 1* 285,    314, 1* 314,    336,    346,MM  3668
1      375,    410,    439,    480,    511,    548,    578,    606,MM  3669
2      644,    674,    716,    755,    790,    826,    858,    896,MM  3670
3      931,    961,   1003,   1034,   1070,   1099,   1122,   1154,MM  3671
4     1183,   1219,   1253,   1281,   1314,   1346,   1388, 1*1388,MM  3672
5     1441,   1457,   1469,   2341, 1*2341,   2353, 1*2353,   2373,MM  3673
6   1*2373,   2391, 1*2391,   2402, 1*2402,   1622, 4*1622,   1639,MM  3674
7   4*1639,     60, 4*  60,     75, 9*  75,    106, 4* 106,    135,MM  3675
8   4* 135,    160, 4* 160,    192, 4* 192,   2099,   2109,   2120,MM  3676
9     2130,   2142,   2150,   2162,   2170,   2180,   2189,    376/MM  3677
 DATA A19/                                                        MM  3678
1      411,    440,    481,    512,    549,    579,    607,    645,MM  3679
2      675,    717,    756,    791,    827,    859,    897,    932,MM  3680
3      962,   1004,   1035,   1071,   1100,   1123, 2*1123,   1220,MM  3681
4     1254,   1282,   1315,   1347,   2322,   2327, 1*2327,   2331,MM  3682
5   1*2331,   1494, 2*1494,   1544,   1564,   1590, 4*1590,   1623,MM  3683
6   4*1623,   1640, 4*1640,     60, 4*  60,     75, 9*  75,    106,MM  3684
7   4* 106,    135, 4* 135,    160, 4* 160,    192, 4* 192,   2100,MM  3685
8   1*2100,   2121, 1*2121,   2143, 1*2143,   2163, 1*2163,   2181,MM  3686
9   1*2181,    377,    412,    441,    482,    513,    550,    580,MM  3687
1      608,    646,    676,    718,    757,    792,    828,    860,MM  3688
2      898,    933,    963,   1005,   1036,   1072,   1101,   1123,MM  3689
3   2*1123,   1221,   1255,   1283,   1316,   1348,   2323,   2327,MM  3690
4   1*2327,   2331, 1*2331,   1494, 2*1494,   1545,   1565,   1590,MM  3691
5   4*1590,   1623, 4*1623,   1640, 4*1640,     60, 4*  60,     75,MM  3692
6   9*  75,    106, 4* 106,    135, 4* 135,    160, 4* 160,    192,MM  3693
7   4* 192,   2100, 1*2100,   2121, 1*2121,   2143, 1*2143,   2163,MM  3694
8   1*2163,   2181, 1*2181,    378,    413,    442,    483,    514,MM  3695
9      551,    581,    609,    647,    677,    719,    758,    793/MM  3696
 END                                                              MM  3697
```

```
      BLOCK DATA                                                  MM  3698
C                                                                 MM  3699
C         BLOCK DATA PROGRAM NUMBER THREE                         MM  3700
C         VARIABLE SIZE GRID ID NUMBERS FOR ST. LOUIS             MM  3701
      INTEGER A20,A21,A22,A23,A24,A25,A26,A27,A28,A29,A30          MM  3702
      DIMENSION   A20(212), A21(206), A22(210), A23(235), A24(230),MM  3703
     5            A25(220), A26(219), A27(224), A28(266), A29(281),MM  3704
     6            A30(317)                                        MM  3705
      COMMON /VGRID/  ITGRID,   IVG(100,100)                      MM  3706
      EQUIVALENCE (A20(1),IVG(59,53))                             MM  3707
      EQUIVALENCE (A21(1),IVG(71,55))                             MM  3708
      EQUIVALENCE (A22(1),IVG(77,57))                             MM  3709
      EQUIVALENCE (A23(1),IVG(87,59))                             MM  3710
      EQUIVALENCE (A24(1),IVG(22,62))                             MM  3711
      EQUIVALENCE (A25(1),IVG(52,64))                             MM  3712
```

```
      EQUIVALENCE (A26(1),IVG(72,66))                                    MM  3713
      EQUIVALENCE (A27(1),IVG(91,68))                                    MM  3714
      EQUIVALENCE (A28(1),IVG(15,71))                                    MM  3715
      EQUIVALENCE (A29(1),IVG(81,73))                                    MM  3716
      EQUIVALENCE (A30(1),IVG(62,76))                                    MM  3717
C                                                                        MM  3718
      DATA A20/                                                          MM  3719
     1      829,    861,    899,    934,    964,   1006,   1037,   1073,MM  3720
     2     1102,   1123, 2*1123,   1222, 2*1222,   1317,   1349,   2324,MM  3721
     3     2328,   2330,   2332,   2334,   1494, 2*1494,   1546,   1566,MM  3722
     4     1590, 4*1590,   1623, 4*1623,   1640, 4*1640,     60, 4*  60,MM  3723
     5       75, 9*  75,    106, 4* 106,    135, 4* 135,    160, 4* 160,MM  3724
     6      192, 4* 192,   2101, 1*2101,   2122, 1*2122,   2144, 1*2144,MM  3725
     7     2164, 1*2164,   2182, 1*2182,    379,    414,    443,    484,MM  3726
     8      515,    552,    582,    610,    648,    678,    720,    759,MM  3727
     9      794,    830,    862,    900,    935,    965,   1007,   1038,MM  3728
     1     1074,   1103,   1124,   1155,   1184,   1222, 2*1222,   1318,MM  3729
     2     1350,   2325,   2329, 1*2329,   2333, 1*2333,   1495,   1513,MM  3730
     3     1522,   1547,   1567,   1590, 4*1590,   1623, 4*1623,   1640,MM  3731
     4  4*1640,     60, 4*  60,     75, 9*  75,    106, 4* 106,    135,MM  3732
     5  4* 135,    160, 4* 160,    192, 4* 192,   2101, 1*2101,   2122,MM  3733
     6  1*2122,   2144, 1*2144,   2164, 1*2164,   2182, 1*2182,    380,MM  3734
     7      415,    444,    485,    516,    553,    583,    611,    649,MM  3735
     8      679,    721,    760,    795,    831,    863,    901,    936,MM  3736
     9      966,   1008,   1039,   1075,   1104,   1125,   1156,   1185/MM  3737
      DATA A21/                                                          MM  3738
     1     1222, 2*1222,   1319,   1351,   2326,   2329, 1*2329,   2333,MM  3739
     2  1*2333,   1496,   1514,   1523,   1548,   1568,   1590, 4*1590,MM  3740
     3     1623, 4*1623,   1640, 4*1640,     70, 4*  70,     76, 9*  76,MM  3741
     4      107, 4* 107,    136, 4* 136,   2045, 2*2045,   2055, 2*2055,MM  3742
     5     2069, 2*2069,   2083,   2102, 1*2102,   2123, 1*2123,   2145,MM  3743
     6  1*2145,   2165, 1*2165,   2183, 1*2183,    381,    416,    445,MM  3744
     7      486,    517,    554,    584,    612,    650,    680,    722,MM  3745
     8      761,    796,    832,    864,    902,    937,    967,   1009,MM  3746
     9     1040,   1076,   1105,   1126,   1157,   1186,   1223,   1256,MM  3747
     1     1284,   1320,   1352,   1391, 4*1391,   1497,   1515,   1524,MM  3748
     2     1549,   1569,   1591, 4*1591,   1624, 4*1624,   1641, 4*1641,MM  3749
     3       70, 4*  70,     76, 9*  76,    107, 4* 107,    136, 4* 136,MM  3750
     4     2045, 2*2045,   2055, 2*2055,   2069, 2*2069,   2084,   2102,MM  3751
     5  1*2102,   2123, 1*2123,   2145, 1*2145,   2165, 1*2165,   2183,MM  3752
     6  1*2183,    382,    417,    446,    487,    518,    555,    585,MM  3753
     7      613,    651,    681,    723,    762,    797,    833,    865,MM  3754
     8      903,    938,    968,   1010, 1*1010,   1077,   1106,   1127,MM  3755
     9     1158,   1187,   1224,   1257,   1285,   1321,   1353,   1391/MM  3756
      DATA A22/                                                          MM  3757
     1  4*1391,   1498,   1516,   1525,   1550,   1570,   1591, 4*1591,MM  3758
     2     1624, 4*1624,   1641, 4*1641,     70, 4*  70,     76, 9*  76,MM  3759
     3      107, 4* 107,    136, 4* 136,   2045, 2*2045,   2055, 2*2055,MM  3760
     4     2069, 2*2069,   2085,   2103, 1*2103,   2124, 1*2124,   2146,MM  3761
     5  1*2146,   2166, 1*2166,   2184, 1*2184,    383,    418,    447,MM  3762
     6      488,    519,    556,    586,    614,    652,    682,    724,MM  3763
     7      763,    798,    834,    866,    904,    939,    969,   1010,MM  3764
     8  1*1010,   1078,   1107,   1128,   1159,   1188,   1225,   1258,MM  3765
     9     1286,   1322,   1354,   1391, 4*1391,   1499, 2*1499,   1551,MM  3766
     1     1571,   1591, 4*1591,   1624, 4*1624,   1641, 4*1641,     70,MM  3767
```

```
2    4*  70,     76, 9*  76,    107, 4* 107,    136, 4* 136,   2046,MM  3768
3    1*2046,  2053, 1*2053,  2062, 1*2062,  2070, 1*2070,   2078,MM  3769
4    1*2078,  2103, 1*2103,  2124, 1*2124,  2146, 1*2146,   2166,MM  3770
5    1*2166,  2184, 1*2184,   384,    419,    448,    489,    520,MM  3771
6       557,   587,    615,   653,    683,    727,    764,    799,MM  3772
7       835,   867,    905,   940,    970,   1011, 1*1011,   1079,MM  3773
8    1*1079,  1129,   1160,  1189,   1226,   1259,   1287,   1323,MM  3774
9       1355,  1391, 4*1391,  1499, 2*1499,  1552,   1572,   1591/MM  3775
 DATA A23/                                                        MM  3776
1    4*1591,  1624, 4*1624,  1641, 4*1641,     70, 4*  70,     76,MM  3777
2    9*  76,   107, 4* 107,   136, 4* 136,   2046, 1*2046,   2053,MM  3778
3    1*2053,  2062, 1*2062,  2070, 1*2070,  2078, 1*2078,   2104,MM  3779
4    1*2104,  2125, 1*2125,  2147, 1*2147,  2167, 1*2167,   2185,MM  3780
5    1*2185,   385,    420,   449,    490,    521,    558,    588,MM  3781
6       616,   654,    684,   726,    765,    800,    836,    868,MM  3782
7       906,   941,    971,  1011, 1*1011,  1079, 1*1079,   1130,MM  3783
8       1161,  1190,   1227,  1260,   1288,   1324,   1356,   1391,MM  3784
9    4*1391,  1499, 2*1499,  1553,   1573,   1591, 4*1591,   1624,MM  3785
1    4*1624,  1641, 4*1641,    71, 4*  71,     76, 9*  76,    108,MM  3786
2    4* 108,   137, 4* 137,   163, 4* 163,    195, 4* 195,   2104,MM  3787
3    1*2104,  2125, 1*2125,  2147, 1*2147,  2167, 1*2167,   2185,MM  3788
4    1*2185,   386,    421,   450,    491,    522,    559,    589,MM  3789
5       617,   655,    685,   727,    766,    801,    837,    869,MM  3790
6       907,   942,    972,  1012,   1041,   1080, 1*1080,   1131,MM  3791
7       1162,  1191,   1228,  1261,   1289,   1325,   1357,   1392,MM  3792
8    4*1392,  1500, 4*1500,  1592, 4*1592,  1625, 4*1625,   1641,MM  3793
9    4*1641,    71, 4*  71,    76, 9*  76,    108, 4* 108,    137/MM  3794
 DATA A24/                                                        MM  3795
1    4* 137,   163, 4* 163,   195, 4* 195,   2105, 1*2105,   2126,MM  3796
2    1*2126,  2148, 1*2148,  2168, 1*2168,  2186, 1*2186,    387,MM  3797
3       422,   451,    492,   523,    560,    590,    618,    656,MM  3798
4       686,   728,    767,   802,    838,    870,    908,    943,MM  3799
5       973,  1013,   1042,  1080, 1*1080,  1132,   1163,   1192,MM  3800
6       1229,  1262,   1290,  1326,   1358,   1392, 4*1392,   1500,MM  3801
7    4*1500,  1592, 4*1592,  1625, 4*1625,  1641, 4*1641,     71,MM  3802
8    4*  71,    76, 9*  76,   108, 4* 108,    137, 4* 137,    163,MM  3803
9    4* 163,   195, 4* 195,  2105, 1*2105,  2126, 1*2126,   2148,MM  3804
1    1*2148,  2168, 1*2168,  2186, 1*2186,   388,    423,    452,MM  3805
2       493,   524,    561,   591,    619,    657,    687,    729,MM  3806
3       768,   803,    839,   871,    909,    944,    974,   1014,MM  3807
4       1043,  1081, 2*1081,  1164,   1193,   1230,   1263,   1291,MM  3808
5       1327,  1359,   1392, 4*1392,  1500, 4*1500,   1592, 4*1592,MM  3809
6       1625, 4*1625,  1641, 4*1641,    71, 4*  71,     76, 9*  76,MM  3810
7       108, 4* 108,    137, 4* 137,   163, 4* 163,    195, 4* 195,MM  3811
8       2106, 1*2106,  2127, 1*2127,  2149, 1*2149,   2169, 1*2169,MM  3812
9       2187, 1*2187,   389, 1* 389,   453,    494,    525,    562/MM  3813
 DATA A25/                                                        MM  3814
1       592,   620,    658,   688,    730,    769,    804,    840,MM  3815
2       872,   910,    945,   975,   1015,   1044,   1081, 2*1081,MM  3816
3       1165,  1194,   1231,  1264,   1292,   1328,   1360,   1392,MM  3817
4    4*1392,  1500, 4*1500,  1592, 4*1592,  1625, 4*1625,   1641,MM  3818
5    4*1641,    71, 4*  71,    76, 9*  76,    108, 4* 108,    137,MM  3819
6    4* 137,   163, 4* 163,   195, 4* 195,  2106, 1*2106,   2127,MM  3820
7    1*2127,  2149, 1*2149,  2169, 1*2169,  2187, 1*2187,    389,MM  3821
8    1* 389,   454,    495,   526,    563,    593,    621,    659,MM  3822
```

```
9      689,    731,    770,    805,    841,    873,    911,    946,MM  3823
1      976,   1016,   1045,   1081, 2*1081,   1166,   1195,   1232,MM  3824
2     1265,   1293,   1329,   1361,   1392, 4*1392,   1500, 4*1500,MM  3825
3     1592, 4*1592,   1625, 4*1625,   1641, 4*1641,     61, 4*  61,MM  3826
4       77, 4*  77,     88, 4*  88,    109, 1* 109,    117, 1* 117,MM  3827
5      122,   2031,   2035, 2*2035,   2037, 2*2037,   2054, 2*2054,MM  3828
6      196, 4* 196,    237,    245,    256,    277,    286,    308,MM  3829
7    4* 308,   2201, 1*2201,   2215, 1*2215,   2229,    564,    594,MM  3830
8      622,    660,    690,    732,    771,    806,    842,    874,MM  3831
9     2264, 1*2264,   2274,   2280, 1*2280,   1082, 4*1082,   1233/MM  3832
 DATA A26/                                                        MM  3833
1    4*1233,   1393, 4*1393,   2342, 1*2342,   2354, 1*2354,   2374,MM  3834
2    1*2374,   2392, 1*2392,   2403, 1*2403,   1626, 4*1626,   1642,MM  3835
3    4*1642,     61, 4*  61,     77, 4*  77,     88, 4*  88,    109,MM  3836
4    1* 109,    117, 1* 117,    123,   2032,   2035, 2*2035,   2037,MM  3837
5    2*2037,   2054, 2*2054,    196, 4* 196,    238,    246,    257,MM  3838
6       278,    287,    308, 4* 308,   2201, 1*2201,   2215, 1*2215,MM  3839
7     2230,    565,    595,    623,    661,    691,    733,    772,MM  3840
8      807,    843,    875,   2264, 1*2264,   2275,   2280, 1*2280,MM  3841
9     1082, 4*1082,   1233, 4*1233,   1393, 4*1393,   2342, 1*2342,MM  3842
1     2354, 1*2354,   2374, 1*2374,   2392, 1*2392,   2403, 1*2403,MM  3843
2     1626, 4*1626,   1642, 4*1642,     61, 4*  61,     77, 4*  77,MM  3844
3       88, 4*  88,    110, 2* 110,    118, 1* 118,   2033,   2035,MM  3845
4    2*2035,   2037, 2*2037,   2054, 2*2054,    196, 4* 196,    239,MM  3846
5      247,    258,    279,    288,    308, 4* 308,   2202, 2*2202,MM  3847
6     2216,   2231,    566,    596,    624,    662,    692,    734,MM  3848
7      773,    808,    844,    876,   2265,   2271,   2276, 2*2276,MM  3849
8     1082, 4*1082,   1233, 4*1233,   1393, 4*1393,   2343, 1*2343,MM  3850
9     2355, 1*2355,   2375,   2382,   2393, 1*2393,   2404, 1*2404/MM  3851
 DATA A27/                                                        MM  3852
1     1626, 4*1626,   1642, 4*1642,     61, 4*  61,     77, 4*  77,MM  3853
2       88, 4*  88,    110, 2* 110,    118, 1* 118,   2034, 1*2034,MM  3854
3     2036, 1*2036,   2038, 1*2038,   2047, 1*2047,   2056, 1*2056,MM  3855
4      196, 4* 196,    240, 1* 240,    259,    280, 1* 280,    308,MM  3856
5    4* 308,   2202, 2*2202,   2217, 1*2217,    567, 1* 567,    625,MM  3857
6      663,    693,    735,    774,    809,    845, 1* 845,   2266,MM  3858
7    1*2266,   2276, 2*2276,   1082, 4*1082,   1233, 4*1233,   1393,MM  3859
8    4*1393,   2343, 1*2343,   2355, 1*2355,   2376,   2383,   2393,MM  3860
9    1*2393,   2404, 1*2404,   1626, 4*1626,   1642, 4*1642,     61,MM  3861
1    4*  61,     77, 4*  77,     88, 4*  88,    110, 2* 110,    119,MM  3862
2      124,   2034, 1*2034,   2036, 1*2036,   2038, 1*2038,   2047,MM  3863
3    1*2047,   2056, 1*2056,    196, 4* 196,    240, 1* 240,    260,MM  3864
4      280, 1* 280,    308, 4* 308,   2202, 2*2202,   2217, 1*2217,MM  3865
5      567, 1* 567,    626,    664,    694,    736,    775,    810,MM  3866
6      845, 1* 845,   2266, 1*2266,   2276, 2*2276,   1082, 4*1082,MM  3867
7     1233, 4*1233,   1393, 4*1393,   2344, 1*2344,   2356,   2364,MM  3868
8     2377,   2384,   2394,   2398,   2405, 1*2405,   1626, 4*1626,MM  3869
9     1642, 4*1642,     61, 4*  61,     78, 4*  78,     89, 3*  89/MM  3870
 DATA A28/                                                        MM  3871
1       94, 3*  94,    120, 1* 120,    140, 4* 140,    166, 4* 166,MM  3872
2      197, 4* 197,    241, 4* 241,    309, 4* 309,    392, 4* 392,MM  3873
3      568, 4* 568,    737, 4* 737,    914, 4* 914,   2287, 1*2287,MM  3874
4     2295, 1*2295,   2302, 1*2302,   2308,   1294,   1330,   1362,MM  3875
5     1394,   1419, 3*1419,   2344, 1*2344,   2357,   2365,   2378,MM  3876
6     2385,   2395,   2399,   2405, 1*2405,   1627, 4*1627,   1642,MM  3877
```

```
7    4*1642,     61, 4*  61,     78, 4*  78,     89, 3*  89,     94,MM  3878
8    3*  94,    120, 1* 120,    140, 4* 140,    166, 4* 166,    197,MM  3879
9    4* 197,    241, 4* 241,    309, 4* 309,    392, 4* 392,    568,MM  3880
1    4* 568,    737, 4* 737,    914, 4* 914,   2287, 1*2287,   2295,MM  3881
2    1*2295,   2302, 1*2302,   2309,   1295,   13  ,   1363,   1395,MM  3882
3      1419, 3*1419,   2345, 1*2345,   2358,   23  ,   2379,   2386,MM  3883
4      2396, 1*2396,   2406, 1*2406,   1627, 4*1627,   1642, 4*1642,MM  3884
5        61, 4*  61,     78, 4*  78,     89, 3*  89,     94, 3*  94,MM  3885
6       121, 1* 121,    140, 4* 140,    166, 4* 166,    197, 4* 197,MM  3886
7       241, 4* 241,    309, 4* 309,    392, 4* 392,    568, 4* 568,MM  3887
8       737, 4* 737,    914, 4* 914,   2288, 2*2288,   2299,   2303,MM  3888
9      1235,   1266,   1296,   1332,   1364,   1396,   1419, 3*1419/MM  3889
 DATA A29/                                                        MM  3890
1      2345, 1*2345,   2359,   2367,   2380,   2387,   2396, 1*2396,MM  3891
2      2406, 1*2406,   1627, 4*1627,   1642, 4*1642,     61, 4*  61,MM  3892
3        78, 4*  78,     89, 3*  89,     94, 3*  94,    121, 1* 121,MM  3893
4       140, 4* 140,    166, 4* 166,    197, 4* 197,    241, 4* 241,MM  3894
5       309, 4* 309,    392, 4* 392,    568, 4* 568,    737, 4* 737,MM  3895
6       914, 4* 914,   2288, 2*2288,   2300, 1*2300,   1236,   1267,MM  3896
7      1297,   1333,   1365,   1397,   1419, 3*1419,   2346, 1*2346,MM  3897
8      2360, 1*2360,   2381, 1*2381,   2397, 1*2397,   2407, 1*2407,MM  3898
9      1627, 4*1627,   1642, 4*1642,     61, 4*  61,     78, 4*  78,MM  3899
1        90, 1*  90,     93, 1*  93,     95, 5*  95,    140, 4* 140,MM  3900
2       166, 4* 166,    197, 4* 197,    241, 4* 241,    309, 4* 309,MM  3901
3       392, 4* 392,    568, 4* 568,    737, 4* 737,    914, 4* 914,MM  3902
4      2288, 2*2288,   2300, 1*2300,   1237,   1268,   1298,   1334,MM  3903
5      1366,   1398,   1420,   1442, 2*1442,   2346, 1*2346,   2360,MM  3904
6    1*2360,   2381, 1*2381,   2397, 1*2397,   2407, 1*2407,   1627,MM  3905
7    4*1627,   1642, 4*1642,      0, 9*   0,     90, 1*  90,     93,MM  3906
8    1*  93,     95, 5*  95,    141, 4* 141,      0, 9*   0,    242,MM  3907
9    9* 242,    393, 4* 393,    569, 4* 569,    738, 4* 738,    915/MM  3908
 DATA A30/                                                        MM  3909
1    2* 915,   1017,   1046,   1085,   1108,   1133,   1167,   1196,MM  3910
2      1238,   1269,   1299,   1335,   1367,   1399,   1421,   1442,MM  3911
3    2*1442,   1505, 4*1505,   1595, 9*1595,   1643, 4*1643,      0,MM  3912
4    9*   0,     91, 3*  91,     95, 5*  95,    141, 4* 141,      0,MM  3913
5    9*   0,    242, 9* 242,    393, 4* 393,    569, 4* 569,    738,MM  3914
6    4* 738,    915, 2* 915,   1018,   1047,   1086,   1109,   1134,MM  3915
7      1168,   1197,   1239,   1270,   1300,   1336,   1368,   1400,MM  3916
8      1422,   1442, 2*1442,   1505, 4*1505,   1595, 9*1595,   1643,MM  3917
9    4*1643,      0, 9*   0,     91, 3*  91,     95, 5*  95,    141,MM  3918
1    4* 141,      0, 9*   0,    242, 9* 242,    393, 4* 393,    569,MM  3919
2    4* 569,    738, 4* 738,    915, 2* 915,   1019,   1048,   1087,MM  3920
3      1110,   1135,   1169,   1198,   1240,   1271,   1301,   1337,MM  3921
4      1369,   1401,   1423,   1443,   1458,   1470,   1505, 4*1505,MM  3922
5      1595, 9*1595,   1643, 4*1643,      0, 9*   0,     91, 3*  91,MM  3923
6        95, 5*  95,    141, 4* 141,      0, 9*   0,    242, 9* 242,MM  3924
7       393, 4* 393,    569, 4* 569,    738, 4* 738,    916, 1* 916,MM  3925
8       977,   1020,   1049,   1088,   1111,   1136,   1170,   1199,MM  3926
9      1241,   1272,   1302,   1338,   1370,   1402,   1424,   1444/MM  3927
 END                                                              MM  3928
```

```
      BLOCK DATA                                                        MM  3929
C                                                                       MM  3930
C         BLOCK DATA PROGRAM NUMBER FOUR                                MM  3931
C         VARIABLE SIZE GRID ID NUMBERS  FOR ST. LOUIS                  MM  3932
      INTEGER A31, A32, A33                                             MM  3933
      DIMENSION          A31(393), A32(524), A33(1205)                  MM  3934
      COMMON /VGRID/  ITGRID,   IVG(100,100)                            MM  3935
      EQUIVALENCE (A31(1),IVG(79,79))                                   MM  3936
      EQUIVALENCE (A32(1),IVG(72,83))                                   MM  3937
      EQUIVALENCE (A33(1),IVG(96,88))                                   MM  3938
C                                                                       MM  3939
      DATA A31/                                                         MM  3940
     1   1*1444,   1471,   1505, 4*1505,   1595, 9*1595,   1643, 4*1643,MM  3941
     2         0, 9*   0,     91, 3*  91,     95, 5*  95,    141, 4* 141,MM  3942
     3         0, 9*   0,    242, 9* 242,    393, 4* 393,    569, 4* 569,MM  3943
     4       738, 4* 738,    916, 1* 916,    978,   1021,   1050,   1089,MM  3944
     5      1112,   1137,   1171,   1200,   1242,   1273,   1303,   1339,MM  3945
     6      1371,   1403,   1425,   1444, 1*1444,   1472,   1505, 4*1505,MM  3946
     7      1595, 9*1595,   1643, 4*1643,       0,34*   0,    242, 9* 242,MM  3947
     8       394, 4* 394,   2234, 4*2234,    739, 4* 739,    917, 1* 917,MM  3948
     9       979,   1022,   1051,   1090,   1113,   1138,   1172,   1201,MM  3949
     1      2306, 1*2306,   2311,   2315,   2321,   1404,   1426,   1445,MM  3950
     2   1*1445,   1473,   1506, 4*1506,   1595, 9*1595,   1643, 4*1643,MM  3951
     3         0,34*   0,    242, 9* 242,    394, 4* 394,   2234, 4*2234,MM  3952
     4       739, 4* 739,    917, 1* 917,    980,   1023,   1052,   1091,MM  3953
     5      1114,   1139,   1173,   1202,   2306, 1*2306,   2312,   2316,MM  3954
     6   1*2316,   1405,   1427,   1445, 1*1445,   1474,   1506, 4*1506,MM  3955
     7      1595, 9*1595,   1643, 4*1643,       0,34*   0,    242, 9* 242,MM  3956
     8       394, 4* 394,   2234, 4*2234,    739, 4* 739,    918, 1* 918,MM  3957
     9       981,   1024,   1053,   1092, 1*1092,   1140, 2*1140,   2307/MM  3958
      DATA A32/                                                         MM  3959
     1   2*2307,   2316, 1*2316,   1406,   1428,   1446, 2*1446,   1506,MM  3960
     2   4*1506,   1595, 9*1595,   1643, 4*1643,       0,34*   0,    242,MM  3961
     3   9* 242,    394, 4* 394,   2234, 4*2234,    739, 4* 739,    918,MM  3962
     4   1* 918,    982,   1025,   1054,   1092, 1*1092,   1140, 2*1140,MM  3963
     5     2307, 2*2307,   2317, 1*2317,   1407,   1429,   1446, 2*1446,MM  3964
     6     1506, 4*1506,   1595, 9*1595,   1643, 4*1643,       0,34*   0,MM  3965
     7      242, 9* 242,    394, 4* 394,   2234, 4*2234,    739, 4* 739,MM  3966
     8     2267, 1*2267,    983,   2281, 1*2281,   1093,   1115,   1140,MM  3967
     9   2*1140,   2307, 2*2307,   2317, 1*2317,   1408,   1430,   1446,MM  3968
     1   2*1446,   1506, 4*1506,   1595, 9*1595,   1643, 4*1643,       0,MM  3969
     2  54*   0,    740, 4* 740,   2267, 1*2267,   2277,   2281, 1*2281,MM  3970
     3     1094, 4*1094,   1245, 4*1245,   1409, 4*1409,   1507, 4*1507,MM  3971
     4     1596, 4*1596,   1628, 4*1628,   1644, 4*1644,       0,54*   0,MM  3972
     5      740, 4* 740,   2268, 2*2268,   2282, 1*2282,   1094, 4*1094,MM  3973
     6     1245, 4*1245,   1409, 4*1409,   1507, 4*1507,   1596, 4*1596,MM  3974
     7     1628, 4*1628,   1644, 4*1644,       0,54*   0,    740, 4* 740,MM  3975
     8     2268, 2*2268,   2282, 1*2282,   1094, 4*1094,   1245, 4*1245,MM  3976
     9     1409, 4*1409,   1507, 4*1507,   1596, 4*1596,   1628, 4*1628/MM  3977
      DATA A33/                                                         MM  3978
     1     1644, 4*1644,       0,54*   0,    740, 4* 740,   2268, 2*2268,MM  3979
     2     2283, 1*2283,   1094, 4*1094,   1245, 4*1245,   1409, 4*1409,MM  3980
     3     1507, 4*1507,   1596, 4*1596,   1628, 4*1628,   1644, 4*1644,MM  3981
     4         0,54*   0,    740, 4* 740,   2269,   2272,   2278,   2283,MM  3982
     5   1*2283,   1094, 4*1094,   1245, 4*1245,   1409, 4*1409,   1507,MM  3983

     6   4*1507,   1596, 4*1596,   1628, 4*1628,   1644, 4*1644,  100*0,MM  3984
     7   100*0, 100*0, 100*0, 100*0, 100*0, 100*0, 100*0, 100*0, 100*0/ MM  3985
      END                                                               MM  3986
```

2. Emission Module Listing

```
      PROGRAM EMMOD                                                          EM     1
C                                                                            EM     2
C       EMMOD IS THE DRIVER FOR THE   ERT  EMISSIONS MODULE.                 EM     3
C                                                                            EM     4
C       THIS IS A SPECIAL VERSION OF EMMOD SUITABLE ONLY FOR                 EM     5
C       CREATING EMISSION FLUX HISTORIES FROM THE RAPS DATA BASE.            EM     6
C                                                                            EM     7
C       SUBROUTINES REQUIRED                                                 EM     8
C             ADHOUR      AREAEM      GRIDIT    PONTEM    LOCATE           EM     9
C             PLUMAS      PARTIT      GESTAB    SEGMET    STACKS           EM    10
C             PREDAT      NEWPAG      XMIT      MDATE     BLOCKDATA        EM    11
C             SECOND      DHPLUM                                           EM    12
C                                                                            EM    13
C       FUNCTIONS REQUIRED                                                   EM    14
C             ERF         TRPLOG      DISTAN                               EM    15
C                                                                            EM    16
C       PERIPHERALS REQUIRED                                                 EM    17
C             TAPE1 = PUNCHED OUTPUT     (UNIT LPUNCH)                     EM    18
C             TAPE5 = INPUT                                                EM    19
C             TAPE3 = INTERMEDIATE I/O     (UNIT LIN)                      EM    20
C             TAPE6 = OUTPUT      (UNIT LOUT)                              EM    21
C             LUGRID = LOGICAL UNIT OF GRID FILE                           EM    22
C             LUAREA = LOGICAL UNIT OF AREA  SOURCE EMISSIONS FILE         EM    23
C             LUPONT = LOGICAL UNIT OF POINT SOURCE EMISSIONS FILE         EM    24
      COMMON /HISTRY/  TIMEHS(520),       IDHS(520),       UTMHS(2,520),  EM    25
     1                 NUMHS                                               EM    26
      COMMON /TABLES/  THOUR(24)                                           EM    27
      COMMON /INPUTS/  TITLE(20),         JDATE(10),       NCASE          EM    28
      COMMON /LABLIN/  NAMIN(11),         NFLXIN,          WTMOLI(11),    EM    29
     1                 ADJUST(11)                                          EM    30
      COMMON /LABOUT/  NAMOUT(7),         NFXOUT,          WTMOLO(7)      EM    31
      COMMON /ANSWER/  YES,               RNEG,            IYES,          EM    32
     1                 NO,                SMALL                           EM    33
      COMMON /WIND/    T(100),            V(100),          TH(100),       EM    34
     1                 NPTS                                                EM    35
      COMMON /TRAJ/    P(2,100)                                            EM    36
      COMMON /PARCEL/  PWIDTH,            HPWIDT,          PLENTH,        EM    37
     1                 PAREA,             DTFREZ,          SIGEOG,        EM    38
     2                 ZEE(5),            HTINTF(6),       NOSTAT,        EM    39
     3                 ROAIR                                               EM    40
      COMMON /TEMPS/   TMTEMP(25),        TEMPSF(25),      NTEMP,         EM    41
     1                 ITEMP                                               EM    42
      COMMON /MXHITE/  TMXHIT(25),        HTMIXL(25),      NMIXL,         EM    43
     1                 IMIXL                                               EM    44
      COMMON /ORIGIN/  UTMXOR,            UTMYOR                          EM    45
      COMMON /PASQAL/  TMSTAB(25),        KSTABL(25),      ISTBMX         EM    46
      COMMON /EMRATE/  EMARAT(7,520),     EMABAS(7),       NOSUMS         EM    47
C                                                                            EM    48
      DATA YES, RNEG, IYES, NO /4HYES ,4HNO  ,4HYES ,4HNO   /              EM    49
      DATA PMOR, TEND   /4HMORE, 4HEND  /                                  EM    50
      DATA NCASE, KOK /1,-1/                                               EM    51
      DATA THOUR /0.0,60.0,120.0,180.0,240.0,300.0,360.0,420.0,480.0,      EM    52
     1 540.0,600.0,660.0,720.0,780.0,840.0,900.0,960.0,1020.0,1080.0,      EM    53
     1 1140.0,1200.0,1260.0,1320.0,1380.0/                                 EM    54
      DATA  SMALL,  ROAIR,  RAD  /1.E-4,  1178.,  .01745322925 /           EM    55
```

```
      DATA SIGEDG, DTFREZ /2.15 , 60./                                  EM     56
      DATA LIN, LOUT /3,6/                                              EM     57
      TERM = TEND                                                       EM     58
      DEBUG =YES                                                        EM     59
      CALL MDATE(JDATE)                                                 EM     60
C                                                                       EM     61
C        READ GENERAL INPUTS                                            EM     62
C                                                                       EM     63
100   CONTINUE                                                          EM     64
      CALL SECOND(A1)                                                   EM     65
      CALL PREDAT                                                       EM     66
      READ(LIN,1)  TITLE                                                EM     67
      READ(LIN,2)  NDAREA                                               EM     68
      READ(LIN,2)  NOPONT                                               EM     69
      READ(LIN,2)  NOSUMS                                               EM     70
      READ(LIN,2)  IPUNCH                                               EM     71
      READ(LIN,3)  LUGRID                                               EM     72
      READ(LIN,3)  LUAREA                                               EM     73
      READ(LIN,3)  LUPONT                                               EM     74
      READ(LIN,3)  NFLXIN                                               EM     75
      READ(LIN,3)  NFXOUT                                               EM     76
      READ(LIN,11)  (NAMIN(I),ADJUST(I),I=1,NFLXIN)                     EM     77
      READ(LIN,2)  (NAMOUT(I),I=1,NFXOUT)                               EM     78
      READ(LIN,23)  PLENTH                                              EM     79
      READ(LIN,23)  PWIDTH                                              EM     80
      PAREA = PLENTH*PWIDTH                                             EM     81
      HPWIDT = .50*PWIDTH                                               EM     82
      IF(NDAREA.NE.IYES)  GO TO 180                                     EM     83
C                                                                       EM     84
C          AREA SOURCE INPUTS AND DETERMINATION                         EM     85
C                                                                       EM     86
      READ(LIN,13)  (EMABAS(I),I=1,NFXOUT)                              EM     87
      DO 120 K = 1,501                                                  EM     88
      READ(LIN,4) A, B, C, ID                                           EM     89
      IF(A.LT.0.)  GO TO 140                                            EM     90
      TIMEHS(K) = A                                                     EM     91
      UTMHS(1,K) = B                                                    EM     92
      UTMHS(2,K) = C                                                    EM     93
      IDHS(K) = ID                                                      EM     94
120   CONTINUE                                                          EM     95
      WRITE(LOUT,5)                                                     EM     96
      GO TO 500                                                         EM     97
140   NUMHS = K-1                                                       EM     98
C                                                                       EM     99
      CALL ACHOUR                                                       EM    100
C                                                                       EM    101
      CALL NEWPAG(TITLE,0,JDATE)                                        EM    102
      WRITE(LOUT,14)                                                    EM    103
      DO 160 K = 1,NUMHS                                                EM    104
      IF(MOD(K,25).NE.0) GO TO 150                                      EM    105
      CALL NEWPAG(TITLE,0,JDATE)                                        EM    106
      WRITE(LOUT,14)                                                    EM    107
150   WRITE(LOUT,6) TIMEHS(K),UTMHS(1,K),UTMHS(2,K),IDHS(K)             EM    108
160   CONTINUE                                                          EM    109
      IF(NCASE.GT.1)  GO TO 170                                         EM    110
```

```
C                                                                       EM   111
      CALL GRIDIT(LUGRID,KOK)                                           EM   112
C                                                                       EM   113
      IF(KOK.LE.0) GO TO 170                                            EM   114
      WRITE(LOUT,12)  KOK                                               EM   115
      GO TO 500                                                         EM   116
170   CONTINUE                                                          EM   117
      IF(NOAREA.NE.IYES) GO TO 180                                      EM   118
C                                                                       EM   119
      CALL AREAEM(LUAREA,IPUNCH,KOK)                                    EM   120
C                                                                       EM   121
      IF(KOK.LE.0)  GO TO 180                                           EM   122
      WRITE(LOUT,8)  KOK                                                EM   123
      GO TO 500                                                         EM   124
180   CONTINUE                                                          EM   125
      IF(NOPONT.NE.IYES)  GO TO 400                                     EM   126
C                                                                       EM   127
C          POINT SOURCE INPUTS AND DETERMINATION                        EM   128
C                                                                       EM   129
      READ(LIN,20)  UTMXOR, UTMYOR                                      EM   130
      READ(LIN,20)  P(1,1),  P(2,1)                                     EM   131
      DO 200 I = 1,101                                                  EM   132
      READ(LIN,20)  A, B, C                                             EM   133
      IF(A .LT. 0.0)  GO TO 210                                         EM   134
      T(I) = A                                                          EM   135
      V(I) = B/60.                                                      EM   136
      TH(I) = C*RAD                                                     EM   137
200   CONTINUE                                                          EM   138
      WRITE(LOUT,21)                                                    EM   139
      GO TO 500                                                         EM   140
210   NPTS = I - 1                                                      EM   141
      DO 220 I = 2,NPTS                                                 EM   142
      J = I - 1                                                         EM   143
      DT = T(I) - T(J)                                                  EM   144
      DX = V(J)*COS(TH(J))*DT                                           EM   145
      DY = V(J)*SIN(TH(J))*DT                                           EM   146
      P(1,I) = P(1,J) + DX                                              EM   147
      P(2,I) = P(2,J) + DY                                              EM   148
220   CONTINUE                                                          EM   149
      CALL NEWPAG(TITLE,0,JDATE)                                        EM   150
      WRITE(LOUT,27)                                                    EM   151
      DO 230 I = 1,NPTS                                                 EM   152
      IF(MOD(I,25).NE.0)  GO TO 225                                     EM   153
      CALL NEWPAG(TITLE,0,JDATE)                                        EM   154
      WRITE(LOUT,27)                                                    EM   155
225   VPHR =V(I)*60.                                                    EM   156
      THDEG = TH(I)/RAD                                                 EM   157
      WRITE(LOUT,28) I, T(I), P(1,I), P(2,I), VPHR, THDEG               EM   158
230   CONTINUE                                                          EM   159
C                                                                       EM   160
      READ(LIN,3)  NOSTAT                                               EM   161
      IF(NOSTAT .LE. 5)  GO TO 240                                      EM   162
      WRITE(LOUT,22)                                                    EM   163
      GO TO 500                                                         EM   164
240   READ(LIN,23)  (ZEE(I),I=1,NOSTAT)                                 EM   165
```

```
      HTINTF(1) = ZEE(1)                                                EM  166
      DO 250 J = 2,NOSTAT                                               EM  167
      I = J - 1                                                         EM  168
      HTINTF(J) = 0.50*(ZEE(I) + ZEE(J))                                EM  169
250   CONTINUE                                                          EM  170
      HTINTF(NOSTAT+1) = ZEE(NOSTAT)                                    EM  171
C                                                                       EM  172
C     READ(LIN,23)  DTFREZ                                              EM  173
C     READ(LIN,23)  SIGEDG                                              EM  174
      DO 260 I = 1,26                                                   EM  175
      READ(LIN,20)  A,B                                                 EM  176
      IF(A .LT. -9.)  GO TO 270                                         EM  177
      TMTEMP(I) = A                                                     EM  178
      TEMPSF(I) = B                                                     EM  179
260   CONTINUE                                                          EM  180
      WRITE(LOUT,24)                                                    EM  181
      GO TO 500                                                         EM  182
270   NTEMP = I - 1                                                     EM  183
      ITEMP = 1                                                         EM  184
C                                                                       EM  185
      K=1                                                               EM  186
      DO 280 I = 1,26                                                   EM  187
      READ(LIN,25) A, IS                                                EM  188
      IF(A .LT. 0.)  GO TO 300                                          EM  189
      IF(I.EQ.1) GO TO 275                                              EM  190
      IF(IS.EQ.KSTABL(K-1))  GO TO 280                                  EM  191
275   CONTINUE                                                          EM  192
      TMSTAB(K) = A                                                     EM  193
      KSTABL(K) = IS                                                    EM  194
            K  = K+1                                                    EM  195
280   CONTINUE                                                          EM  196
      WRITE(LOUT,26)                                                    EM  197
      GO TO 500                                                         EM  198
300   CONTINUE                                                          EM  199
      ISTBMX = K-1                                                      EM  200
      DO 310 I = 1,26                                                   EM  201
      READ(LIN,20) A, B                                                 EM  202
      IF(A.LT.0.0)  GO TO 320                                           EM  203
      TMXHIT(I) = A                                                     EM  204
      HTMIXL(I) = B                                                     EM  205
310   CONTINUE                                                          EM  206
      WRITE(LOUT,32)                                                    EM  207
      GO TO 500                                                         EM  208
320   NMIXL = I - 1                                                     EM  209
      IMIXL = 1                                                         EM  210
C                                                                       EM  211
      CALL NEWPAG(TITLE,0,JDATE)                                        EM  212
      WRITE(LOUT,29) (TMSTAB(I), KSTABL(I), I=1,ISTBMX)                 EM  213
      WRITE(LOUT,30) (TMTEMP(I), TEMPSF(I), I=1,NTEMP)                  EM  214
      CALL NEWPAG(TITLE,0,JDATE)                                        EM  215
      WRITE(LOUT,31) (TMXHIT(I),HTMIXL(I),I=1,NMIXL)                    EM  216
C                                                                       EM  217
      CALL PONTEM(LUPONT,IPUNCH,KOK,NDSUMS)                             EM  218
C                                                                       EM  219
      IF(KOK.GT.0)  GO TO 400                                           EM  220
```

```
      WRITE(LOUT,9)  KOK                                                    EM   221
      GO TO 500                                                             EM   222
400   CONTINUE                                                              EM   223
C                                                                           EM   224
      CALL NEWPAG(TITLE,0,JDATE)                                            EM   225
      CALL SECOND(A2)                                                       EM   226
      A2 = A2 - A1                                                          EM   227
      WRITE(LOUT,10) NCASE, A2                                              EM   228
410   READ(LIN,1) TERM                                                      EM   229
      IF(TERM.EQ.RMOR)  GO TO 420                                           EM   230
      IF(TERM.NE.TEND)  GO TO 410                                           EM   231
      GO TO 500                                                             EM   232
420   CONTINUE                                                              EM   233
      NCASE = NCASE + 1                                                     EM   234
      GO TO 100                                                             EM   235
500   STOP                                                                  EM   236
C                                                                           EM   237
1     FORMAT (20A4)                                                         EM   238
2     FORMAT (40X,A4)                                                       EM   239
3     FORMAT (40X,I2)                                                       EM   240
4     FORMAT (10X,3F10.2,I10)                                               EM   241
5     FORMAT (54H1TOO MANY GRID SQUARE CROSSINGS INPUT -- JOB ABORTED  )EM   242
6     FORMAT (1H0,6X,3F12.2,I12)                                            EM   243
7     FORMAT (40X,A4,6X,2F10.0)                                             EM   244
8     FORMAT (50H1CALCULATIONS STOPPED IN 'AREAEM'  --  JOB ABORTED ,I9)EM   245
9     FORMAT (50H1CALCULATIONS STOPPED IN 'PONTEM'  --  JOB ABORTED ,I9)EM   246
10    FORMAT (20H0END OF CASE NUMBER   ,I3,10X,10HCP TIME =  ,F8.2,       EM   247
     1  5H  SEC )                                                         EM   248
11    FORMAT (40X,A4,6X,F10.0)                                              EM   249
12    FORMAT (50H1CALCULATIONS STOPPED IN 'GRIDIT'  --  JOB ABORTED ,I9)EM   250
13    FORMAT (40X,F10.0)                                                    EM   251
14    FORMAT (1H0,31HTRAJECTORY DATA BY GRID SQUARE    ,                    EM   252
     1  //   ,12X,4HTIME,8X,6HUTM-X ,6X,6HUTM-Y  ,10X,2HID )               EM   253
20    FORMAT (40X,3F10.0)                                                   EM   254
21    FORMAT (1H0,48HTOO MANY TRAJECTORY NODES INPUT --  JOB ABORTED  ) EM   255
22    FORMAT (1H0,48HTOO MANY VERTICAL STATIONS INPUT - JOB ABORTED   ) EM   256
23    FORMAT (40X,F10.0)                                                    EM   257
24    FORMAT (1H0,48HTOO MANY TEMPERATURES INPUT  --  JOB ABORTED     ) EM   258
25    FORMAT (40X,F10.0,I10)                                                EM   259
26    FORMAT (1H0,48HTOO MANY STABILITY UPDATES INPUT - JOB ABORTED   ) EM   260
27    FORMAT(1H08X16HTRAJECTORY DATA  ,//,4X,3HLEG,5X,4HTIME,8X,1HX,10X,EM   261
     11HY,10X,8HVELOCITY,6X,5HTHETA,/,4X,3HNO.,5X,5H(MIN),6X,4H(KM),     EM   262
     2  7X,4H(KM),8X,8H(KM/HR) ,4X,  9H(DEGREES)  )                       EM   263
28    FORMAT (1H0,I5,F11.1,2F11.2,2F13.3)                                   EM   264
29    FORMAT(1H0,8X,20HSTABILITY CLASS DATA  ,//,11X,4HTIME,5X,5HCLASS, EM   265
     1  25(//,10X,F5.0,7X,I1))                                            EM   266
30    FORMAT(1H0,8X,26HSURFACE TEMPERATURE DATA    ,//,11X,4HTIME,7X,     EM   267
     1  11HTEMPERATURE ,25(//,10X,F5.0,7X,F7.1))                          EM   268
31    FORMAT(1H0,8X,25HMIXING LAYER DEPTH DATA    ,//,11X,4HTIME,6X,      EM   269
     1  6HDEPTH ,25(//,10X,F5.0,5X,F6.1))                                 EM   270
32    FORMAT(51H0TOO MANY MIXING LAYER HEIGHTS INPUT -- JOB ABORTED )       EM   271
      END                                                                   EM   272
```

```
      SUBROUTINE ADHOUR                                                 EM   273
C                                                                       EM   274
C          *ADHOUR*  INSERTS TRAJECTORY TIMES ON THE EVEN HOURS         EM   275
C                                                                       EM   276
      DIMENSION  IWORK(520)                                             EM   277
      COMMON /HISTRY/ TIMEHS(520),     IDHS(520),       UTMHS(2,520),   EM   278
     1                NUMHS                                            EM   279
      COMMON /TABLES/ THOUR(24)                                         EM   280
      COMMON /WORKER/ IDATE,           NG, IHR,         WORK(22440)     EM   281
      EQUIVALENCE (IWORK,WORK)                                          EM   282
C                                                                       EM   283
      K = 1                                                             EM   284
      JT = IFIX(TIMEHS(1)+SMALL)/60 + 2                                 EM   285
      DO 100 I = 1,NUMHS                                                EM   286
      IM1 = I - 1                                                       EM   287
60    IF(TIMEHS(I).LE.THOUR(JT)) GO TO 90                               EM   288
      IWORK(K) = IDHS(IM1)                                              EM   289
      WORK(K+520) = THOUR(JT)                                           EM   290
      TPORT = (THOUR(JT)-TIMEHS(IM1))/(TIMEHS(I)-TIMEHS(IM1))           EM   291
      WORK(K+1040) = UTMHS(1,IM1)+TPORT*(UTMHS(1,I)-UTMHS(1,IM1))       EM   292
      WORK(K+1560) = UTMHS(2,IM1)+TPORT*(UTMHS(2,I)-UTMHS(2,IM1))       EM   293
      K = K + 1                                                         EM   294
      JT = JT + 1                                                       EM   295
      GO TO 60                                                          EM   296
90    CONTINUE                                                          EM   297
      IWORK(K) = IDHS(I)                                                EM   298
      WORK(K+520) = TIMEHS(I)                                           EM   299
      WORK(K+1040) = UTMHS(1,I)                                         EM   300
      WORK(K+1560) = UTMHS(2,I)                                         EM   301
      K = K + 1                                                         EM   302
100   CONTINUE                                                          EM   303
      NUMHS = K - 1                                                     EM   304
      DO 120 I = 1,NUMHS                                                EM   305
      IDHS(I) = IWORK(I)                                                EM   306
      TIMEHS(I) = WORK(I+520)                                           EM   307
      UTMHS(1,I) = WORK(I+1040)                                         EM   308
      UTMHS(2,I) = WORK(I+1560)                                         EM   309
120   CONTINUE                                                          EM   310
      K = NUMHS + 1560                                                  EM   311
      CALL XMIT (-K,0.,WORK)                                            EM   312
      RETURN                                                            EM   313
      END                                                               EM   314



      SUBROUTINE AREAEM(LUAREA,IPUNCH,KOK)                              EM   315
C                                                                       EM   316
C        AREAEM READS THE RAPS AREA SOURCE EMISSIONS TAPE AND           EM   317
C        DETERMINES THE AREA SOURCE EMISSION RATE HISTORY FOR           EM   318
C        A SPECIFIED TRAJECTORY.                                        EM   319
C                                                                       EM   320
C       RAP FILE DESCRIPTION                                            EM   321
C                                                                       EM   322
C          IDATE NGRID IHOUR  PART SOX  NOX  THC  CO   DUST             EM   323
C                             NRHC PARF OLEF AROM ALDE                  EM   324
```

```
C                                PART SOX  .....                          EM
C                                                .....ALDE                 EM
C                                (REPEATED NGRID TIMES PER HOUR)           EM
C                                                                          EM
      DIMENSION        ASOURC(11,2000),  IDLAST(44),       SUM(510)       EM
     1               , ISUM(3)                                             EM
      COMMON /DEGRID/ ID(2000),          AREA(2000),      NGRID           EM
      COMMON /INPUTS/ TITLE(20),         JDATE(10),        NCASE          EM
      COMMON /HISTRY/ TIMEHS(520),       IDHS(520),        UTMHS(2,520),  EM
     1                NUMHS                                               EM
      COMMON /TABLES/ THOUR(24)                                            EM
      COMMON /PARCEL/ PWIDTH,            HPWIDT,          PLENTH,         EM
     1                PAREA,             DTFREZ,          SIGEDG,         EM
     2                ZEE(5),            HTINTF(6),       NOSTAT,         EM
     3                ROAIR                                               EM
      COMMON /WORKER/ IDATE,             NG,IHOUR,         WORK(22440)    EM
      COMMON /EMRATE/ EMARAT(7,520),     EMABAS(7),        NOSUMS         EM
      COMMON /ANSWER/ YES,RNEG,          IYES,NO,          SMALL          EM
      COMMON /LABLIN/ NAMIN(11),         NFLXIN,          WTMOLI(11),     EM
     1                ADJUST(11)                                          EM
      COMMON /LABOUT/ NAMOUT(7),         NFXOUT,          WTMOLO(7)       EM
      EQUIVALENCE   (ASOURC, WORK )                                        EM
      EQUIVALENCE   (ISUM, SUM)                                            EM
      DATA ALL, LOUT, LPUNCH /4HALL ,6,1/                                  EM
      DATA NUMREC, NWPREC /44,510/                                         EM
      DEBUG = RNEG                                                         EM
      PARERR = RNEG                                                        EM
      NM1 = NUMREC - 1                                                     EM
      IF(NCASE.GT.1)   GO TO 80                                            EM
C        GENERATE IDLAST ARRAY WHICH CONTAINS THE GRID SQUARE ID NUMBER EM
C        ASSOCIATED WITH THE LAST SET OF NFLXIN VARIABLE IN EACH RECORD.EM
      DO 70 J = 1,NUMREC                                                   EM
      JL = (NWPREC*J - 3)/NFLXIN                                           EM
      IF(JL.LE.NGRID)  GO TO 65                                            EM
      IDLAST(J) = ID(NGRID) + 1                                            EM
      GO TO 70                                                             EM
65    CONTINUE                                                             EM
      IDLAST(J) = ID(JL)                                                   EM
70    CONTINUE                                                             EM
      IF(DEBUG.EQ.YES)  WRITE(LOUT,9) (IDLAST(J),J=1,NUMREC)               EM
80    CONTINUE                                                             EM
C                                                                          EM
C        SKIP 'PRE-TRAJECTORY START TIME DATA' IN FILE                     EM
C                                                                          EM
      INDEX = IFIX(TIMEHS(1)+SMALL)/60                                     EM
      IF(INDEX.EQ.0)  GO TO 125                                            EM
      IF(INDEX.GT.23) STOP                                                 EM
      DO 120 I = 1,INDEX                                                   EM
CUNIVAC  READ(LUAREA,ERR=100,END=510) (ISUM(JJ),JJ=1,3)                    EM
      BUFFER IN (LUAREA,1) (ISUM(1),ISUM(3))                               EM
90    IF(UNIT(LUAREA))   95,  510,  100                                    EM
95    IF(DEBUG.EQ.YES)  WRITE(LOUT,1) (ISUM(JJ),JJ=1,3)                    EM
100   DO 110 J = 2,NUMREC                                                  EM
      READ(LUAREA)                                                         EM
110   CONTINUE                                                             EM
```

```
120   CONTINUE                                                          EM   380
C                                                                       EM   381
125   INEXT = 1                                                         EM   382
      K2MAX = NFLXIN*2000                                               EM   383
      IF(NOSUMS.NE.IYES)  GO TO 130                                     EM   384
      CALL NEWPAG(TITLE,0,JDATE)                                        EM   385
      WRITE(LOUT,6)  (NAMIN(JJ),JJ=1,NFLXIN)                            EM   386
C                                                                       EM   387
C         HOURLY TIME CYCLE BEGINS HERE                                 EM   388
C                                                                       EM   389
130   IHIST = INEXT                                                     EM   390
      DONTRD = RNEG                                                     EM   391
      INDEX = INDEX + 1                                                 EM   392
      IDMAX = -1                                                        EM   393
      DO 150 I = IHIST,NUMHS                                            EM   394
      JT = IFIX(TIMEHS(I)+SMALL)/60 + 1                                 EM   395
      IF(JT.LE.INDEX)  GO TO 140                                        EM   396
      INEXT = I                                                         EM   397
      GO TO 160                                                         EM   398
140   IDMAX = MAX0(IDMAX,IDHS(I))                                       EM   399
150   CONTINUE                                                          EM   400
      INEXT = I + 1                                                     EM   401
160   IF(IDMAX.GT.0)   GO TO 200                                        EM   402
C                                                                       EM   403
C        ALL TRAJECTORY NODES THIS HOUR ARE OFF GRID AND                EM   404
C        THE DEFAULT AREA SOURCE EMISSION RATES ARE USED                EM   405
C                                                                       EM   406
      DONTRD = YES                                                      EM   407
      READ(LUAREA) IDATE , NG , IHOUR                                   EM   408
      IF(IOCHEC(LUAREA))  185,180                                       EM   409
180   IF(DEBUG.EQ.YES) WRITE(LOUT,1) IDATE,NG,IHOUR                     EM   410
185   CONTINUE                                                          EM   411
      DO 190 J = 2,NUMREC                                               EM   412
      READ(LUAREA)                                                      EM   413
      IF(IOCHEC(LUAREA))  190,190 .                                     EM   414
190   CONTINUE                                                          EM   415
      GO TO 300                                                         EM   416
200   CONTINUE                                                          EM   417
C                                                                       EM   418
C        FIND LAST RECORD TO BE READ FOR THIS HOUR                      EM   419
C                                                                       EM   420
      DO 220 J = 1,NUMREC                                               EM   421
      IF(IDMAX.GE.IDLAST(J))  GO TO 220                                 EM   422
      LSTREC = J                                                        EM   423
      GO TO 230                                                         EM   424
220   CONTINUE                                                          EM   425
      LSTREC = J                                                        EM   426
230   CONTINUE                                                          EM   427
      IF(NOSUMS.EQ.IYES)  LSTREC = NUMREC                               EM   428
      IF(DEBUG.EQ.YES)  WRITE(LOUT,2)  LSTREC                           EM   429
      K2 = NWPREC *3                                                    EM   430
C                                                                       EM   431
C       ON THE UNIVAC REPLACE BUFFER IN AND IF(UNIT) WITH               EM   432
C     READ(LUAREA,ERR=245,END=510) IDATE,NG,IHOUR,(SUM(JJ),JJ=1,K2)     EM   433
C                                                                       EM   434
```

```
      BUFFER IN (LUAREA,1)  (SUM(1),SUM(NWPREC))                        EM   435
240   IF(UNIT(LUAREA))   242,  510,  245                                EM   436
242   IDATE = ISUM(1)                                                   EM   437
      NG = ISUM(2)                                                      EM   438
      IHOUR = ISUM(3)                                                   EM   439
      CALL XMIT(K2,SUM(4),WORK)                                         EM   440
      GO TO 250                                                         EM   441
245   PARERR = YES                                                      EM   442
      WRITE(LOUT,7)                                                     EM   443
250   CONTINUE                                                          EM   444
      IF(LSTREC.EQ.1)  GO TO 275                                        EM   445
      DO 270 J = 2,NUMREC                                               EM   446
      K1 = K2 + 1                                                       EM   447
      K2 = K2 + NWPREC                                                  EM   448
      IF(J.GT.LSTREC)    GO TO 265                                      EM   449
C                                                                       EM   450
C        ON THE UNIVAC REPLACE BUFFER IN AND IF(UNIT) WITH              EM   451
C     READ(LUAREA,ERR=263,END=510) (SUM(JJ),JJ=1,NWPREC)                EM   452
C                                                                       EM   453
      BUFFER IN (LUAREA,1) (SUM(1), SUM(NWPREC))                        EM   454
260   IF(UNIT(LUAREA))   262,  510,  263                                EM   455
262   CALL XMIT(NWPREC,SUM,WORK(K1))                                    EM   456
      GO TO 270                                                         EM   457
263   PARERR = YES                                                      EM   458
      WRITE(LOUT,7)                                                     EM   459
      GO TO 270                                                         EM   460
265   READ (LUAREA)                                                     EM   461
      IF(IOCHEC(LUAREA)) 270,270                                        EM   462
270   CONTINUE                                                          EM   463
275   CONTINUE                                                          EM   464
      IF(DEBUG.NE.ALL)  GO TO 280                                       EM   465
      WRITE(LOUT,1) IDATE, NG, IHOUR                                    EM   466
      N = LSTREC*NWPREC                                                 EM   467
      WRITE(LOUT,5) (WORK(K),K=1,N)                                     EM   468
280   CONTINUE                                                          EM   469
      IF(NDSUMS.NE.IYES)  GO TO 300                                     EM   470
      IF(PARERR.EQ.YES)  GO TO 300                                      EM   471
      DO 290 I = 1,NFLXIN                                               EM   472
      SUM(I) = 0.0                                                      EM   473
      DO 285 J = 1,NGRID                                                EM   474
      SUM(I) = SUM(I) + ASOURC(I,J)                                     EM   475
285   CONTINUE                                                          EM   476
      SUM(I) = SUM(I)*ADJUST(I)                                         EM   477
290   CONTINUE                                                          EM   478
      WRITE(LOUT,18) THOUR(INDEX), (SUM(I),I=1,NFLXIN)                  EM   479
C                                                                       EM   480
300   CONTINUE                                                          EM   481
C                                                                       EM   482
      INM1 = INEXT - 1                                                  EM   483
      DO 360 I = IHIST,INM1                                             EM   484
      IDSQ = IDHS(I)                                                    EM   485
      IF(IDSQ.LE.0)  GO TO 340                                          EM   486
      DO 310 J = 1,NGRID                                                EM   487
      IF(IDSQ.NE.IO(J))  GO TO 310                                      EM   488
      JDEX = J                                                          EM   489
```

```
      GO TO 320                                                          EM   490
310   CONTINUE                                                           EM   491
      GO TO 530                                                          EM   492
320   CONTINUE                                                           EM   493
      AREAFR = PAREA/(AREA(JDEX))                                        EM   494
C                                                                        EM   495
C     NOX, PARF, OLEF, AROM, ALDE, CO, SOX                               EM   496
C                                                                        EM   497
      EMARAT(1,I) = ASOURC( 3,JDEX)*ADJUST( 3)*AREAFR                    EM   498
      EMARAT(2,I) = ASOURC( 8,JDEX)*ADJUST( 8)*AREAFR                    EM   499
      EMARAT(3,I) = ASOURC( 9,JDEX)*ADJUST( 9)*AREAFR                    EM   500
      EMARAT(4,I) = ASOURC(10,JDEX)*ADJUST(10)*AREAFR                    EM   501
      EMARAT(5,I) = ASOURC(11,JDEX)*ADJUST(11)*AREAFR                    EM   502
      EMARAT(6,I) = ASOURC( 5,JDEX)*ADJUST( 5)*AREAFR                    EM   503
      EMARAT(7,I) = ASOURC( 2,JDEX)*ADJUST( 2)*AREAFR                    EM   504
C                                                                        EM   505
      IF(DEBUG.NE.YES)  GO TO 360                                        EM   506
      WRITE(LOUT,4) TIMEHS(I),IDSQ,(EMARAT(J,I),J=1,NFXOUT)              EM   507
      GO TO 360                                                          EM   508
340   CONTINUE                                                           EM   509
      DO 350 J = 1,NFXOUT                                                EM   510
      EMARAT(J,I) = EMABAS(J)                                            EM   511
350   CONTINUE                                                           EM   512
      IF(DEBUG.NE.YES)  GO TO 360                                        EM   513
      WRITE(LOUT,4) TIMEHS(I),IDSQ,(EMARAT(J,I),J=1,NFXOUT)              EM   514
360   CONTINUE                                                           EM   515
      IF(INM1 .LT. NUMHS)   GO TO 130                                    EM   516
C                                                                        EM   517
C           OUTPUT                                                       EM   518
C                                                                        EM   519
      CALL NEWPAG(TITLE,0,JDATE)                                         EM   520
      WRITE(LOUT,16) (NAMOUT(K),K=1,NFXOUT)                              EM   521
      DO 380 J =1,NUMHS                                                  EM   522
      IF(MOD(J,25).NE.0)  GO TO 370                                      EM   523
      CALL NEWPAG(TITLE,0,JDATE)                                         EM   524
      WRITE(LOUT,16) (NAMOUT(K),K=1,NFXOUT)                              EM   525
370   WRITE(LOUT,17) TIMEHS(J),IDHS(J),(EMARAT(I,J),I=1,NFXOUT)          EM   526
380   CONTINUE                                                           EM   527
      CALL XMIT(-NFXOUT,0.0,SUM)                                         EM   528
      TIMEHS(NUMHS+1) = TIMEHS(NUMHS) + SMALL                            EM   529
C       CONVERT FROM MOLES/HR TO MOLE FRACTION - METERS/MIN              EM   530
      A = 28.97/(ROAIR*PAREA*60.E6)                                      EM   531
      DO 400 I = 1,NUMHS                                                 EM   532
      DT = TIMEHS(I+1) - TIMEHS(I)                                       EM   533
      DT = DT/60.                                                        EM   534
      DO 400 J = 1,NFXOUT                                                EM   535
      SUM(J) = SUM(J) + EMARAT(J,I)*DT                                   EM   536
      EMARAT(J,I) = A*EMARAT(J,I)                                        EM   537
400   CONTINUE                                                           EM   538
      WRITE(LOUT,15) (SUM(K),K=1,NFXOUT)                                 EM   539
C                                                                        EM   540
      A = -10.                                                           EM   541
      CALL NEWPAG(TITLE,0,JDATE)                                         EM   542
      WRITE(LOUT,11) (NAMOUT(K),K=1,NFXOUT)                              EM   543
      IF(IPUNCH.EQ.IYES) WRITE(LPUNCH,12)                                EM   544
```

```
      DO 420 J = 1,NUMHS                                                EM   545
      IF(MOD(J,25).NE.0)  GO TO 410                                     EM   546
      CALL NEWPAG(TITLE,0,JDATE)                                        EM   547
      WRITE(LOUT,11) (NAMOUT(K),K=1,NFXOUT)                             EM   548
410   WRITE(LOUT,13) TIMEHS(J), (UTMHS(K,J),K=1,2), (EMARAT(K,J),       EM   549
     1                                              K =1,NFXOUT )      EM   550
      IF(IPUNCH.EQ.IYES)  WRITE(LPUNCH,14)                              EM   551
     1                     J,TIMEHS(J),(EMARAT(K,J),K=1,NFXOUT)       EM   552
420   CONTINUE                                                          EM   553
      IF(IPUNCH.EQ.IYES)   WRITE(LPUNCH,14) J,A                         EM   554
      RETURN                                                            EM   555
C                                                                       EM   556
510   KOK = 999                                                         EM   557
      WRITE(LOUT,8)  KOK                                                EM   558
      RETURN                                                            EM   559
520   KOK = 9999                                                        EM   560
      WRITE(LOUT,7)  KOK                                                EM   561
      RETURN                                                            EM   562
530   KOK = 99999                                                       EM   563
      WRITE(LOUT,10)  IDSQ, KOK                                         EM   564
      RETURN                                                            EM   565
C                                                                       EM   566
1     FORMAT (1H0,7HIDATE =,I8,10H   NGRID =  ,I7,10H   IHOUR =   ,I4)  EM   567
2     FORMAT (1H0,18H LAST RECORD NO.=   I4)                            EM   568
4     FORMAT (1H ,6HTIME =,F8.2,6H  ID = ,I6,3X,7E11.3,9H MOLES/HR )    EM   569
5     FORMAT (1H ,11E11.3)                                              EM   570
6     FORMAT (1H0,35HREGIONAL AREA SOURCE EMISSION SUMS ,30X,           EM   571
     1  12H(MOLES/HOUR),//,8H   TIME ,6X,11(A4,7X))                    EM   572
7     FORMAT (50H0 SUBROUTINE AREAEM ENCOUNTERED PARITY ERRORS  --      EM   573
     1 45HSOME DATA FROM PREVIOUS HOUR USED                  I4)       EM   574
8     FORMAT (80H1 SUBROUTINE AREAEM ENCOUNTERED UNEXPECTED EOF  --     EM   575
     1    JOB ABORTED                                        I8)       EM   576
9     FORMAT(1H0,8HIDLAST    )                                          EM   577
10    FORMAT (1H0,//,23H  GRID SQUARE ID NUMBER  ,I6,2X,                EM   578
     1  36HIS NOT ON THE GRID  --  JOB ABORTED   ,//, 6H KOK=  ,I8)     EM   579
11    FORMAT (1H0,27HAREA SOURCE EMISSION FLUXES ,30X,13HMOLE FRACTION  EM   580
     1 17H - METERS/MINUTE   ,//,6X,5HTIME ,5X,5HUTM-X,5X,5HUTM-Y,     EM   581
     2  7(5X,A4,3X))                                                   EM   582
12    FORMAT (30H   AREA SOURCE EMISSIONS                    )          EM   583
13    FORMAT (1H0,3F10.1,7E12.4)                                        EM   584
14    FORMAT (12HAREA SOURCES,I4,4X,F10.2,4E12.4,/,30X,4E12.4)          EM   585
15    FORMAT(1H ,20X,7(2X,10H----------),//,4X,12HTOTAL MOLES   ,       EM   586
     1   5X,7E12.4)                                                     EM   587
16    FORMAT (1H0,46HAREA SOURCE EMISSION RATES    --    MOLES/HR   ,   EM   588
     1  //,5X,5HTIME ,5X,2HID,4X,7(5X,A4,3X))                          EM   589
17    FORMAT (1H0,F10.2,3X,I4,3X,7E12.4)                                EM   590
18    FORMAT (1H0,F7.1,2X,11E11.3)                                      EM   591
      END                                                               EM   592
```

```
      SUBROUTINE DHPLUM(TA,TS,U,ISTAB,ACMM,HTINV,HTPS,DH,PENTFR,DTDZ)   EM   593
C                                                                       EM   594
C        DHPLUM CALCULATES PLUME RISE AND INVERSION PENETRATION.        EM   595
C                                                                       EM   596
C     INPUTS                                                            EM   597
C          TA     =  AMBIENT TEMPERATURE  (DEG C)                       EM   598
C          TS     =  STACK GAS TEMPERATURE  (DEG C)                     EM   599
C          U      =  WIND SPEED   (M/SEC)                               EM   600
C          ISTAB  =  ATMOSPHERIC STABILITY CLASS (1=6  FOR  A=F)        EM   601
C          ACMM   =  STACK VOLUMETRIC FLOW RATE   (M**3/MIN)            EM   602
C          HTINV  =  INVERSION HEIGHT (METERS)                          EM   603
C          HTPS   =  STACK HEIGHT  (METERS)                             EM   604
C          DTDZ   =  POTENTIAL TEMPERATURE GRADIENT  (DEG C/METER)      EM   605
C                                                                       EM   606
C     OUTPUTS                                                           EM   607
C          DH     =  PLUME RISE  ( METERS )                             EM   608
C          PENTFR =  INVERSION PENETRATION FRACTION  (0 TO 1)           EM   609
C                                                                       EM   610
      DATA ERLMT, NITRAT, YES /.005, 150, 4HYES /                       EM   611
      DATA DFDTDZ, GRAV, BETPSQ /.0137,  9.80665,  .160/                EM   612
      DATA UMIN /1./                                                    EM   613
      DEBUG = GRAV                                                      EM   614
      TAK = TA + 273.2                                                  EM   615
      TSK = TS + 273.2                                                  EM   616
      ACMS = ACMM/188.49556                                             EM   617
      F = GRAV*ACMS*(TSK-TAK)/TSK                                       EM   618
      UU = AMAX1(U,UMIN)                                                EM   619
      UDTDZ = DTDZ                                                      EM   620
      IF(DTDZ.LT.0.0  .OR.  DTDZ.GT.0.10)  UDTDZ = DFDTDZ               EM   621
      S = GRAV*UDTDZ/TAK                                                EM   622
      ZB = HTINV - HTPS                                                 EM   623
      ZBM = ZB - 5.                                                     EM   624
      PENTFR = 0.0                                                      EM   625
      IF(ZBM.LE.0.0)  PENTFR = 1.0000                                   EM   626
      IF(UU.GT.1.39   .OR. ISTAB.LE.4)  GO TO 10                        EM   627
C     -------  CALM CONDITIONS  --------                                EM   628
      DH = 5.0*(F/S**1.5)**.25                                          EM   629
      GO TO 100                                                         EM   630
10    IF(ISTAB.LE.4)  GO TO 20                                          EM   631
C     -------  STABLE WITH WIND  --------                               EM   632
      DH = 2.9*(F/(UU*S))**.333333                                      EM   633
      GO TO 100                                                         EM   634
20    CONTINUE                                                          EM   635
C     --------  NEUTRAL OR UNSTABLE WITH WIND, WITHOUT INVERSION -----  EM   636
C                 (IE....STACK HEIGHT ABOVE INVERSION HEIGHT)           EM   637
      IF(F.GT.55.) GO TO 30                                             EM   638
      XSTAR = 14.*F**.625                                               EM   639
      GO TO 40                                                          EM   640
30    XSTAR = 34.5*F**.40                                               EM   641
40    XRSTAR = 3.5*XSTAR                                                EM   642
      DH = 1.6*((F*XRSTAR**2)**.333333)/UU                              EM   643
      IF(ZBM.LE.0.0)   GO TO 100                                        EM   644
      DHMAX = DH                                                        EM   645
C     -------  NEUTRAL OR UNSTABLE WITH WIND BELOW A STABLE LAYER  -----EM   646
C              DEFINE ZR = ZEQ/ZB                                       EM   647
```

```
C                    A  = (3*F/(BETPSQ*U*S*ZB**3)-2/9)*.666666          EM   648
C               SOLVE FOR THE ROOTS OF                                  EM   649
C                     FCT(ZR) = 0 = ZR**3 - ZR**2 - A                   EM   650
C               USING THE NEWTON-RAPHSON METHOD                         EM   651
60    A = .666666*(3.*F/(BETPSQ*UU*S*ZB**3) -.2222222)                  EM   652
      ZR = .75                                                          EM   653
      DO 70 I = 1,NITRAT                                                EM   654
      FCT = ZR**3 - ZR**2 - A                                           EM   655
      DFCT = 3.*ZR**2 -2.*ZR                                            EM   656
      ZRNU = ZR - FCT/DFCT                                              EM   657
      ERR = ABS(ZRNU-ZR)                                                EM   658
      IF(ERR.LE.ERLMT)  GO TO 80                                        EM   659
      ZR = ZRNU                                                         EM   660
70    CONTINUE                                                          EM   661
      DH = ZR*ZB                                                        EM   662
      WRITE(6,1) I,ERLMT,HTINV,HTPS,DH                                  EM   663
      GO TO 90                                                          EM   664
80    DH = ZR*ZB                                                        EM   665
      DH = AMIN1(DHMAX,DH)                                              EM   666
      IF(DEBUG.EQ.YES) WRITE(6,2) I,HTINV,HTPS,ZB,DH                    EM   667
90    PENTFR = 1.5 - 1./ZR                                              EM   668
      PENTFR = AMAX1(0.0,PENTFR)                                        EM   669
      PENTFR = AMIN1(1.0,PENTFR)                                        EM   670
100   RETURN                                                            EM   671
C                                                                       EM   672
1     FORMAT(1H0,10X,61HCONVERGENCE TO ROOTS OF PLUME RISE EQUATION NOT EM   673
     X ACHIEVED IN   ,I4,12H ITERATIONS  ,/,10X,5F12.3)                 EM   674
2     FORMAT(1H ,10X,8HITERAT = ,I4,9H  HTINV = ,F6.1,9H  HTPS  =, F6.1,EM   675
     1  7H  ZB =   ,F6.1,   6H DH =   ,F6.1 )                          EM   676
      END                                                               EM   677



      FUNCTION DISTAN(IPT,TS,TE)                                        EM   678
C                                                                       EM   679
C        DETERMINES THE DISTANCE ALONG THE TRAJECTORY                   EM   680
C        BETWEEN TIMES *TS* AND *TE*.                                   EM   681
C                                                                       EM   682
      COMMON /WIND/  T(100),  V(100),  TH(100),  NPTS                   EM   683
      COMMON /LINKS/ SEGLEN(100),  RADISQ(100),  RADMUL,  PDISMX(6)     EM   684
      IF(TE.LT.T(NPTS))  GO TO 5                                        EM   685
      DISTAN = 1000000.                                                 EM   686
      RETURN                                                            EM   687
5     IP1 = IPT + 1                                                     EM   688
      IF(T(IP1).LT.TE)  GO TO 10                                        EM   689
      DISTAN = (TE - TS)*V(IPT)                                         EM   690
      RETURN                                                            EM   691
10    DX = (T(IP1) - TS)*V(IPT)                                         EM   692
15    I1 = IP1                                                          EM   693
      IP1 = IP1 + 1                                                     EM   694
      IF(T(IP1) - TE)  20,25,25                                         EM   695
20    DX = DX + SEGLEN(I1)                                              EM   696
      GO TO 15                                                          EM   697
25    DX = DX + (TE - T(I1))*V(I1)                                      EM   698
      DISTAN = DX                                                       EM   699
```

CORRECTION

After EM 670, insert

```
IF (PENTFR.LT. .1) PENTFR=0
```

```
      RETURN                                                            EM   700
      END                                                               EM   701



      FUNCTION ERF(X)                                                   EM   702
C                                                                       EM   703
C        RATIONAL APPROXIMATIION FOR THE ERROR FUNCTION FROM THE NBS    EM   704
C        HANDBOOK OF MATHEMATICAL FUNCTIONS (AMS-55)  PG. 299.          EM   705
C        MAX ERROR  =  2.5 * 10**-5                                     EM   706
C                                                                       EM   707
      DATA A1, A2, A3, P / .3480242, -.0958798, .7478556, .47047 /      EM   708
      T = ABS(X)                                                        EM   709
      IF(T-5.)  2, 2, 10                                                EM   710
2     V = 1./(1.+P*T)                                                   EM   711
      Q = 1. - ((((A3*V)+A2)*V+A1)*V)*EXP(-T*T)                         EM   712
      GO TO 20                                                          EM   713
10    Q = 1.000000                                                      EM   714
20    ERF = SIGN(Q,X)                                                   EM   715
      RETURN                                                            EM   716
      END                                                               EM   717



      SUBROUTINE GESTAB(TIME,ISTAB,DTNEXT,NXSTAB)                       EM   718
C                                                                       EM   719
C        DETERMINES STABILITY CLASS                                     EM   720
C                                                                       EM   721
C         ISTAB  = STABILITY CLASS AT TIME SPECIFIED                    EM   722
                                                                        EM   723
C         DTNEXT = TIME TO NEXT STABILITY CLASS CHANGE                  EM   724
C         NXSTAB = STABILITY CLASS AT NEXT UPDATE TIME                  EM   725
C                                                                       EM   726
      COMMON /PASQAL/  TMSTAB(25),  KSTABL(25),  ISTBMX                 EM   727
      IF(TIME.LT.TMSTAB(ISTBMX))  GO TO 10                              EM   728
5     ISTAB = KSTABL(ISTBMX)                                            EM   729
      DTNEXT = 1000000.                                                 EM   730
      RETURN                                                            EM   731
10    CONTINUE                                                          EM   732
      IMAX = ISTBMX - 1                                                 EM   733
      DO 15 I = 1,IMAX                                                  EM   734
      IP = I + 1                                                        EM   735
      IF( (TIME.GE.TMSTAB(I)) .AND. (TIME.LT.TMSTAB(IP)))  GO TO 20     EM   736
15    CONTINUE                                                          EM   737
      GO TO 5                                                           EM   738
20    ISTAB = KSTABL(I)                                                 EM   739
      NXSTAB = KSTABL(IP)                                               EM   740
      DTNEXT = TMSTAB(IP) - TIME                                        EM   741
      RETURN                                                            EM   742
      END                                                               EM   743
```

```
      SUBROUTINE GRIDIT(LUGRID,KOK)                                     EM   744
C                                                                       EM   745
C        *GRIDIT* READS THE RAPS/ST.LOUIS EMISSIONS GRID                EM   746
C                 DISCRIPTION FILE FROM LUGRID,                         EM   747
C                 IT STORES ONLY THE GRID SQUARE IDENTIFIERS AND        EM   748
C                 CORRESPONDING AREAS FOR USE IN *AREAEM*,              EM   749
C                                                                       EM   750
C     RAPS FILE DESCRIPTION                                             EM   751
C          NGRID                                                        EM   752
C          ID  IAREA  UTMH  UTMV  STATE  COUNTY  HITE  RO               EM   753
C          ID ... REPEATED NGRID TIMES                                  EM   754
C                                                                       EM   755
C                                                                       EM   756
      DIMENSION   IWORK(16000),   IAREA(2000)                           EM   757
      COMMON /ANSWER/  YES,RNEG,    IYES,NO,        SMALL               EM   758
      COMMON /DEGRID/  ID(2000),    AREA(2000),     NGRID               EM   759
      COMMON /WORKER/  IDATE,    NG,   IHR,  WORK(22440)                EM   760
      EQUIVALENCE(IWORK,WORK)                                           EM   761
      EQUIVALENCE(IAREA,WORK(16001))                                    EM   762
      DATA NUMREC, NWPREC, NVAR /32,510, 8/                             EM   763
      DATA ALL, LOUT /4HALL , 6/                                        EM   764
      DEBUG = RNEG                                                      EM   765
      IREC = 1                                                          EM   766
      K1 = 0                                                            EM   767
      K2 = NWPREC - 1                                                   EM   768
C                                                                       EM   769
C       OI. THE UNIVAC REPLACE BUFFER IN AND IF UNIT WITH               EM   770
C     READ(LUGRID,ERR=220,END=210) IHR,(WORK(JJ),JJ=1,K2)               EM   771
C                                                                       EM   772
      BUFFER IN (LUGRID,1) (IHR,WORK(K2))                               EM   773
120   IF(UNIT(LUGRID))    130,   210,   220                             EM   774
130   NGRID = IHR                                                       EM   775
      IF(DEBUG.NE.ALL)  GO TO 140                                       EM   776
      WRITE(LOUT,5) NGRID                                               EM   777
      WRITE(LOUT,6) (IWORK(I),I=K1,K2)                                  EM   778
140   CONTINUE                                                          EM   779
      K2MAX = NVAR*NGRID                                                EM   780
      DO 170 IREC = 2,NUMREC                                            EM   781
      K1 = K1 + NWPREC                                                  EM   782
      K2 = K2 + NWPREC                                                  EM   783
      IF(IREC.EQ.NUMREC)  K2 = MIN0(K2,K2MAX)                           EM   784
C        ON THE UNIVAC REPLACE BUFFER IN AND IF(UNIT) WITH              EM   785
C     READ(LUGRID,ERR=220,END=210) (WORK(JJ),JJ=K1,K2)                  EM   786
C                                                                       EM   787
      BUFFER IN (LUGRID,1)  (WORK(K1),WORK(K2))                         EM   788
150   IF(UNIT(LUGRID))   160,  210,  220                                EM   789
160   IF(DEBUG.NE.ALL)  GO TO 170                                       EM   790
      WRITE(LOUT,6)  IREC                                               EM   791
      WRITE(LOUT,6) (IWORK(I),I=K1,K2)                                  EM   792
170   CONTINUE                                                          EM   793
      K1 = 1                                                            EM   794
      K2 = 2                                                            EM   795
      DO 180 J = 1,NGRID                                                EM   796
      ID(J) = IWORK(K1)                                                 EM   797
      IAREA(J) = IWORK(K2)                                              EM   798
```

```
      AREA(J) = FLOAT(IAREA(J))                                          EM   799
      K1 = K1 + NVAR                                                     EM   800
      K2 = K2 + NVAR                                                     EM   801
180   CONTINUE                                                           EM   802
      IF( (DEBUG.EQ.YES).OR.(DEBUG.EQ.ALL) )  GO TO 190                  EM   803
      RETURN                                                             EM   804
190   WRITE(LOUT,5)  NGRID                                               EM   805
      WRITE(LOUT,7)  (ID(J), IAREA(J), J=1,NGRID)                        EM   806
      RETURN                                                             EM   807
210   KOK = 9                                                            EM   808
      WRITE(LOUT,4) KOK                                                  EM   809
      RETURN                                                             EM   810
220   KOK = 99                                                           EM   811
      WRITE(LOUT,3) KOK                                                  EM   812
      RETURN                                                             EM   813
3     FORMAT (44H1PARITY ERROR ON GRID FILE  -- JOB ABORTED  ,I8)        EM   814
4     FORMAT (46H1UNEXPECTED EOF IN GRID FILE  -- JOB ABORTED  ,I8)      EM   815
5     FORMAT (1H0,10H    NGRID=  ,I6)                                    EM   816
6     FORMAT (1H ,3(8I5,3X))                                             EM   817
7     FORMAT (1H ,10(2I4,4X))                                            EM   818
      END                                                                EM   819



      SUBROUTINE LOCATE(IPT,XP,YP,PDIST,TPASS,KFLAG,ISTAB)               EM   820
C                                                                        EM   821
C         LOCATE DETERMINES WHETHER OR NOT A POINT SOURCE                EM   822
C         IS PASSED ON A TRAJECTORY SEGMENT AND WHETHER OR               EM   823
C         NOT THE POINT SOURCE IS CLOSE ENOUGH TO BE CONSIDERED.         EM   824
C                                                                        EM   825
C           IPT    =   TRAJECTORY NODE INDEX                           EM   826
C           XP     =   X COORDINATE OF POINT SOURCE IN KM              EM   827
C           YP     =   Y COORDINATE OF POINT SOURCE IN KM              EM   828
C           PDIST  =   PERPENDICULAR DISTANCE TO POINT SOURCE IN KM    EM   829
C           TPASS  =   TIME SOURCE IS PASSED                           EM   830
C           KFLAG  =   FLAG WHICH IS POSITIVE IF SOURCE MEETS ALL      EM   831
C                      CRITERIA FOR CONSIDERATION ON TRAJECTORY SEGMENT IEM   832
C           ISTAB  =   STABILITY CLASS                                 EM   833
C                                                                        EM   834
      DIMENSION DX(2), DY(2), DIST(2), ALFA(2), BETA(2)                  EM   835
C                                                                        EM   836
      COMMON  /WIND/  T(100),  V(100),  TH(100),  NPTS                   EM   837
      COMMON /LINKS/  SEGLEN(100),  RADISQ(100),  RADMUL, PDISMX(6)      EM   838
      COMMON  /TRAJ/  P(2,100)                                           EM   839
      COMMON /ANGLES/ GAMA1(100),  GAMA2(100),  ETA1(100),  ETA2(100)    EM   840
      DATA PIE20 /6.283185307/                                           EM   841
C                                                                        EM   842
      DX(1) = XP - P(1,IPT)                                              EM   843
      DY(1) = YP - P(2,IPT)                                              EM   844
      DISTSQ = DX(1)**2  +  DY(1)**2                                     EM   845
C         CHECK DISTANCE TO POINT SOURCE RELATIVE TO RADMUL*SEGMENT LENGEM   846
      IF(DISTSQ.LE.RADISQ(IPT))  GO TO 10                                EM   847
      KFLAG = -1                                                         EM   848
      RETURN                                                             EM   849
10    BETA(1) = ATAN2(DY(1),DX(1))                                       EM   850
```

```
      IF(BETA(1).LT. 0.0)  BETA(1) = BETA(1) + PIE20                    EM   851
      ALFA(1) = BETA(1) - TH(IPT)                                       EM   852
      IF(ALFA(1).LT. 0.0)  ALFA(1) = ALFA(1) + PIE20                    EM   853
      IF(ALFA(1).GT. PIE20)  ALFA(1) = ALFA(1) - PIE20                  EM   854
      DIST(1) = SQRT(DISTSQ)                                            EM   855
      PDIST = DIST(1)*SIN(ALFA(1))                                      EM   856
C         CHECK PERPENDICULAR DISTANCE                                  EM   857
      IF(ABS(PDIST).LT.PDISMX(ISTAB))  GO TO 20                         EM   858
      KFLAG = -1                                                        EM   859
      RETURN                                                            EM   860
20    CONTINUE                                                          EM   861
      IP1 = IPT + 1                                                     EM   862
C         CHECK BISECTOR ANGLES                                         EM   863
      DX(2) = XP - P(1,IP1)                                             EM   864
      DY(2) = YP - P(2,IP1)                                             EM   865
      BETA(2) = ATAN2(DY(2),DX(2))                                      EM   866
      IF(BETA(2).LT. 0.00)   BETA(2) = BETA(2) + PIE20                  EM   867
      ALFA(2) = BETA(2) - TH(IPT)                                       EM   868
      IF(ALFA(2) .LT. 0.00)  ALFA(2) = ALFA(2) + PIE20                  EM   869
      IF(ALFA(2) .GT. PIE20)  ALFA(2) = ALFA(2) - PIE20                 EM   870
      KFLAG = -1                                                        EM   871
C                                                                       EM   872
      IF(ETA1(IPT) .GE. GAMA1(IPT))  GO TO  60                          EM   873
C                                                                       EM   874
40    IF(PDIST .LT. 0.00)   GO TO 50                                    EM   875
      IF((ALFA(1).LE.GAMA1(IPT)) .AND. (ALFA(2).GT.ETA1(IPT))) KFLAG = 1EM   876
      GO TO 100                                                         EM   877
50    IF((ALFA(1).GE.GAMA2(IPT)) .AND. (ALFA(2).LT.ETA2(IPT))) KFLAG = 2EM   878
      GO TO 100                                                         EM   879
60    IF(PDIST .GE. 0.00)   GO TO 70                                    EM   880
      IF((ALFA(1).GE.GAMA2(IPT)) .AND. (ALFA(2).LT.ETA2(IPT))) KFLAG = 3EM   881
      GO TO 100                                                         EM   882
70    IF((ALFA(1).LE.GAMA1(IPT)) .AND. (ALFA(2).GT.ETA1(IPT))) KFLAG = 4EM   883
100   IF(KFLAG.LT.0)  RETURN                                            EM   884
      DAXIAL = DIST(1)*COS(ALFA(1))                                     EM   885
      TPASS = T(IPT) + DAXIAL/V(IPT)                                    EM   886
      TPASS = AMAX1(T(IPT),TPASS)                                       EM   887
      TPASS = AMIN1(T(IP1),TPASS)                                       EM   888
      RETURN                                                            EM   889
      END                                                               EM   890



      SUBROUTINE PARTIT(X,HT,HTINVR,HTINTF,NOSTAT,ISTAB,TALL,PENTFR     EM   891
     1                                                    ,VFRACT)      EM   892
C                                                                       EM   893
C       *PARTIT*  PERFORMS THE VERTICAL INTEGRATION OF THE GAUSSIAN     EM   894
C       DISTRIBUTION OVER VERTICAL CELLS DEFINED BY HTINTF.  IT ASSUMES EM   895
C       A REFLECTION AT THE SURFACE AND AT THE INVERSION BASE.          EM   896
C       THE MAXIMUM VALUE OF THE INTEGRAL HAS BEEN NORMALIZED TO        EM   897
C       ONE MINUS THE INVERSION PENETRATION FRACTION.                   EM   898
C                                                                       EM   899
C   INPUTS                                                              EM   900
C       X =       DOWNWIND DISTANCE (KM)                                EM   901
C       HT =      SOURCE EFFECTIVE HEIGTH (METERS)                      EM   902
```

```
C       HTINVR = HEIGHT OF INVERSION (METERS)                          EM   903
C       HTINTF = HEIGHTS OF CELL INTERFACES  (METERS)                  EM   904
C       NOSTAT = NUMBER OF CELLS                                       EM   905
C       ISTAB =  STABILTY (1=6 CORRESPONDING TO A=F)                   EM   906
C       TALL  =  TALL STACK INDICATOR (EQUALS YES OR NO)               EM   907
C       PENTFR = INVERSION PENETRATION FRACTION (0 TO 1)               EM   908
C   OUTPUT                                                             EM   909
C       VFRACT = VECTOR OF FRACTIONS OF NORMAL DISTRIBUTION            EM   910
C                IN EACH CELL (DEFINED BY HTINTF)                      EM   911
C                                                                      EM   912
      DIMENSION  ZE(6),   HTINTF(1),    VFRACT(1)                      EM   913
      COMMON /SIGMAS/  NDY,   XDY(4),   SIGYD(4,6),  NDZ, XDZ(12),    EM   914
     1                 SIGZD(12,6),     NCAT                         EM   915
      COMMON /SIGTAL/ ATALL(6), BTALL(6), CTALL(6), DTALL(6)           EM   916
      DATA SQRT2 /1.414213562/                                         EM   917
      DATA YES, RNEG, DZINV /4HYES ,4HNO  ,10./                        EM   918
C                                                                      EM   919
      DO 10 I = 1,NOSTAT                                               EM   920
      VFRACT(I) = 0.000                                                EM   921
10    CONTINUE                                                         EM   922
      CPENTR = 1.000 - PENTFR                                          EM   923
      NSP1 = NOSTAT + 1                                                EM   924
      S = -1.00                                                        EM   925
      IF(HT.GE.HTINTF(NSP1))  GO TO 200                                EM   926
C                                                                      EM   927
      IF(TALL.EQ.YES)  GO TO 5                                         EM   928
      SIGZ = TRPLOG(X,XDZ,SIGZD(1,ISTAB),NDZ,1)                        EM   929
      GO TO 8                                                          EM   930
5     SIGZ = CTALL(ISTAB)*X**DTALL(ISTAB)                              EM   931
8     SIGZ2 = SIGZ*SQRT2                                               EM   932
      DO 15 I = 2,NSP1                                                 EM   933
      IF(HTINVR.LT.HTINTF(I))  GO TO 20                                EM   934
15    CONTINUE                                                         EM   935
20    IMAX = I - 1                                                     EM   936
C     IF(HTINVR - HT) 120,120,25     .                                 EM   937
      IF(PENTFR.GT. 0.99) GO TO 120                                    EM   938
C     ******** SOURCE HEIGHT BELOW INVERSION HEIGHT                    EM   939
25    DO  70 NP = 1,2                                                  EM   940
      NBELOW = 0                                                       EM   941
      DO 30 J = 1,NSP1                                                 EM   942
      ZE(J) = HTINTF(J) + HT*S                                         EM   943
      IF(ZE(J).LT.0.00)   NBELOW = NBELOW + 1                          EM   944
30    CONTINUE                                                         EM   945
      HTLMT = HTINVR + HT*S                                            EM   946
      I = 1                                                            EM   947
      IF(NBELOW.LT.1)  GO TO 50                                        EM   948
      DO 40 I = 1,NBELOW                                               EM   949
      ZF = ABS(ZE(I)/SIGZ2)                                            EM   950
      ZN = ZE(I+1)/SIGZ2                                               EM   951
      ZN = AMIN1(ZN,0.0000)                                            EM   952
      ZN = ABS(ZN)                                                     EM   953
      VFRACT(I) = VFRACT(I) + .5*(ERF(ZF) - ERF(ZN))                   EM   954
40    I=NBELOW                                                         EM   955
50    CONTINUE                                                         EM   956
      DO 60 J = I,NOSTAT                                               EM   957
```

```
      ZN = ZE(J)/SIGZ2                                                  EM   958
      ZN = AMAX1(ZN,0.000)                                              EM   959
      ZF = ZE(J+1)                                                      EM   960
      IF (ZF.LT.HTLMT)  GO TO 55                                        EM   961
      ZF = 10000.                                                       EM   962
      VFRACT(K) = VFRACT(K) + .5*(ERF(ZF) - ERF(ZN))                    EM   963
      GO TO 65                                                          EM   964
55    ZF = ZF/SIGZ2                                                     EM   965
      VFRACT(J) = VFRACT(J) + .5*(ERF(ZF) - ERF(ZN))                    EM   966
60    CONTINUE                                                          EM   967
65    S = 1.00                                                          EM   968
70    CONTINUE                                                          EM   969
      DO 80 I = 1,NOSTAT                                                EM   970
      S = S - VFRACT(I)                                                 EM   971
80    CONTINUE                                                          EM   972
C     --------- DISTRIBUTE REMAINDER EVENLY BELOW THE INVERSION         EM   973
      SPMETR = S/HTINVR                                                 EM   974
      DO 90 I = 1,IMAX                                                  EM   975
      HT1 = AMIN1(HTINVR,HTINTF(I+1))                                   EM   976
      ZF = SPMETR*(HT1 - HTINTF(I))                                     EM   977
      VFRACT(I) = (VFRACT(I) + ZF)*CPENTR                               EM   978
90    CONTINUE                                                          EM   979
      HT1 = HTINTF(IMAX+1) - HTINVR                                     EM   980
      IF(HT1 - DZINV) 100,100,110                                       EM   981
100   IMAX = MIN0(IMAX+1,NOSTAT)                                        EM   982
      WRITE(6,7)                                                        EM   983
7     FORMAT(1H ,* INVERSION HEIGHT CLOSE TO TOP OF CELL --  PENETRATIONEM   984
     1  FRACTION PLACED IN NEXT CELL *)                                 EM   985
110   VFRACT(IMAX) = VFRACT(IMAX) + PENTFR                              EM   986
      RETURN                                                            EM   987
C                                                                       EM   988
120   CONTINUE                                                          EM   989
C       -------- SOURCE HEIGHT ABOVE INVERSION HEIGHT                  EM   990
      DO 130 J = 2,NSP1                                                 EM   991
      IF(HT.LT.HTINTF(J))  GO TO 140                                    EM   992
130   CONTINUE                                                          EM   993
140   JMAX = J - 1                                                      EM   994
      VFRACT(JMAX) = VFRACT(JMAX) +  1.00                               EM   995
      RETURN                                                            EM   996
C       -------- SOURCE HEIGHT ABOVE TOP OF MESH                       EM   997
200   WRITE(6,1)  HT,HTINTF(NSP1)                                       EM   998
      VFRACT(NOSTAT) = 1.00                                             EM   999
      RETURN                                                            EM  1000
1     FORMAT(1H0, 25HEFFECTIVE SOURCE HEIGHT = ,F10.1,//,               EM  1001
     118H MESH HEIGHT    =  F10.1,  //, 20H SOURCE IN TOP CELL        ) EM  1002
      END                                                               EM  1003
```

CORRECTION

Insert after EM979

```
IF(VFRACT(I).LT. 1.E-4) VFRACT(I)=0.
```

```
      SUBROUTINE PLUMAS(IPT,TPASS,PDIST,PLUMFR,ISTAB,DXFREZ,TALL,KPLAG) EM  1004
C                                                                       EM  1005
C       *PLUMAS* PERFORMS THE LATERAL INTERGRATION OF THE GUASSIAN       EM  1006
C       DISTRIBUTION TO DETERMINE THE PORTION OF THE PLUME'S MASS       EM  1007
C       REMAINING IN THE LAGRANGIAN AIR PARCEL AFTER DTFREZ MINUTES.    EM  1008
C                                                                       EM  1009
C       IPT       = TRAJECTORY NODE INDEX                               EM  1010
C       TPASS     = TIME AT WHICH POINT SOURCE WAS PASSED               EM  1011
C       PDIST     = PERPENDICULAR DISTANCE TO SOURCE                    EM  1012
C       PLUMFR    = RELATIVE PORTION OF PLUME'S MASS REMAINING IN PARCEL EM  1013
C                   (  0.< PLUMFR < 1.0  )                              EM  1014
C       ISTAB     = STABILITY (1=6 CORRESPONDING TO A-F) RETURNED       EM  1015
C       DXFREZ    = DOWNWIND DISTANCE TRAVELLED IN DTFREZ CORRECTED     EM  1016
C                   FOR STABILITY CLASS CHANGES.                        EM  1017
C       TALL      = TALL STACK (GT 100 METERS) FLAG = YES OR NO         EM  1018
C       KPLAG     =  1      LATERAL DIFFUSION OUT OF PARCEL IS DOMINANT EM  1019
C       KPLAG     =  2      LATERAL DIFFUSION INTO PARCEL IS DOMINANT   EM  1020
C       KPLAG     = -1      PLUME'S LATERAL SPREAD IS TOO SLOW TO REACH EM  1021
C                           PARCEL   --  SOURCE SHOULD BE IGNORED ENTIRELEM  1022
C                                                                       EM  1023
      COMMON /SIGMAS/  NDY,  XDY(4),  SIGYD(4,6),  NDZ,  XDZ(12),      EM  1024
     1                 SIGZD(12,6),  NCAT                               EM  1025
      COMMON /ANSWER/  YES, RNEG, IYES, NO, SMALL                       EM  1026
      COMMON /SIGTAL/  ATALL(6), BTALL(6), CTALL(6), DTALL(6)           EM  1027
      COMMON /PARCEL/ PWIDTH, HPWIDT, PLENTH, PAREA, DTFREZ, SIGED0,    EM  1028
     1                ZEE(5), HTINTF(6),  NOSTAT,  ROAIR               EM  1029
      DATA SQRT2,   SMALL   /1.414213562,  .020/                        EM  1030
C                                                                       EM  1031
      KPLAG = - 1                                                       EM  1032
      IP1 = IPT + 1                                                     EM  1033
      PDIST = ABS(PDIST)                                                EM  1034
      TFREEZ = TPASS + DTFREZ                                           EM  1035
      DXFREZ = DISTAN(IPT,TPASS,TFREEZ)                                 EM  1036
      VBAR = DXFREZ/DTFREZ                                              EM  1037
      CALL GESTAB(TPASS,ISTAB,DTSTCH,NXSTAB)                            EM  1038
      IF(DTSTCH.GT.DTFREZ-5.) GO TO 100                                 EM  1039
C                                                                       EM  1040
C       ADJUST DXFREZ TO ACCOUNT FOR STABILITY CLASS CHANGE             EM  1041
C                                                                       EM  1042
      TSTCH = TPASS + DTSTCH                                            EM  1043
      DXSTCH = DISTAN(IPT,TPASS,TSTCH)                                  EM  1044
      IF(TALL.EQ.YES)   GO TO 80                                        EM  1045
      SIGSCH = TRPLOG(DXSTCH,XDY,SIGYD(1,ISTAB),NDY,1)                  EM  1046
      DXVIRT = TRPLOG(SIGSCH,SIGYD(1,NXSTAB),XDY,NDY,1)                 EM  1047
      GO TO 90                                                          EM  1048
80    SIGSCH = ATALL(ISTAB)*(DXSTCH**BTALL(ISTAB))                      EM  1049
      DXVIRT = (SIGSCH/ATALL(NXSTAB))**(1./BTALL(NXSTAB))               EM  1050
90    DXDIFF = DXSTCH - DXVIRT                                          EM  1051
      DXFREZ = DXFREZ - DXDIFF                                          EM  1052
      ISTAB = NXSTAB                                                    EM  1053
C                                                                       EM  1054
C       INTEGRATE PLUMES FROM SOURCES WITHIN PARCEL WIDTH               EM  1055
C                                                                       EM  1056
100   CONTINUE                                                          EM  1057
      IF(TALL.EQ.YES)  GO TO 110                                        EM  1058
```

```
      SIGY = TRPLOG(DXFREZ,XDY,SIGYD(1,ISTAB),NDY,1)                    EM  1059
      GO TO 115                                                         EM  1060
110   SIGY = ATALL(ISTAB)*(DXFREZ**BTALL(ISTAB))                        EM  1061
115   SIGYSP = SIGY*SQRT2                                               EM  1062
C                                                                       EM  1063
      IF(PDIST.GT.HPWIDT)  GO TO 120                                    EM  1064
C                                                                       EM  1065
      D1 = (HPWIDT - PDIST)/SIGYSP                                      EM  1066
      D2 = (HPWIDT + PDIST)/SIGYSP                                      EM  1067
      PLUMFR = 0.50*(ERF(D1) + ERF(D2))                                 EM  1068
      KPLAG = 1                                                         EM  1069
      RETURN                                                            EM  1070
C                                                                       EM  1071
C       INTEGRATE PLUMES FROM SOURCES OUTSIDE PARCEL WIDTH              EM  1072
C       AND ADJUST TPASS TO ACCOUNT FOR THE TIME DELAY BEFORE           EM  1073
C       INTERSECTION OF THE PLUME.                                      EM  1074
C                                                                       EM  1075
120   CONTINUE                                                          EM  1076
      IF(PDIST.LT. HPWIDT + SMALL)  GO TO 140                           EM  1077
      SIGY2 = (PDIST - HPWIDT)/SIGEDG                                   EM  1078
      IF(TALL.EQ.YES)  GO TO 125                                        EM  1079
      DXEDG = TRPLOG(SIGY2,SIGYD(1,ISTAB),XDY,NDY,1)                    EM  1080
      GO TO 130                                                         EM  1081
125   DXEDG = (SIGY2/ATALL(ISTAB))**(1./BTALL(ISTAB))                   EM  1082
130   IF(DXEDG .GT.DXFREZ)   RETURN                                     EM  1083
      TCROSS = TFREEZ *(DXFREZ - DXEDG)/VBAR                            EM  1084
      TPASS = AMAX1(TPASS,TCROSS)                                       EM  1085
140   CONTINUE                                                          EM  1086
      D1 = (PDIST - HPWIDT)/SIGYSP                                      EM  1087
      D2 = (PDIST + HPWIDT)/SIGYSP                                      EM  1088
      PLUMFR = 0.50*(ERF(D2) - ERF(D1))                                 EM  1089
      KPLAG = 2                                                         EM  1090
      RETURN                                                            EM  1091
      END                                                               EM  1092
```

```
      SUBROUTINE PONTEM(LUPONT,IPUNCH,KOK,NOSUMS)                       EM  1093
C                                                                       EM  1094
C          *PONTEM* DETERMINES THE POINT SOURCE EMISSIONS                EM  1095
C          HISTORY ALONG A SPECIFIED TRAJECTORY.                        EM  1096
C                                                                       EM  1097
C          THIS IS THE RAPS VERSION OF PONTEM                           EM  1098
C                                                                       EM  1099
      DIMENSION         VFRACT(5),      IWORK(200),    WORK1(200),      EM  1100
     1                  WORK2(5,200),   FULRAT(7,200), WORK4(7,200)     EM  1101
      COMMON /INPUTS/   TITLE(20),      IDATE(10),     NCASE            EM  1102
      COMMON /WORKER/   IXT(3),         WORK(22440)                     EM  1103
      COMMON /ANSWER/   YES,RNEG,       IYES,NO,       SMALL            EM  1104
      COMMON /LABLIN/   NAMIN(11),      NFLXIN,        WTMOLI(11),      EM  1105
     1                  ADJUST(11)                                      EM  1106
      COMMON /LABOUT/   NAMOUT(7),      NFXOUT,        WTMOLO(7)        EM  1107
      COMMON /TABLES/   THOUR(24)                                       EM  1108
      COMMON /PASSES/   TPASS(200),     VFR(5,200),    PSMASS(7,200),   EM  1109
     1                  PMFR(200),      PSPD(200),     NKPS             EM  1110
```

```
      COMMON /TRAJ/     P(2,100)                                                EM  1111
      COMMON /WIND/     T(100),         V(100),         TH(100),                EM  1112
     1                  NPTS                                                    EM  1113
      COMMON /LINKS/    SEGLEN(100),    RADISQ(100),    RADMUL,                 EM  1114
     1                  PDISMX(6)                                               EM  1115
      COMMON /PARCEL/   PWIDTH,         HPWIDT,         PLENTH,                 EM  1116
     1                  PAREA,          DTFREZ,         SIGEDG,                 EM  1117
     2                  ZEE(5),         HTINTF(6),      NUSTAT,                 EM  1118
     3                  RDAIR                                                   EM  1119
      COMMON /TEMPS/    TMTEMP(25),     TEMPSF(25),     NTEMP,ITEMP             EM  1120
      COMMON /MXHITE/   TMXHIT(25),     HTMIXL(25),     NMIXL,IMIXL             EM  1121
      COMMON /PSOAT/    PSXY(2,505),    PSRAT(7,505), PSTEMP(505),              EM  1122
     1                  PSFLOW(505),    PSHITE(505),    NPS                     EM  1123
      COMMON /ORIGIN/   UTMXOR,         UTMYOR                                  EM  1124
      COMMON /ANGLES/   GAMA1(100),     GAMA2(100),     ETA1(100),              EM  1125
     1                  ETA2(100)                                               EM  1126
      COMMON /SIGMAS/   NDY,            XDY(4),         SIGYD(4,6),             EM  1127
     1                  NDZ,            XDZ(12),        SIGZD(12,6),NCAT        EM  1128
      EQUIVALENCE(IWORK,WORK(1))                                                EM  1129
      EQUIVALENCE(WORK1,WORK(201))                                              EM  1130
      EQUIVALENCE(WORK2,WORK(401))                                              EM  1131
      EQUIVALENCE(WORK4,WORK(1401))                                             EM  1132
      EQUIVALENCE(FULRAT,WORK(2801))                                            EM  1133
      DATA VOXFRC, KPSMAX  /0.3333,  200/                                       EM  1134
      DATA LOUT, LPUNCH /6,1/                                                   EM  1135
C                                                                               EM  1136
      DEBUG = RNEG                                                              EM  1137
      KOK = 100                                                                 EM  1138
      NM1 = NPTS - 1                                                            EM  1139
      KPS = 0                                                                   EM  1140
      IWAS = 0                                                                  EM  1141
      DTDZ = .0137                                                              EM  1142
      DO 50 I = 1,NPTS                                                          EM  1143
      P(1,I) = P(1,I) - UTMXOR                                                  EM  1144
      P(2,I) = P(2,I) - UTMYOR                                                  EM  1145
50    CONTINUE                                                                  EM  1146
      CALL SEGMET(P)                                                            EM  1147
C                                                                               EM  1148
      CALL STACKS(T(1),LUPONT,KOK,IWAS,NDSUMS)                                  EM  1149
      IWAS = 1                                                                  EM  1150
C                                                                               EM  1151
      IF(KOK .LT. 0)   RETURN                                                   EM  1152
C                                                                               EM  1153
      DO 200 IL = 1,NM1                                                         EM  1154
      IN = IL + 1                                                               EM  1155
C                                                                               EM  1156
      CALL GESTAB(T(IL),ISTAB,DX,N)                                             EM  1157
C                                                                               EM  1158
      HRFRAC = PLENTH/(V(IL)*60.)                                               EM  1159
      HTV =HTMIXL(IMIXL)                                                        EM  1160
      TA = TEMPSF(ITEMP)                                                        EM  1161
C                                                                               EM  1162
C       LOCATE POINT SOURCES AND INTEGRATE THEIR PLUMES                         EM  1163
C                                                                               EM  1164
      DO 100 JS = 1,NPS                                                         EM  1165
```

```
C                                                                       EM  1166
      CALL LOCATE(IL,PSXY(1,JS),PSXY(2,JS),PDIST,TPASIT,KFLAG,ISTAB)    EM  1167
C                                                                       EM  1168
      IF(KFLAG.LT.0)   GO TO 100                                        EM  1169
      IF(DEBUG.EQ.YES)                                                  EM  1170
     1 WRITE(LOUT,21) JS,PSXY(1,JS),PSXY(2,JS),PD ST,TPASIT,ISTAB,KFLAG EM  1171
C                                                                       EM  1172
      TALL = RNEG                                                       EM  1173
      HTPS = PSHITE(JS)                                                 EM  1174
      IF(HTPS.GE.100.)  TALL = YES                                      EM  1175
      CALL PLUMAS(IL,TPASIT,PDIST,PMFRAC,ISTAB,DXFREZ,TALL,KPLAG)       EM  1176
C                                                                       EM  1177
      IF(KPLAG.LT.0)   GO TO 100                                        EM  1178
      IF(DEBUG.EQ.YES)                                                  EM  1179
     1 WRITE(LOUT,21)JS,TPASIT,PDIST,PMFRAC,DXFREZ,ISTAB,KPLAG          EM  1180
      DO 70 JR = 1,NFXOUT                                               EM  1181
      IF(PSRAT(JR,JS).GT.0.0)  GO TO 75                                 EM  1182
70    CONTINUE                                                          EM  1183
      GO TO 100                                                         EM  1184
75    KPS = KPS + 1                                                     EM  1185
      IF(KPS.GT.KPSMAX)   GO TO 510                                     EM  1186
      TPASS(KPS) = TPASIT                                               EM  1187
      PMFR(KPS) = PMFRAC                                                EM  1188
      PSPD(KPS) = PDIST                                                 EM  1189
      F = PSFLOW(JS)                                                    EM  1190
      TS = PSTEMP(JS)                                                   EM  1191
      DH = 0.0                                                          EM  1192
      IF(TS.LE.TA)  GO TO 90                                            EM  1193
      IF(F.LT.SMALL)  GO TO 90                                          EM  1194
      U = V(IL)*16.6667                                                 EM  1195
      CALL DHPLUM(TA,TS,U,ISTAB,F,HTV,HTPS,DH,PF,DTDZ)                  EM  1196
      IF(DEBUG.EQ.YES) WRITE(LOUT,21) JS,HTEFF,HTPS,HTV                 EM  1197
90    HTEFF = DH + HTPS                                                 EM  1198
      DXFREZ = VDXFRC*DXFREZ                                            EM  1199
C                                                                       EM  1200
      CALL PARTIT(DXFREZ,HTEFF,HTV,HTINTF,NOSTAT,ISTAB,TALL,PF,VFRACT)  EM  1201
C                                                                       EM  1202
      IF(DEBUG.EQ.YES) WRITE(LOUT,20)  KPS, VFRACT                      EM  1203
      CALL XMIT(NOSTAT,VFRACT,VFR(1,KPS))                               EM  1204
      DO 95 K = 1,NFXOUT                                                EM  1205
      FULRAT(K,KPS) = PSRAT(K,JS)                                       EM  1206
      PSMASS(K,KPS) = PMFRAC*HRFRAC*PSRAT(K,JS)                         EM  1207
95    CONTINUE                                                          EM  1208
100   CONTINUE                                                          EM  1209
C                                                                       EM  1210
      IF(IN.EQ.NPTS)  GO TO 210                                         EM  1211
      IF(ITEMP.GE.NTEMP)  GO TO 120                                     EM  1212
      IF(T(IN).LT.TMTEMP(ITEMP+1)-SMALL)  GO TO 120                     EM  1213
      ITEMP = ITEMP + 1                                                 EM  1214
120   CONTINUE                                                          EM  1215
      IF(IMIXL.GE.NMIXL) GO TO 130                                      EM  1216
      IF(T(IN).LT.TMXHIT(IMIXL+1)-SMALL)  GO TO 130                     EM  1217
      IMIXL = IMIXL + 1                                                 EM  1218
130   CONTINUE                                                          EM  1219
      CALL STACKS(T(IN),LUPONT,KOK,IWAS,NDSUMS)                         EM  1220
```

```
      IF(KOK .LT. 0)  GO TO 210                                          EM  1221
200   CONTINUE                                                           EM  1222
C                                                                        EM  1223
210   CONTINUE                                                           EM  1224
      IF(KPS .LE. 0)  GO TO 520                                          EM  1225
C                                                                        EM  1226
C        ORDER PASSING ARRAYS CHRONOLOGICALLY                            EM  1227
C                                                                        EM  1228
      DO 220 N = 1,KPS                                                   EM  1229
      IWORK(N) = N                                                       EM  1230
220   CONTINUE                                                           EM  1231
      DO 230 N = 1,KPS                                                   EM  1232
      M = KPS + 1 - N                                                    EM  1233
      CALL FMINF(TPASS(N),M,FMIN,NMIN)                                   EM  1234
      NMIN = NMIN + N - 1                                                EM  1235
      J = IWORK(N)                                                       EM  1236
      IWORK(N) = IWORK(NMIN)                                             EM  1237
      IWORK(NMIN) = J                                                    EM  1238
      XX = TPASS(N)                                                      EM  1239
      TPASS(N) = FMIN                                                    EM  1240
      TPASS(NMIN) = XX                                                   EM  1241
230   CONTINUE                                                           EM  1242
C                                                                        EM  1243
      CALL XMIT(KPS, PMFR, WORK1)                                        EM  1244
      N = 5*KPS                                                          EM  1245
      CALL XMIT(N,VFR,WORK2)                                             EM  1246
      DO 240 N = 1,KPS                                                   EM  1247
      J = IWORK(N)                                                       EM  1248
      PMFR(N) = WORK1(J)                                                 EM  1249
      DO 240 K = 1,NOSTAT                                                EM  1250
      VFR(K,N) = WORK2(K,J)                                              EM  1251
240   CONTINUE                                                           EM  1252
C                                                                        EM  1253
      CALL XMIT(-NFXOUT,0.0,WORK1)                                       EM  1254
      M = 7*KPS                                                          EM  1255
      CALL XMIT(M,PSMASS,WORK4)                                          EM  1256
      DO 250 N = 1,KPS                                                   EM  1257
      J = IWORK(N)                                                       EM  1258
      DO 250 K = 1,NFXOUT                                                EM  1259
      PSMASS(K,N) = WORK4(K,J)                                           EM  1260
      WORK1(K) = WORK1(K) + PSMASS(K,N)                                  EM  1261
250   CONTINUE                                                           EM  1262
      CALL XMIT(KPS,PSPD,SEGLEN)                                         EM  1263
      CALL XMIT(M,FULRAT,WORK4)                                          EM  1264
      DO 260 N = 1,KPS                                                   EM  1265
      J = IWORK(N)                                                       EM  1266
      PSPD(N) = SEGLEN(J)                                                EM  1267
      DO 260 K = 1,NFXOUT                                                EM  1268
      FULRAT(K,N) = WORK4(K,J)                                           EM  1269
260   CONTINUE                                                           EM  1270
C                                                                        EM  1271
C     OUTPUTS                                                            EM  1272
C     1. ACTUAL SOURCE RATES, FRACTION INCLUDED, AND PERPENDICULAR DISTANEM  1273
C     2. ABSOLUTE MASS (MOLES) AND VERTICAL DISTRIBUTION                 EM  1274
C     3. MOLE FRACTION-METER OF PARCEL AIR                               EM  1275
```

```
C                                                                       EM  1276
      CALL NEWPAG(TITLE,0,JDATE)                                        EM  1277
      WRITE(LOUT,5)  (NAMOUT(K),K=1,NFXOUT)                             EM  1278
      DO 310 J = 1,KPS                                                  EM  1279
      IF(MOD(J,25).NE.0)  GO TO 300                                     EM  1280
      CALL NEWPAG(TITLE,0,JDATE)                                        EM  1281
      WRITE(LOUT,5)  (NAMOUT(K),K=1,NFXOUT)                             EM  1282
300   WRITE(LOUT,6) TPASS(J),PMFR(J),PSPD(J),(FULRAT(K,J),K=1,NFXOUT)   EM  1283
310   CONTINUE                                                          EM  1284
      CALL NEWPAG(TITLE,0,JDATE)                                        EM  1285
      WRITE(LOUT,7)  (NAMOUT(K),K=1,NFXOUT)                             EM  1286
      WRITE(LOUT,8)  (HTINTF(K),HTINTF(K+1),K=1,NOSTAT)                 EM  1287
      DO 330 J = 1,KPS                                                  EM  1288
      IF(MOD(J,18).NE.0)  GO TO 320                                     EM  1289
      CALL NEWPAG(TITLE,0,JDATE)                                        EM  1290
      WRITE(LOUT,7)  (NAMOUT(K),K=1,NFXOUT)                             EM  1291
      WRITE(LOUT,8)  (HTINTF(K),HTINTF(K+1),K=1,NOSTAT)                 EM  1292
320   WRITE(LOUT,9)  TPASS(J), (PSMASS(K,J),K=1,NFXOUT)                 EM  1293
      WRITE(LOUT,10) (VFR(K,J),K=1,NOSTAT)                              EM  1294
330   CONTINUE                                                          EM  1295
      WRITE(LOUT,11) (WORK1(K),K=1,NFXOUT)                              EM  1296
C     CONVERT TO MOLE FRACTION-METER OF AIR PARCEL                      EM  1297
      A = 28.97/(ROAIR*PAREA*1.E6)                                      EM  1298
      DO 340 J = 1,KPS                                                  EM  1299
      DO 340 K = 1,NFXOUT                                               EM  1300
      PSMASS(K,J) = A*PSMASS(K,J)                                       EM  1301
340   CONTINUE                                                          EM  1302
      CALL NEWPAG(TITLE,0,JDATE)                                        EM  1303
      WRITE(LOUT,12)  (NAMOUT(K),K=1,NFXOUT)                            EM  1304
      DO 360 J = 1,KPS                                                  EM  1305
      IF(MOD(J,25).NE.0)  GO TO 350                                     EM  1306
      CALL NEWPAG(TITLE,0,JDATE)                                        EM  1307
      WRITE(LOUT,12)  (NAMOUT(K),K=1,NFXOUT)                            EM  1308
350   WRITE(LOUT,9)  TPASS(J),  (PSMASS(K,J),K=1,NFXOUT)                EM  1309
      IF(IPUNCH.NE.IYES)  GO TO 360                                     EM  1310
      WRITE(LPUNCH,14) J, TPASS(J), (PSMASS(K,J),K=1,NFXOUT)            EM  1311
      WRITE(LPUNCH,15) J, (VFR(K,J),K=1,NOSTAT)                         EM  1312
360   CONTINUE                                                          EM  1313
370   RETURN                                                            EM  1314
C                                                                       EM  1315
510   WRITE(LOUT,22)                                                    EM  1316
      KOK = -100                                                        EM  1317
      RETURN                                                            EM  1318
520   WRITE(LOUT,23)                                                    EM  1319
      RETURN                                                            EM  1320
C                                                                       EM  1321
5     FORMAT(47H0POINT SOURCE EMISSION RATES   --   MOLES/HOUR   ,      EM  1322
     1 //,5X,5HTIME ,5X,20HFRACTION    DISTANCE    ,7(5X,A4,3X))        EM  1323
6     FORMAT(1H0,F10.2,F10.4,F11.3,4X,7E12.4)                           EM  1324
7     FORMAT(66H0POINT SOURCE EMISSIONS   --   MOLES AND VERTICAL DISTRIEM  1325
     1BUTION       ,//,8X,7HSPECIES,6X,7(5X,A4,3X))                     EM  1326
8     FORMAT(1H ,7X,10HELEVATIONS,4X,5(F5.0,1H-,F4.0,2X),/,5X,4HTIME)   EM  1327
9     FORMAT(1H0,F10.2,10X,7E12.4)                                      EM  1328
10    FORMAT(1H ,16X,7F12.4)                                            EM  1329
11    FORMAT(1H ,20X,7(2X,10H---------- )//,5X,12HTOTAL MOLES 4X,7E12.4)EM  1330
```

```
12    FORMAT(49HOPOINT SOURCE EMISSIONS  **  MOLE FRACTION*METERS,      EM  1331
     1  //,5X,5HTIME ,11X,7(5X,A4,3X))                                  EM  1332
14    FORMAT(13HPOINT SOURCES,I3,2X,F12.2,4E12.4,/,30X,3E12.4)          EM  1333
15    FORMAT(13X,I3,2X,5E12.4)                                          EM  1334
20    FORMAT(1H ,I5,6F12.4)                                             EM  1335
21    FORMAT(1H ,I5,4F12.4,2I5)                                         EM  1336
22    FORMAT(40HOTPASS ARRAY OVERFLOW  **  JOB ABORTED    )            EM  1337
23    FORMAT(46HOTHERE ARE NO CLOSE ELEVATED POINT SOURCES      )       EM  1338
      END                                                               EM  1339



      SUBROUTINE SEGMET(P)                                              EM  1340
C                                                                       EM  1341
C       SEGMET COMPUTES THE LENGTH AND BISECTOR ANGLES FOR EACH         EM  1342
C       SEGMENT OF THE TRAJECTORY.                                      EM  1343
C       SEGMET COMPUTES THE MAXIMUM RADII AND PERPENDICULAR DISTANCE    EM  1344
C       USED IN SELECTING POINT SOURCES.                                EM  1345
C                                                                       EM  1346
      DIMENSION   P(2,100),  HTH(100)                                   EM  1347
      COMMON /WIND/   T(100),   V(100),   TH(100),   NPTS               EM  1348
      COMMON /ANSWER/ YES,RNEG, IYES,NO,  SMALL                         EM  1349
      COMMON /LINKS/  SEGLEN(100),  RADISQ(100), RADMUL, PDISMX(6)      EM  1350
      COMMON /ANGLES/ GAMA1(100), GAMA2(100), ETA1(100), ETA2(100)      EM  1351
      COMMON /PARCEL/ PWIDTH, HPWIDT, PLENTH, PAREA, DTFREZ, SIGEDG,    EM  1352
     1                ZEE(5), HTINTF(6), NOSTAT,  ROAIR                 EM  1353
      COMMON /SIGMAS/ NDY,XDY(4),SIGYD(4,6),NDZ,XDZ(12),SIGZD(12,6),NCATEM 1354
      DATA PIE05, PIE10, PIE15, PIE20 /1.570796327, 3.141592654,        EM  1355
     1     4.71238898,  6.283185307 /                                   EM  1356
      DATA  RADMUL  /2.5/                                               EM  1357
C                                                                       EM  1358
      DEBUG = RNEG                                                      EM  1359
      VMAX = 0.0000                                                     EM  1360
      HTH(1) = 0.0000                                                   EM  1361
      NM1 = NPTS - 1                                                    EM  1362
      DO 50 I = 1,NM1                                                   EM  1363
      J = I + 1                                                         EM  1364
      DSQ = (P(1,J) - P(1,I))**2   +   (P(2,J) - P(2,I))**2             EM  1365
      RADISQ(I) = DSQ*RADMUL                                            EM  1366
      SEGLEN(I) = SQRT(DSQ)                                             EM  1367
      THET = TH(J) - TH(I)                                              EM  1368
      HTH(J) = 0.50*THET                                                EM  1369
      IF(HTH(J).GT.PIE10)   HTH(J) = HTH(J) - PIE20                     EM  1370
      IF(HTH(J).LT.-PIE10)  HTH(J) = HTH(J) + PIE20                     EM  1371
C   (IE..... -PIE <  HALF THETA  <  PIE  )                              EM  1372
      IF  (V(I).LE.VMAX)  GO TO 50                                      EM  1373
      VMAX = V(I)                                                       EM  1374
50    CONTINUE                                                          EM  1375
C                                                                       EM  1376
      DO 100 I = 1,NM1                                                  EM  1377
      J = I + 1                                                         EM  1378
      GAMA1(I) = PIE05 - HTH(I)                                         EM  1379
      GAMA2(I) = PIE15 - HTH(I)                                         EM  1380
      ETA1(I)  = PIE05 + HTH(J)                                         EM  1381
      ETA2(I)  = PIE15 + HTH(J)                                         EM  1382
```

```
      IF(GAMA1(I).GE.0.0)  GO TO 90                                    EM  1383
      GAMA1(I) = GAMA1(I) + PIE10                                      EM  1384
      GAMA2(I) = GAMA2(I) + PIE10                                      EM  1385
90    IF(ETA1(I).LT.PIE20) GO TO 100                                   EM  1386
      ETA1(I) = ETA1(I) - PIE10                                        EM  1387
      ETA2(I) = ETA2(I) - PIE10                                        EM  1388
100   CONTINUE                                                         EM  1389
C                                                                      EM  1390
      DXMAX = VMAX*DTFREZ                                              EM  1391
      DO 150 I = 1,6                                                   EM  1392
      SIGY = TRPLOG(DXMAX,XDY,SIGYD(1,I),NDY,1)                        EM  1393
      PDISMX(I) = SIGY*SIGEDG                                          EM  1394
      PDISMX(I) = AMAX1(HPWIDT,PDISMX(I))                              EM  1395
150   CONTINUE                                                         EM  1396
C                                                                      EM  1397
      IF(DEBUG.NE.YES) RETURN                                          EM  1398
C                                                                      EM  1399
      WRITE(6,1)                                                       EM  1400
      DEG = 180./PIE10                                                 EM  1401
      DO 200 I = 1,NM1                                                 EM  1402
      A = TH(I)*DEG                                                    EM  1403
      B = HTH(I)*DEG                                                   EM  1404
      C = HTH(I+1)*DEG                                                 EM  1405
      D = GAMA1(I)*DEG                                                 EM  1406
      E = GAMA2(I)*DEG                                                 EM  1407
      F = ETA1(I)*DEG                                                  EM  1408
      G = ETA2(I)*DEG                                                  EM  1409
      WRITE(6,2) I,A,B,C,D,E,F,G                                       EM  1410
200   CONTINUE                                                         EM  1411
      WRITE(6,3) (I,PDISMX(I),I=1,6)                                   EM  1412
      RETURN                                                           EM  1413
1     FORMAT(1H0,12H ANGLES A-G       )                                EM  1414
2     FORMAT(1H ,I5,7F10.2)                                            EM  1415
3     FORMAT(41H0MAXIMUM PERPENDICULAR DISTANCE CRITERIA     ,/        EM  1416
     1  ,37HOSTABILITY CLASS    DISTANCE . (KM)              ,         EM  1417
     2  6(//10X,I1,10X,F6.3))                                         EM  1418
      END                                                              EM  1419


      SUBROUTINE STACKS(TIME,LUP,KOK,IWAS,NDSUMS)                      EM  1420
C                                                                      EM  1421
C          *STACKS* READS THE RAPS POINT SOURCE EMISSIONS FILE         EM  1422
C           AND FILLS THE ARRAYS IN COMMON BLOCK /PSDAT/ EACH HOUR.    EM  1423
C                                                                      EM  1424
C           PSXY     = UTM X & Y COORDINATES (KILOMETERS)              EM  1425
C           PSHITE   = STACK HEIGHTS     (METERS)                      EM  1426
C           PSTEMP   = STACK EFFLUENT TEMPERATURE   (CENTIGRADE)       EM  1427
C           PSFLOW   = STACK EFFLUENT VOLUMETRIC FLOW RATE  (M**3/MIN) EM  1428
C           PSRATE   = EMISSION RATES FOR VARIOUS SPECIES  (MOLES/HR)  EM  1429
C           NPS      = NUMBER OF POINT SOURCES  (FOR THE CURRENT HOUR) EM  1430
C           TIME     = TIME IN MINUTES FROM MIDNIGHT                   EM  1431
C           LUP      = LOGICAL UNIT OF POINT SOURCE EMISSIONS FILE     EM  1432
C           NDSUMS   = REQUEST REGIONAL TOTALS (YES OR NO)             EM  1433
C                                                                      EM  1434
```

```
C          RAPS FILE DESCRIPTION                                                EM  1435
C                                                                               EM  1436
C              IDAT NSTACKS UTMX UTMY HITE DIAM TEMP FLOW PART                  EM  1437
C                           SOX  NOX  THC  CO   NRHC PARF OLEF                  EM  1438
C                           AROM ALDE STACKID                                   EM  1439
C                           UTMX UTMY HITE ..............                       EM  1440
C                           ............................                        EM  1441
C                           ......... STACKID                                   EM  1442
C                           (REPEATED NSTACK TIMES PER HOUR)                    EM  1443
C                                                                               EM  1444
      DIMENSION ISOURC(17,505)                                                  EM  1445
      COMMON /ANSWER/  YES,RNEG,      IYES,NO,     SMALL                        EM  1446
      COMMON /INPUTS/  TITLE(20),     JDATE(10),   NCASE                        EM  1447
      COMMON /ORIGIN/  UTMXOR,        UTMYOR                                    EM  1448
      COMMON /LABLIN/  NAMIN(11),     NFLXIN, WT(11), ADJUST(11)                EM  1449
      COMMON /  WORKER/  IDAT,NSTACK,  SUM(510),  WORK(8585),                   EM  1450
     1                   RSUM(100),     IWORK(13246)                           EM  1451
      COMMON /PSDAT/   PSXY(2,505),  PSRAT(7,505),    PSTEMP(505),             EM  1452
     1                 PSFLOW(505),  PSHITE(505),     NPS                      EM  1453
      EQUIVALENCE (ISOURC,WORK)                                                 EM  1454
      DATA NVAR, NWPREC /17, 510/                                               EM  1455
      DEBUG = RNEG                                                              EM  1456
      NWPRM2 = NWPREC - 2                                                       EM  1457
      INDEX = IFIX(TIME + SMALL)/60                                             EM  1458
      IF(INAS.LE.0) LINDEX = -1                                                 EM  1459
      IF(INDEX.GT.LINDEX) GO TO 50                                              EM  1460
      IF(INDEX.EQ.LINDEX) RETURN                                                EM  1461
      KOK = -8888                                                               EM  1462
      WRITE(6,1)                                                                EM  1463
      RETURN                                                                    EM  1464
50    CONTINUE                                                                  EM  1465
CUNIVAC       READ(LUP,ERR=520,END=510) IDAT,NSTACK,(SUM(I),I=1,NWPRM2)         EM  1466
      BUFFER IN(LUP,1) (IDAT,SUM(NWPRM2))                                       EM  1467
      IF(UNIT(LUP))  60,510,520                                                 EM  1468
60    IYR = IDAT/100000                                                         EM  1469
      IDAY = IDAT/100 - IYR*1000                                                EM  1470
      IHR = IDAT - IYR*100000 - IDAY*100                                        EM  1471
      NWHR = NSTACK*NVAR                                                        EM  1472
      NWLFT = NWHR - NWPRM2                                                     EM  1473
      NBLKS = NWLFT/NWPREC                                                      EM  1474
      IF(MOD(NWLFT,NWPREC).NE.0) NBLKS = NBLKS + 1                              EM  1475
      IF(DEBUG.EQ.YES)  WRITE(6,2) IYR, IDAY, IHR, NSTACK                       EM  1476
C     IF(DEBUG.EQ.YES) WRITE(6,11) (SUM(IJK),IJK=1,NWPRM2)                      EM  1477
11    FORMAT(1H ,100(17F7.3,/,1X))                                              EM  1478
      IF(IHR.EQ.INDEX)  GO TO 100                                               EM  1479
      DO 80 I = 1,NBLKS                                                         EM  1480
      READ(LUP)                                                                 EM  1481
80    CONTINUE                                                                  EM  1482
      GO TO 50                                                                  EM  1483
100   CONTINUE                                                                  EM  1484
      CALL XMIT(NWPRM2,SUM,WORK)                                                EM  1485
      K1 = -1                                                                   EM  1486
      K2 = NWPRM2                                                               EM  1487
      DO 120 I = 1,NBLKS                                                        EM  1488
      K1 = K1 + NWPREC                                                          EM  1489
```

```
CUNIVAC  READ(LUP,ERR=520,END=510)  (SUM(J),J=1,NWPREC)                    EM  1490
      BUFFER IN (LUP,1) (SUM(1), SUM(NWPREC))                             EM  1491
110   IF(UNIT(LUP))     115,   510,   520                                  EM  1492
115   CALL XMIT(NWPREC,SUM,WORK(K1))                                      EM  1493
C     IF(I.EQ.1.AND.DEBUG.EQ.YES)WRITE(6,11)(SUM(IJK),IJK=1,NWPREC)       EM  1494
120   CONTINUE                                                            EM  1495
C                                                                         EM  1496
      DO 150 K = 1,NSTACK                                                 EM  1497
      N = (K-1)*NVAR                                                      EM  1498
      PSXY(1,K) = WORK(1+N) - UTMXOR                                      EM  1499
      PSXY(2,K) = WORK(2+N) - UTMYOR                                      EM  1500
C       NOX,PART,OLEF,ANOM,ALDE,CO,SOX                                    EM  1501
      PSRAT(1,K) = WORK(9+N)*ADJUST(3)                                    EM  1502
      PSRAT(2,K) = WORK(13+N)*ADJUST(8)                                   EM  1503
      PSRAT(3,K) = WORK(14+N)*ADJUST(9)                                   EM  1504
      PSRAT(4,K) = WORK(15+N)*ADJUST(10)                                  EM  1505
      PSRAT(5,K) = WORK(16+N)*ADJUST(11)                                  EM  1506
      PSRAT(6,K) = WORK(11+N)*ADJUST(5)                                   EM  1507
      PSRAT(7,K) = WORK(8+N)*ADJUST(2)                                    EM  1508
      PSHITE(K) = FLOAT(ISOURC(3,K))                                      EM  1509
      PSTEMP(K) = FLOAT(ISOURC(5,K))                                      EM  1510
      PSFLOW(K) = WORK(6+N)                                               EM  1511
150   CONTINUE                                                            EM  1512
C     DO 155 K= 1,5                                                       EM  1513
C     WRITE(6,14) ITEMPR(K), IHITE(K)                                     EM  1514
C14    FORMAT(101X,2I10)                                                  EM  1515
C     WRITE(6,12) PSXY(1,K),PSXY(2,K),(PSRAT(I5,K),I5=1,7),               EM  1516
C    1  PSFLOW(K), PSTEMP(K), PSHITE(K)                                   EM  1517
C12    FORMAT(1H ,12E10.3)                                                EM  1518
C155   CONTINUE                                                           EM  1519
C                                                                         EM  1520
      NPS = NSTACK                                                        EM  1521
      LINDEX = INDEX                                                      EM  1522
      IF(NDSUMS.NE.IYES)     RETURN                                       EM  1523
C                                                                         EM  1524
C       REGIONAL SUMS                                                     EM  1525
C                                                                         EM  1526
C       (NOTE                                                             EM  1527
      NINM1 = NFLXIN + 1                                                  EM  1528
      CALL XMIT(-NINM1,0,0,RSUM)                                          EM  1529
      DO 170 K = 1,NPS                                                    EM  1530
      N = (K-1)*NVAR + 6                                                  EM  1531
      DO 160 J = 1,NINM1                                                  EM  1532
      RSUM(J) = RSUM(J) + WORK(J+N)                                       EM  1533
160   CONTINUE                                                            EM  1534
170   CONTINUE                                                            EM  1535
C                                                                         EM  1536
      DO 180 J = 1,NINM1                                                  EM  1537
      N = J                                                               EM  1538
      IF(J.GE.6)  N = N + 1                                               EM  1539
      RSUM(J) = RSUM(J)*ADJUST(N)                                         EM  1540
180   CONTINUE                                                            EM  1541
C                                                                         EM  1542
      IF(IWAS.GT.0) GO TO 190                                             EM  1543
      CALL NEWPAG(TITLE,0,JDATE)                                          EM  1544
```

```
      WRITE(6,3) (NAMIN(I),I=1,5),(NAMIN(I),I=7,11)                    EM  1545
190   WRITE(6,4) TIME,  (RSUM(I),I=1,NINM1)                            EM  1546
      RETURN                                                           EM  1547
510   KOK = 8                                                          EM  1548
      WRITE(6,5)                                                       EM  1549
      RETURN                                                           EM  1550
520   KOK = 88                                                         EM  1551
      WRITE(6,6)                                                       EM  1552
      RETURN                                                           EM  1553
C                                                                      EM  1554
1     FORMAT(48H0TRAJECTORY NOT CHRONOLOGICAL  **  JOB ABORTED    )    EM  1555
2     FORMAT(12H0    YEAR =  ,I6,7H  DAY = ,I4,8H  HOUR = ,I4,        EM  1556
     1  25H  NO. OF POINT SOURCES =  ,I5)                              EM  1557
3     FORMAT(1H0,35HREGIONAL POINT SOURCE EMISSION SUMS  ,             EM  1558
     1  30X,12H(MOLES/HOUR)  ,//,8H   TIME  ,6X,11(A4,7X))            EM  1559
4     FORMAT(1H0,F7.1,2X,11E11.3)                                      EM  1560
5     FORMAT(50H0UNEXPECTED EOF ENCOUNTERED IN POINT SOURCE FILE       EM  1561
     1       ,30H       **          JOB ABORTED )                     EM  1562
6     FORMAT(50H0PARITY ERROR IN POINT SOURCE FILE - JOB ABORTED  )    EM  1563
      END                                                              EM  1564
```

```
      FUNCTION TRPLOG(X,XS,FS,NXF,NF)                                  EM  1565
C                                                                      EM  1566
C        LOG-LOG INTERPOLATOR FOR TABULAR FUNCTIONS                    EM  1567
C        (ASSUMES LINEAR EXTENSIONS OF END SEGMENTS ON LOG-LOG         EM  1568
C        GRAPH TO CALCULATE BEYOND RANGE OF XS.)                       EM  1569
C                                                                      EM  1570
      DIMENSION XS(NXF),  FS(NF,NXF)                                   EM  1571
C                                                                      EM  1572
      IF(X.GT.0.0)   GO TO 80                                          EM  1573
      WRITE(6,1) X                                                     EM  1574
1     FORMAT(1H0,10HTRPLOG X=   ,E12.4)                                EM  1575
      STOP                                                             EM  1576
80    CONTINUE                                                         EM  1577
      XLOG = ALOG(X)                                                   EM  1578
      NXFM1 = NXF - 1                                                  EM  1579
      IF( XS(NXF) - XS(1) )   100,  120,  120                          EM  1580
100   DO 110 K = 2,NXFM1                                               EM  1581
      IF(X - XS(K))   110,  110,  140                                  EM  1582
110   CONTINUE                                                         EM  1583
      K = NXF                                                          EM  1584
      GO TO 140                                                        EM  1585
120   DO 130 K = 2,NXFM1                                               EM  1586
      IF(XS(K) - X)  130,  130,  140                                   EM  1587
130   CONTINUE                                                         EM  1588
      K = NXF                                                          EM  1589
140   KL = K - 1                                                       EM  1590
C                                                                      EM  1591
      FA = ALOG(FS(1,K))                                               EM  1592
      FB = ALOG(FS(1,KL))                                              EM  1593
      XA = ALOG(XS(K))                                                 EM  1594
      XB = ALOG(XS(KL))                                                EM  1595
      FLOG = FB + (XLOG - XB)*(FA - FB)/(XA - XB)                      EM  1596
```

```
      TRPLOG = EXP(FLOG)                                                EM  1597
      RETURN                                                            EM  1598
      END                                                               EM  1599



      BLOCK DATA                                                        EM  1600
C                                                                       EM  1601
C         TABULAR DATA FOR HORIZONTAL(Y) AND VERTICAL(Z) SIGMAS AS      EM  1602
C         FUNCTIONS OF DOWNWIND DISTANCE FROM TURNERS WORKBOOK.         EM  1603
C                                                                       EM  1604
C         AND TALL STACK SIGMAS FROM ASME.                              EM  1605
C                                                                       EM  1606
C                                                                       EM  1607
      COMMON /SIGMAS/   NDY,  XDY(4),  SIGYD(4,6),                     EM  1608
     1                  NDZ,  XDZ(12), SIGZD(12,6),  NCAT              EM  1609
C                                                                       EM  1610
      COMMON /SIGTAL/ ATALL(6),BTALL(6),CTALL(6),DTALL(6)               EM  1611
C         HORIZONTAL IN KILOMETERS                                      EM  1612
C         VERTICAL IN METERS                                            EM  1613
C                                                                       EM  1614
      DATA NDY,  NDZ,  NCAT  / 4, 12, 6/                               EM  1615
      DATA XDY /.10,  1.00,  10.00,  80.00/                             EM  1616
      DATA SIGYD /                                                      EM  1617
     A            .0270,   .212,   1.570,   9.000,                     EM  1618
     B            .0190,   .157,   1.190,   6.800,                     EM  1619
     C            .0125,   .104,    .840,   5.050,                     EM  1620
     D            .0080,   .068,    .555,   3.350,                     EM  1621
     E            .0060,   .050,    .410,   2.510,                     EM  1622
     F            .0040,   .034,    .273,   1.680/                     EM  1623
C                                                                       EM  1624
      DATA XDZ /  .100,.200,.300,.500,1.00,2.00,3.00,5.00,              EM  1625
     1            10.0,20.0,50.0,100. /                                EM  1626
      DATA SIGZD/                                                       EM  1627
     A           14.,29.5,48.0,105.,450.,1950.,4600.,13562.,           EM  1628
     A           58818.,255084.,1774117.,7694071.,                     EM  1629
     B           10.8,20.3,30.2,51.0,110.,232.,365.,640.,              EM  1630
     B           1350.,2900.,7968.,17117.,                             EM  1631
     C           7.4,13.9,20.1,32.0,61.,116.,169.,267.,                EM  1632
     C           500.,950.,2170.,4000.,                                EM  1633
     D           4.6,8.5,12.1,18.6,31.5,50.0,64.5,89.0,                EM  1634
     D           137.,202.,328.,455.,                                  EM  1635
     E           3.5,6.38,8.8,13.0,21.3,33.7,42.7,56.0,                EM  1636
     E           79.0,110.,155.,183.,                                  EM  1637
     F           2.3,4.05,5.6,8.5,14.0,21.5,26.5,34.0,                 EM  1638
     F           46.5,60.0,79.0,93.0/                                  EM  1639
C                                                                       EM  1640
C                                                                       EM  1641
      DATA ATALL /.40,.36,.32,.32,.31,.31/                              EM  1642
      DATA BTALL /.91,.86,.78,.78,.71,.71/                              EM  1643
      DATA CTALL /.40,.33,.22,.22,.06,.06/                              EM  1644
      DATA DTALL /.91,.86,.78,.78,.71,.71/                              EM  1645
C                                                                       EM  1646
      END                                                               EM  1647
```

3. Chemical-Diffusion Module Listing

```
      PROGRAM KEMOD   (INPUT, OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT,          KM     1
     1                 TAPE1, TAPE3, TAPE4, TAPE10, TAPE11)               KM     2
C                                                                         KM     3
C     KEMOD IS THE DRIVER FOR THE  CHEMICAL/DIFFUSION MODULE.             KM     4
C                                                                         KM     5
C     SUBROUTINES REQUIRED                                                KM     6
C             KEMOD2    DIFFUN    RATES     UPRAT2   UPFLX1    PREDAT   KM     7
C             DIFCOF    SKEDUL    PEDERV    RATEMI   ISTATE    STEADY   KM     8
C             JACOB     SOL282    UPFLX1    MATMUL   PHOTOD    SOLAR    KM     9
C             TEMPR     XMIT      MDATE     SECOND   NEWPAG    MCHAR    KM    10
C             SETPLT    COPLOT    FMAX      SCALE    TIMEX     UNMIXR   KM    11
C             DRIVE     INTERP    TSTEP     COSET    PSET      BLKSOL   KM    12
C             SOL       BLKDEC    DEC       ADJUST   CHECKY    UPSORC   KM    13
C             ITHOUR    PRODUK                                          KM    14
C                                                                         KM    15
C     PERIPHERRALS REQUIRED                                               KM    16
C             TAPE1 = PUNCHED OUTPUT OF SURFACE CONCENTRATIONS            KM    17
C             TAPE3 = INTERMEDIATE I/O FOR PREDAT TO COPY INPUT           KM    18
C             TAPE4 = SCRATCH FILE FOR PLOTTING ROUTINE                   KM    19
C             TAPE5 = INPUT                                               KM    20
C             TAPE6 = OUTPUT                                              KM    21
C             TAPE10 = VERIFICATION OF FLUX SCHEDULE UPDATES              KM    22
C             TAPE11 = VERIFICATION OF KZ SCHEDULE UPDATES                KM    23
C                                                                         KM    24
      REAL INTIM                                                          KM    25
      COMMON/INPUTS/TITLE(20),          IDATE(10),          NCURV       KM    26
      COMMON/FLUXES/FLXIN( 7,200),      FLXTIM(200),        NFLUX       KM    27
      COMMON /PS1/  TPASS(200),         PS(7,5,75),         FRACT(3),   KM    28
     1              NPTSR,              NPSFLX,             LOCPSF(7)   KM    29
      COMMON/EPCOM6/PW(4500)                                              KM    30
      DIMENSION  AF(11),   VFR(5,200),    PTSR(7,200)                    KM    31
      EQUIVALENCE (RNEG,NEG),(YES,IES),(RMOR,MORE)                        KM    32
      EQUIVALENCE (VFR,PW) , (PTSR,PW(1001))                              KM    33
      DATA YES,NEG,MORE,END/3HYES,2HNO,4HMORE,3HEND/                      KM    34
      DATA FRACT /.50,.50,0.0/                                            KM    35
      DATA LIN,  LOUT  /3,  6/                                            KM    36
      TERM = END                                                          KM    37
C                                                                         KM    38
100   CALL PREDAT                                                         KM    39
C                                                                         KM    40
      CALL MDATE(IDATE)                                                   KM    41
      READ(LIN,25) TITLE                                                  KM    42
      READ(LIN,29) GOAHED                                                 KM    43
      READ(LIN,29) FLXUNT                                                 KM    44
C                                                                         KM    45
C        READ AREA SOURCE EMISSION FLUXES                                 KM    46
C                                                                         KM    47
      READ(LIN,24) NUMFLX                                                 KM    48
      DO 170 I=1,201                                                      KM    49
      READ(LIN,30)  A, (AF(N), N=1,NUMFLX)                                KM    50
      IF(A .LT. 0.0)  GO TO 180                                           KM    51
      FLXTIM(I) = A                                                       KM    52
      DO 165 J=1,NUMFLX                                                   KM    53
      FLXIN(J,I) = AF(J)                                                  KM    54
165   CONTINUE                                                            KM    55
```

```
170   CONTINUE                                                          KM    56
      WRITE(LOUT,23)                                                    KM    57
      GO TO 600                                                         KM    58
180   NFLUX = I-1                                                       KM    59
      INTIM = FLXTIM(1)                                                 KM    60
      TSTOP = FLXTIM(NFLUX)                                             KM    61
C                                                                       KM    62
C         READ POINT SOURCE EMISSION FLUXES                             KM    63
C                                                                       KM    64
      NPTSR = 0                                                         KM    65
      READ(LIN,24) NPSFLX                                               KM    66
      IF(NPSFLX.EQ.0)  GO TO 500                                        KM    67
      DO 400 K = 1,201                                                  KM    68
      READ(LIN,30) A, (AF(I),I=1,NPSFLX)                                KM    69
      IF(A.LT.0.0)  GO TO 450                                           KM    70
      TPASS(K) = A                                                      KM    71
      DO 390 J = 1,NPSFLX                                               KM    72
      PTSR(J,K) = AF(J)                                                 KM    73
390   CONTINUE                                                          KM    74
      READ(LIN,36)  (VFR(I,K), I=1,5)                                   KM    75
400   CONTINUE                                                          KM    76
      WRITE(LOUT,37)                                                    KM    77
      GO TO 600                                                         KM    78
450   NPTSR = K - 1                                                     KM    79
C                                                                       KM    80
500   CONTINUE                                                          KM    81
C                                                                       KM    82
      IF (GOAHED.EQ.RNEG) GO TO 600                                     KM    83
C                                                                       KM    84
      CALL KEMOD2 (INTIM,TSTOP,NUMFLX,FLXUNT)                           KM    85
C                                                                       KM    86
600   READ(LIN,25) TERM                                                 KM    87
      IF (TERM.EQ.RMOR) GO TO 100                                       KM    88
      IF (TERM.NE.END) GO TO 600                                        KM    89
      STOP                                                              KM    90
23    FORMAT (1H1,10X,37HTOO MANY AREA SOURCE FLUXES INPUT.             KM    91
     1  25H  ---     JOB ABORTED.      )                               KM    92
24    FORMAT(40X,I10)                                                   KM    93
25    FORMAT (20A4)                                                     KM    94
29    FORMAT (40X,A4)                                                   KM    95
30    FORMAT (20X,F10.2,4E12.4,/,30X,4E12.4)                            KM    96
36    FORMAT (18X,5E12.4)                                               KM    97
37    FORMAT (50H1,TOO MANY POINT SOURCES INPUT  --  JOB ABORTED    )   KM    98
      END                                                               KM    99



      SUBROUTINE ADJUST (Y, NO)                                         KM   100
C-----------------------------------------------------------------------KM   101
C THIS SUBROUTINE ADJUSTS THE Y ARRAY ON REDUCTION OF ORDER.            KM   102
C SEE REFERENCE 1 FOR DETAILS.                                          KM   103
C-----------------------------------------------------------------------KM   104
      INTEGER NO                                                        KM   105
      INTEGER JSTART, KFLAG, L, LMAX, METH, MF, N, NQ, NQINDX           KM   106
      INTEGER I, IBACK, J, JP1, NQM1, NQM2                              KM   107
```

```
      DIMENSION Y(NO, 6)                                                    KM   108
C                                                                           KM   109
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART         KM   110
      COMMON /EPCM10/ TAU(13),EL(13),TQ(5),LMAX,METH,NQ,L,NQINDX            KM   111
C(1                                                                         KM   112
      DATA ONE /1.000/, ZERO /0.000/                                        KM   113
      IF (NQ .EQ. 2) RETURN                                                 KM   114
      NQM1 = NQ - 1                                                         KM   115
      NQM2 = NQ - 2                                                         KM   116
      GO TO (100, 200), METH                                                KM   117
C                                                                           KM   118
 100  DO 110 J = 1,LMAX                                                     KM   119
 110    EL(J) = ZERO                                                        KM   120
      EL(2) = ONE                                                           KM   121
      HSUM = ZERO                                                           KM   122
      DO 130 J = 1,NQM2                                                     KM   123
C CONSTRUCT COEFFICIENTS OF X*(X+XI(1))*...*(X+XI(J)). ******************KM   124
        HSUM = HSUM + TAU(J)                                                KM   125
        XI = HSUM/H                                                         KM   126
        JP1 = J + 1                                                         KM   127
        DO 120 IBACK = 1,JP1                                                KM   128
          I = (J + 3) - IBACK                                               KM   129
 120      EL(I) = EL(I)*XI + EL(I-1)                                        KM   130
 130    CONTINUE                                                            KM   131
C CONSTRUCT COEFFICIENTS OF INTEGRATED POLYNOMIAL. **********************KM   132
      DO 140 J = 2,NQM1                                                     KM   133
C(1                                                                         KM   134
  140   EL(J+1) =  FLOAT(NQ)*EL(J)/ FLOAT(J)                                KM   135
      GO TO 300                                                             KM   136
C                                                                           KM   137
 200  DO 210 J = 1,LMAX                                                     KM   138
 210    EL(J) = ZERO                                                        KM   139
      EL(3) = ONE                                                           KM   140
      HSUM = ZERO                                                           KM   141
      DO 230 J = 1,NQM2                                                     KM   142
C CONSTRUCT COEFFICIENTS OF X*X*(X+XI(1))*...*(X+XI(J)). ****************KM   143
        HSUM = HSUM + TAU(J)                                                KM   144
        XI = HSUM/H                                                         KM   145
        JP1 = J + 1                                                         KM   146
        DO 220 IBACK = 1,JP1                                                KM   147
          I = (J + 4) - IBACK                                               KM   148
 220      EL(I) = EL(I)*XI + EL(I-1)                                        KM   149
 230    CONTINUE                                                            KM   150
C                                                                           KM   151
C SUBTRACT CORRECTION TERMS FROM Y ARRAY. ********************************KM   152
 300  DO 320 J = 3,NQ                                                      KM   153
        DO 310 I = 1,N                                                      KM   154
 310      Y(I,J) = Y(I,J) - Y(I,L)*EL(J)                                    KM   155
 320    CONTINUE                                                            KM   156
      RETURN                                                                KM   157
C----------------------  END OF SUBROUTINE ADJUST  ----------------------KM   158
      END                                                                   KM   159
```

```
      SUBROUTINE BLKDEC (QB,NN,KK,IP,A,C,TL,W,NOGO,BCFLAG)                KM   160
C                                                                         KM   161
C     PERFORMS LU DECOMPOSITION ON THE MATRIX Q                           KM   162
C     WHERE Q IS BLOCK TRIDIAGONAL AS FOLLOWS                             KM   163
C             B(1)  C(1)  0 ..........  0                                 KM   164
C             A(1)  B(2)  C(2) 0.......  0                               KM   165
C     Q =     0     A(2)  B(3)  C(3) 0..0                                 KM   166
C                         .                                               KM   167
C                          .                                              KM   168
C                           .                                             KM   169
C                      A(KK-2)  B(KK-1)  C(KK-1)                          KM   170
C                               A(KK-1)  B(KK)                            KM   171
C                                                                         KM   172
C     THE MATRICES A,B,C ARE SQUARE MATRICES OF ORDER NN AND Q IS OF      KM   173
C     BLOCK-ORDER KK.                                                     KM   174
C     IN OUR PROBLEM, THE MATRICES A(I) AND C(I) ARE SCALED IDENTITY      KM   175
C     MATRICES, HENCE WE ONLY STORE ONE SCALE FACTOR PER MATRIX,          KM   176
C     THUS A AND C ARE VECTORS OF LENGTH (KK-1).                          KM   177
C                                                                         KM   178
C     IP IS A 2-DIMENSIONAL ARRAY OF ORDER (NN*KK) USED FOR STORING PIVOKM   179
C     INFORMATION IN THE LU DECOMPOSITION OF THE DIAGONAL SUBMATRICES B(KM   180
C                                                                         KM   181
C     IN PRACTICE, THE MATRIX QB IS A 3-DIMENSIONAL ARRAY WHICH CONTAINSKM   182
C     ONLY THE DIAGONAL B SUBMATRICES.                                    KM   183
C                                                                         KM   184
C     TL IS A 3-DIMENSIONAL ARRAY USED USED TO STORE THE LOWER TRIANGULAKM   185
C     ELEMENTS OF Q.                                                      KM   186
C                                                                         KM   187
C     W IS A WORK VECTOR OF LENGTH NN.                                    KM   188
C                                                                         KM   189
C     BCFLAG IS A BOUNDARY CONDITION CODE VECTOR - SEE KEMOD2 FOR         KM   190
C     A COMPLETE DISCRIPTION.                                             KM   191
C                                                                         KM   192
C     NOGO = 0 IF SOLUTION IS SUCCESSFUL.                                 KM   193
C     NOGO = 1 IF SINGULAR MATRIX IS ENCOUNTERED.                         KM   194
C                                                                         KM   195
      DIMENSION     QB(NN,NN,KK),A(1),C(1),IP(NN,KK),TL(NN,NN,1),W(1)     KM   196
      INTEGER BCFLAG(1)                                                   KM   197
C                                                                         KM   198
C     BEGIN LU DECOMPOSITION OF MATRIX Q.  THE UPPER TRIANGULAR ELEMENTSKM   199
C     U, WILL BE STORED IN QB. THE LOWER TRIANGULAR ELEMENTS, L, WILL BEKM   200
C     STORED IN TL.                                                       KM   201
C                                                                         KM   202
C     FIND L(I-1) FROM L(I-1)*U(I-1) = A(I-1) AND U(I) FROM               KM   203
C     U(I) = B(I) - L(I-1)*C(I-1).                                        KM   204
C     TO START, WE HAVE U(1) = B(1).                                      KM   205
C                                                                         KM   206
C     CHECK BOUNDARY CONDION CODE (BCFLAG)                                KM   207
      DO 100 I=1,NN                                                       KM   208
      IF(BCFLAG(I).NE.2)  GO TO 100                                       KM   209
C     ADJUST B(1) FOR CONSTAT CONCENTRATION SURFACE BOUNDARY CONDITION    KM   210
      DO 90 J=1,NN                                                        KM   211
      QB(I,J,1) = 0.0                                                     KM   212
   90 CONTINUE                                                            KM   213
      QB(I,I,1) =1.0                                                      KM   214
```

```
  100 CONTINUE                                                          KM   215
C                                                                       KM   216
      NOGO = 0                                                          KM   217
      DO 300 I = 2,KK                                                   KM   218
       IM1 = I - 1                                                      KM   219
       CALL DEC(NN,NN,QB(1,1,IM1),IP(1,IM1),IER)                        KM   220
       IF(IER.NE.0) GO TO 420                                           KM   221
C                                                                       KM   222
C     FIND L(I-1). INVERT U(I-1) BY REPEATED CALLS TO SOL.              KM   223
C                                                                       KM   224
       SF = A(IM1)                                                      KM   225
        DO 200 J = 1,NN                                                 KM   226
         CALL XMIT(-NN,0.,W)                                            KM   227
         W(J) = 1.0                                                     KM   228
         CALL SOL(NN,NN,QB(1,1,IM1),W,IP(1,IM1))                        KM   229
          DO 190 K = 1,NN                                               KM   230
           TL(K,J,IM1) = SF*W(K)                                        KM   231
  190      CONTINUE                                                     KM   232
  200    CONTINUE                                                       KM   233
C                                                                       KM   234
C     FIND U(I).                                                        KM   235
C                                                                       KM   236
       SF = C(IM1)                                                      KM   237
      IF(IM1.NE.1)  GO TO 240                                           KM   238
C                                                                       KM   239
      DO 220 J=1,NN                                                     KM   240
C     DECOUPLE  EQUATIONS FOR CONSTANT CONCENTRATION BCFLAG             KM   241
      IF(BCFLAG(J).EQ.2)  GO TO 220                                     KM   242
      DO 210 K=1,NN                                                     KM   243
      QB(K,J,I) = QB(K,J,I) -SF*TL(K,J,IM1)                             KM   244
  210 CONTINUE                                                          KM   245
  220 CONTINUE                                                          KM   246
      GO TO 300                                                         KM   247
C                                                                       KM   248
  240 CONTINUE                                                          KM   249
        DO 250 J = 1,NN                                                 KM   250
        DO 250 K = 1,NN                                                 KM   251
         QB(K,J,I) = QB(K,J,I) - SF*TL(K,J,IM1)                         KM   252
  250   CONTINUE                                                        KM   253
  300 CONTINUE                                                          KM   254
C                                                                       KM   255
C     PERFORM LU DECOMPOSITION ON U(KK).  THIS IS REQUIRED FOR THE      KM   256
C     BACK SUBSTITUTION LATER ON.                                       KM   257
C                                                                       KM   258
      CALL DEC(NN,NN,QB(1,1,KK),IP(1,KK),IER)                           KM   259
      IM1 = KK                                                          KM   260
      IF(IER.NE.0) GO TO 420                                            KM   261
      RETURN                                                            KM   262
C                                                                       KM   263
C     MATRIX IS SINGULAR.  JOB IS ABORTED.                              KM   264
C                                                                       KM   265
  420 CONTINUE                                                          KM   266
      NOGO = 1                                                          KM   267
      WRITE(6,1)  IM1                                                   KM   268
    1 FORMAT(1H0,5X, *SINGULAR MATRIX NO. *,I2,*.  JOB ABORTED.*)       KM   269
```

```
      RETURN                                                            KM   270
      END                                                               KM   271



      SUBROUTINE BLKSOL(QB,NN,KK,RHS,IP,A,C,TL,W,BCFLAG)                KM   272
C                                                                       KM   273
C     SOLVES A SYSTEM OF LINEAR EQUATIONS Q*X = RHS AFTER               KM   274
C     LU DECOMPOSITION HAS BEEN PERFORMED ON Q AND                      KM   275
C     WHERE Q IS BLOCK TRIDIAGONAL AS FOLLOWS                           KM   276
C             B(1)  C(1)  0 ..........  0                               KM   277
C             A(1)  B(2)  C(2) 0.......  0                             KM   278
C     Q =     0     A(2)  B(3)  C(3) 0..0                               KM   279
C                           .                                           KM   280
C                            .                                          KM   281
C                             .                                         KM   282
C                        A(KK-2)  B(KK-1)  C(KK-1)                      KM   283
C                                 A(KK-1)  B(KK)                        KM   284
C                                                                       KM   285
C     THE MATRICES A,B,C ARE SQUARE MATRICES OF ORDER NN AND Q IS OF    KM   286
C     BLOCK-ORDER KK.  RHS IS THE RIGHT-HAND SIDE OF THE EQUATION AND ISKM   287
C     A VECTOR OF LENGTH (NN*KK).  THE SOLUTION IS RETURNED IN RHS.     KM   288
C     IN OUR PROBLEM, THE MATRICES A(I) AND C(I) ARE SCALED IDENTITY    KM   289
C     MATRICES, HENCE WE ONLY STORE ONE SCALE FACTOR PER MATRIX,        KM   290
C     THUS A AND C ARE VECTORS OF LENGTH (KK-1).                        KM   291
C                                                                       KM   292
C     IP IS A 2-DIMENSIONAL ARRAY OF ORDER (NN*KK) USED FOR STORING PIVOKM   293
C     INFORMATION IN THE LU DECOMPOSITION OF THE DIAGONAL SUBMATRICES B(KM   294
C                                                                       KM   295
C     IN PRACTICE, THE MATRIX QB IS A 3-DIMENSIONAL ARRAY WHICH CONTAINSKM   296
C     THE DIAGONAL B SUBMATRICES.                                       KM   297
C                                                                       KM   298
C     TL IS A 3-DIMENSIONAL ARRAY USED USED TO STORE THE LOWER TRIANGULAKM   299
C     ELEMENTS OF Q.                                                    KM   300
C                                                                       KM   301
C     W IS A WORK VECTOR OF LENGTH NN.                                  KM   302
C                                                                       KM   303
C                                                                       KM   304
C     BCFLAG IS A BOUNDARY CONDITION CODE VECTOR - SEE KEMOD2 FOR       KM   305
C     A COMPLETE DISCRIPTION.                                           KM   306
C                                                                       KM   307
      DIMENSION     QB(NN,NN,KK),A(1),C(1),IP(NN,KK),RHS(NN,KK),        KM   308
     1              TL(NN,NN,1),W(1)                                    KM   309
      INTEGER BCFLAG(1)                                                 KM   310
C                                                                       KM   311
C                                                                       KM   312
C     FRONT SUBSTITUTION.                                               KM   313
C     INITIALLY, Y(1) = RHS(1). THE RHS VECTOR IS DESTROYED IN THIS     KM   314
C     OPERATION.                                                        KM   315
C                                                                       KM   316
      DO 350 I = 2,KK                                                   KM   317
       IM1 = I - 1                                                      KM   318
       CALL MATMUL(TL(1,1,IM1),RHS(1,IM1),W,NN,NN,1)                    KM   319
        DO 340 J = 1,NN                                                 KM   320
         RHS(J,I) = RHS(J,I) - W(J)                                     KM   321
```

```
  340   CONTINUE                                                          KM   322
  350 CONTINUE                                                            KM   323
C                                                                         KM   324
C     BACK SUBSTITUTION.                                                  KM   325
C     THE VECTOR RHS PREVIOUSLY COMPUTED IN LOOP 350 IS ONCE AGAIN THE    KM   326
C     RIGHT-HAND SIDE.  THE SOLUTION VECTOR IS AGAIN STORED IN RHS.       KM   327
C     RECALL THAT THE U MATRICES IN QB HAVE ALREADY UNDERGONE LU DECOMPO  KM   328
C     AND THE PIVOT INFORMATION IS STORED IN IP.                          KM   329
C                                                                         KM   330
      CALL SOL(NN,NN,QB(1,1,KK),RHS(1,KK),IP(1,KK))                      KM   331
      N = KK - 2                                                          KM   332
      DO 400 I = 1,N                                                      KM   333
       K = KK - I                                                         KM   334
       J = K + 1                                                          KM   335
       SF = C(K)                                                          KM   336
        DO 390 M = 1,NN                                                   KM   337
         RHS(M,K) = RHS(M,K) - SF*RHS(M,J)                                KM   338
  390   CONTINUE                                                          KM   339
       CALL SOL(NN,NN,QB(1,1,K),RHS(1,K),IP(1,K))                         KM   340
  400 CONTINUE                                                            KM   341
C                                                                         KM   342
C     SOLVE FOR RHS(1,1) WITH BOUNDARY CONDITION CHECK                    KM   343
      SF = C(1)                                                           KM   344
      DO 410 M = 1,NN                                                     KM   345
C     DECOUPLE EQUATIONS WITH CONSTANT CONCENTRATION BOUNDARY CONDITION   KM   346
      IF(BCFLAG(M).EQ.2)  GO TO 410                                       KM   347
      RHS(M,1) = RHS(M,1) - SF*RHS(M,2)                                   KM   348
  410 CONTINUE                                                            KM   349
      CALL SOL(NN,NN,QB(1,1,1),RHS(1,1),IP(1,1) )                         KM   350
       RETURN                                                             KM   351
C                                                                         KM   352
      END                                                                 KM   353
```

```
      SUBROUTINE CHECKY(T,Y,Y0,N,KOK)                                     KM   354
C                                                                         KM   355
C       CHECKY CHECKS FOR NEGATIVE CONCENTRATIONS                         KM   356
C       AND DEPLETION OF NITRIC OXIDE.                                    KM   357
C                                                                         KM   358
      DIMENSION IND(5),  Y(1),  Y0(1)                                     KM   359
      COMMON/CHEM1/ NOSTAT,   NOSTM1,   NOREAC                            KM   360
     1              NOSPEC,   NSTEDY,   NK                               KM   361
      DATA YMIN, YNOMIN,  KMAX  /1.E-18,1.E-6, 5/                         KM   362
      KNTER = 0                                                           KM   363
      KSPEC = 1                                                           KM   364
C         CHECK FOR NITRIC OXIDE DEPLETION                                KM   365
      DO 10 K = 1, NOSTAT                                                 KM   366
      KS = (K - 1)* NK + 1                                                KM   367
      IF(Y(KS) .GT. YNOMIN)  GO TO 10                                     KM   368
      KNTER = KNTER + 1                                                   KM   369
      Y(KS) = Y0(KS)                                                      KM   370
C     WRITE(8,1) T, KSPEC, Y(KS), Y0(KS)                                  KM   371
10    CONTINUE                                                            KM   372
C         CHECK FOR Y VALUES LESS THAN YMIN                               KM   373
```

```
      DO 30 K = 1,N                                                    KM   374
      IF(Y(K).GT.YMIN)  GO TO 30                                       KM   375
      KNTER = KNTER + 1                                                KM   376
      KSPEC = MOD(K,NK)                                                KM   377
C     WRITE(8,1) T,KSPEC, Y(K), Y0(K)                                  KM   378
      Y(K) = AMAX1(Y(K),Y0(K))                                         KM   379
30    CONTINUE                                                         KM   380
      IF(KNTER .GT. 5)  KNK = -100                                     KM   381
      RETURN                                                           KM   382
1     FORMAT(1H 5X,6HTIME =,F8.2,13H   SPEC NO. =   ,I3,3X,            KM   383
     1 20HOLD CONCENTRATION =,E13.4,3X,22HRESET CONCENTRATION =,E13.4)  KM   384
      END                                                              KM   385


      SUBROUTINE COPLOT(NDPT)                                          KM   386
C                                                                      KM   387
C         COPLOT GENERATES A PRINTER-PLOT OF GROUND CONCENTRATIONS      KM   388
C         VERSUS TIME FOR UP TO FIVE SPECIES                           KM   389
C                                                                      KM   390
      COMMON/PRPLOT/     KPS( 5),       KSYM( 5),      GRCON(5,100),  KM   391
     1                   TOUT(100),     VALMAX(5)                     KM   392
      COMMON /INPUTS/  TITLE(20),       IDATE(10),     NCURV          KM   393
      DIMENSION KNAM(5)                                                KM   394
      DATA KNAM /1HN,1H2,1HO,1HS,1H4/                                  KM   395
      DATA ZPLUS/1.E-30/                                               KM   396
      DATA LOUT /6/                                                    KM   397
C                                                                      KM   398
      GO TO 30                                                         KM   399
C                                                                      KM   400
      ENTRY IPLOT                                                      KM   401
      CALL XMIT(NCURV,KNAM,KSYM)                                       KM   402
      CALL XMIT(-NCURV,0.,VALMAX)                                      KM   403
      RETURN                                                           KM   404
C                                                                      KM   405
   30 CONTINUE                                                         KM   406
C     T1 = TOUT(1) + ZPLUS                                             KM   407
C     CALL SCALE(T1,TOUT(NDPT),X0,XR)                                  KM   408
C     -----  FIX HORIZONTAL SCALE FOR ALL DAY  -----                   KM   409
      IF(TOUT(NDPT)-TOUT(1) .GT. 750.)  GO TO 35                       KM   410
      X0 = TOUT(1)                                                     KM   411
      XR = X0 + 750.                                                   KM   412
   35 CONTINUE                                                         KM   413
      CALL FMAX(VALMAX,NCURV,A)                                        KM   414
      CALL SCALE(ZPLUS,A,Y8,YT)                                        KM   415
      IF(2.*A.LT.YT)  YT = .5*YT                                       KM   416
      CALL SETPLT(X0,XR,Y8,YT)                                         KM   417
      DO 50 I=1,NDPT                                                   KM   418
      DO 40 K=1,NCURV                                                  KM   419
      CALL PLTPNT(TOUT(I),GRCON(K,I),KSYM(K))                          KM   420
   40 CONTINUE                                                         KM   421
   50 CONTINUE                                                         KM   422
      CALL PLTOUT(LOUT)                                                KM   423
      WRITE(LOUT,1000)                                                 KM   424
      WRITE(LOUT,1001) TITLE                                           KM   425
```

```
      RETURN                                                            KM   426
C                                                                       KM   427
1000  FORMAT(1H ,60X,16HTIME  (MINUTES)       )                         KM   428
1001  FORMAT(1H0,39X,40HLEGEND                                          KM   429
     1   20H S = S02   C = .1*CO         //,30X,20A4  )                 KM   430
      END                                                               KM   431



      SUBROUTINE CDSET                                                  KM   432
C-----------------------------------------------------------------------KM   433
C COSET IS CALLED BY ISTEP AND SETS COEFFICIENTS FOR USE THERE.         KM   434
C-----------------------------------------------------------------------KM   435
      INTEGER JSTART, KFLAG, L, LMAX, METH, MF, N, NQ, NQINDX           KM   436
      INTEGER I, IBACK, J, JP1, JSTART, KFLAG, L, LMAX, MAXDER,         KM   437
     1        METH, MF,N, NQ, NQINDX, NQM1                              KM   438
      DIMENSION EM(13)                                                  KM   439
C                                                                       KM   440
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART     KM   441
      COMMON /EPCM10/ TAU(13),EL(13),TQ(5),LMAX,METH,NQ,L,NQINDX        KM   442
C(2                                                                     KM   443
      DATA CORTES /0,100/                                               KM   444
      DATA ONE  /1.0E0/, SIX /6.0E0/, TWO /2.0E0/, ZERO /0.0E0/         KM   445
      AHDSS = ONE                                                       KM   446
C(2                                                                     KM   447
      IF (SS .NE. ZERO) AHDSS =  ABS(H)/SS                              KM   448
      FLOTL =  FLOAT(L)                                                 KM   449
      NQM1 = NQ - 1                                                     KM   450
      GO TO (1, 2), METH                                                KM   451
 1    MAXDER = 12                                                       KM   452
      GO TO 100                                                         KM   453
C                                                                       KM   454
 2    MAXDER = 5                                                        KM   455
      GO TO 200                                                         KM   456
C                                                                       KM   457
 100  IF (NQ .NE. 1) GO TO 110                                          KM   458
      EL(1) = ONE                                                       KM   459
      EL(2) = ONE                                                       KM   460
      TQ(1) = ONE                                                       KM   461
      TQ(2) = TWO*AHDSS                                                 KM   462
      TQ(3) = SIX*TQ(2)                                                 KM   463
      TQ(5) = ONE                                                       KM   464
      GO TO 300                                                         KM   465
 110  HSUM = H                                                          KM   466
      EM(1) = ONE                                                       KM   467
      FLOTNQ = FLOTL + ONE                                              KM   468
      DO 115 I = 2,L                                                    KM   469
 115    EM(I) = ZERO                                                    KM   470
      DO 150 J = 1,NQM1                                                 KM   471
        IF ((J .NE. NQM1) .OR. (NQINDX .NE. 1)) GO TO 130               KM   472
        S = ONE                                                         KM   473
        CSUM = ZERO                                                     KM   474
        DO 120 I = 1,NQM1                                               KM   475
C(1                                                                     KM   476
          CSUM = CSUM + S*EM(I)/ FLOAT(I+1)                             KM   477
```

```
 120      S = -S                                                          KM   478
        TQ(1) = AHDSS*EM(NQM1)/(FLOTNQ*CSUM)                             KM   479
 130    RXI = H/HSUM                                                      KM   480
        DO 140 IBACK = 1,J                                               KM   481
          I = (J + 2) - IBACK                                            KM   482
 140      EM(I) = EM(I) + EM(I-1)*RXI                                     KM   483
 150    HSUM = HSUM + TAU(J)                                              KM   484
C COMPUTE INTEGRAL FROM -1 TO 0 OF POLYNOMIAL AND OF X TIMES IT. -------KM   485
      S = ONE                                                            KM   486
      EM0 = ZERO                                                         KM   487
      CSUM = ZERO                                                        KM   488
      DO 160 I = 1,NQ                                                   KM   489
C(1                                                                      KM   490
        FLOTI =  FLOAT(I)                                                KM   491
        EM0 = EM0 + S*EM(I)/FLOTI                                        KM   492
        CSUM = CSUM + S*EM(I)/(FLOTI+1)                                  KM   493
 160    S = -S                                                           KM   494
C IN EL, FORM COEFFICIENTS OF NORMALIZED INTEGRATED POLYNOMIAL. --------KM   495
      S = ONE/EM0                                                        KM   496
      EL(1) = ONE                                                        KM   497
      DO 170 I = 1,NQ                                                   KM   498
C(1                                                                      KM   499
  170   EL(I+1) = S*EM(I)/ FLOAT(I)                                      KM   500
      XI = HSUM/H                                                        KM   501
      TQ(2) = AHDSS*XI*EM0/CSUM                                          KM   502
      TQ(5) = XI/EL(L)                                                   KM   503
      IF (NQINDX .NE. 1) GO TO 300                                       KM   504
C FOR HIGHER ORDER CONTROL CONSTANT, MULTIPLY POLYNOMIAL BY 1+X/XI(Q). -KM   505
      RXI = ONE/XI                                                       KM   506
      DO 180 IBACK = 1,NQ                                               KM   507
        I = (L + 1) - IBACK                                              KM   508
 180    EM(I) = EM(I) + EM(I-1)*RXI                                      KM   509
C COMPUTE INTEGRAL OF POLYNOMIAL. --------------------------------------KM   510
      S = ONE                                                            KM   511
      CSUM = ZERO                                                        KM   512
      DO 190 I = 1,L                                                     KM   513
C(1                                                                      KM   514
      CSUM = CSUM + S*EM(I)/ FLOAT(I+1)                                  KM   515
 190    S = -S                                                           KM   516
      TQ(3) = AHDSS*FLOTL*EM0/CSUM                                       KM   517
      GO TO 300                                                          KM   518
C                                                                        KM   519
 200  DO 210 I = 3,L                                                     KM   520
 210    EL(I) = ZERO                                                     KM   521
      EL(1) = ONE                                                        KM   522
      EL(2) = ONE                                                        KM   523
      HSUM = H                                                           KM   524
      HSUM1 = ZERO                                                       KM   525
      PROD = ONE                                                         KM   526
      RXI = ONE                                                          KM   527
      IF (NQ .EQ. 1) GO TO 240                                           KM   528
      DO 230 J = 1,NQM1                                                  KM   529
C IN EL, CONSTRUCT COEFFICIENTS OF (1+X/XI(1))*...*(1+X/XI(J+1)). ------KM   530
        HSUM = HSUM + TAU(J)                                             KM   531
        HSUM1 = HSUM1 + TAU(J)                                           KM   532
```

```
        PROD = PROD*(HSUM/HSUM1)                                        KM   533
        RXI = H/HSUM                                                    KM   534
        JP1 = J + 1                                                     KM   535
        DO 220 IBACK = 1,JP1                                            KM   536
          I = (J + 3) - IBACK                                           KM   537
 220      EL(I) = EL(I) + EL(I-1)*RXI                                   KM   538
 230    CONTINUE                                                        KM   539
 240  TQ(2) = AHDSS*EL(2)*(ONE + PROD)                                  KM   540
      TQ(5) = (ONE + PROD)/EL(L)                                        KM   541
      IF (NQINDX .NE. 1) GO TO 300                                      KM   542
      CNQM1 = RXI/EL(L)                                                 KM   543
      ELP = EL(2) - RXI                                                 KM   544
      TQ(1) = AHDSS*ELP/CNQM1                                           KM   545
      HSUM = HSUM + TAU(NQ)                                             KM   546
      RXI = H/HSUM                                                      KM   547
      ELP = EL(2) + RXI                                                 KM   548
      TQ(3) = AHDSS*ELP*RXI*(ONE + PROD)*(FLOTL + ONE)                  KM   549
 300  TQ(4) = CORTES*TQ(2)                                              KM   550
      LMAX = MAXDER + 1                                                 KM   551
      RETURN                                                            KM   552
C---------------------  END OF SUBROUTINE COSET  ---------------------KM   553
      END                                                               KM   554


      SUBROUTINE DEC (N, NDIM, A, IP, IER)                              KM   555
C-----------------------------------------------------------------------KM   556
C MATRIX TRIANGULARIZATION BY GAUSSIAN ELIMINATION.                     KM   557
C INPUT..                                                               KM   558
C    N = ORDER OF MATRIX.                                               KM   559
C    NDIM = DECLARED DIMENSION OF ARRAY  A .                            KM   560
C    A = MATRIX TO BE TRIANGULARIZED.                                   KM   561
C OUTPUT..                                                              KM   562
C    A(I,J), I.LE.J = UPPER TRIANGULAR FACTOR, U .                      KM   563
C    A(I,J), I.GT.J = MULTIPLIERS = LOWER TRIANGULAR FACTOR, I - L.     KM   564
C    IP(K), K.LT.N = INDEX OF K-TH PIVOT ROW.                           KM   565
C    IP(N) = (-1)**(NUMBER OF INTERCHANGES) OR 0 .                      KM   566
C    IER = 0 IF A NONSINGULAR, OR K IF A FOUND TO BE                    KM   567
C          SINGULAR AT STAGE K.                                         KM   568
C USE  SOL  TO OBTAIN SOLUTION OF LINEAR SYSTEM.                        KM   569
C DETERM(A) = IP(N)*A(1,1)*A(2,2)*...*A(N,N).                           KM   570
C IF IP(N)=0, A IS SINGULAR, SOL WILL DIVIDE BY ZERO.                   KM   571
C INTERCHANGES FINISHED IN U , ONLY PARTLY IN L .                       KM   572
C                                                                       KM   573
C REFERENCE..                                                           KM   574
C C. B. MOLER, ALGORITHM 423, LINEAR EQUATION SOLVER,                   KM   575
C COMM. ASSOC. COMPUT. MACH., 15 (1972), P. 274.                        KM   576
C-----------------------------------------------------------------------KM   577
      INTEGER IER, IP, N, NDIM                                          KM   578
      INTEGER I, J, K, KP1, M, NM1                                      KM   579
      DIMENSION A(NDIM,N),IP(N)                                         KM   580
C(1                                                                     KM   581
      DATA ONE /1.000/, ZERO /0.000/                                    KM   582
      IER = 0                                                           KM   583
      IP(N) = 1                                                         KM   584
```

```
      IF (N .EQ. 1) GO TO 70                                                KM   585
      NM1 = N - 1                                                           KM   586
      DO 60 K = 1,NM1                                                       KM   587
        KP1 = K + 1                                                         KM   588
        M = K                                                               KM   589
        DO 10 I = KP1,N                                                     KM   590
 10       IF ( ABS(A(I,K)) .GT.  ABS(A(M,K))) M = I                        KM   591
        IP(K) = M                                                           KM   592
        T = A(M,K)                                                          KM   593
        IF (M .EQ. K) GO TO 20                                              KM   594
        IP(N) = -IP(N)                                                      KM   595
        A(M,K) = A(K,K)                                                     KM   596
        A(K,K) = T                                                          KM   597
 20     IF (T .EQ. ZERO) GO TO 80                                           KM   598
        T = ONE/T                                                           KM   599
        DO 30 I = KP1,N                                                     KM   600
 30       A(I,K) = -A(I,K)*T                                               KM   601
        DO 50 J = KP1,N                                                     KM   602
          T = A(M,J)                                                        KM   603
          A(M,J) = A(K,J)                                                   KM   604
          A(K,J) = T                                                        KM   605
          IF (T .EQ. ZERO) GO TO 50                                         KM   606
          DO 40 I = KP1,N                                                   KM   607
 40         A(I,J) = A(I,J) + A(I,K)*T                                     KM   608
 50       CONTINUE                                                         KM   609
 60     CONTINUE                                                           KM   610
 70   K = N                                                                KM   611
      IF (A(N,N) .EQ. ZERO) GO TO 80                                        KM   612
      RETURN                                                                KM   613
 80   IER = K                                                              KM   614
      IP(N) = 0                                                             KM   615
      RETURN                                                                KM   616
C---------------------  END OF SUBROUTINE DEC  ----------------------------KM   617
      END                                                                   KM   618
```

```
      SUBROUTINE DIFCOF(NOSTAT)                                             KM   619
C                                                                           KM   620
C          THIS SUBROUTINE CALCULATES THE SCALED DIFFUSIVITY                KM   621
C          COEFFICIENTS FROM DFINIT                                         KM   622
C          VARIABLE MESH VERSION CODE 7.6.77                                KM   623
C                                                                           KM   624
      INTEGER BCFLAG                                                        KM   625
      COMMON/CHEM3/ ZEE(5),              DELZ(4),          HTCELL(6),       KM   626
     1              TDELZ(2) ,           DFINIT(6),        SCALOW(4),       KM   627
     2              DCOF(5),             FLXWAL(40),       FLXDGE(40),      KM   628
     3              BCFLAG(40),          DPRATE(10),       DEPOWR(10),      KM   629
     4              LOCOPF(10),          NDPFLX ,          SCALUP(4)        KM   630
C                                                                           KM   631
C          COEFFICIENTS FOR GROUND AND EDGE EQUATIONS                       KM   632
      I=1                                                                   KM   633
      DCOF(I) =(DFINIT(I)+DFINIT(I+1))/HTCELL(I)/HTCELL(I)                  KM   634
      SCALUP(I) = DCOF(I)                                                   KM   635
      I=NOSTAT                                                              KM   636
```

```
      DCOF(I) =(DFINIT(I)+DFINIT(I+1))/HTCELL(I)/HTCELL(I)              KM   637
      SCALOW(I-1) = DCOF(I)                                             KM   638
C                                                                       KM   639
C          COEFFICIENTS FOR INTERMIDIATE EQUATIONS                      KM   640
C                                                                       KM   641
      NOSTM1 = NOSTAT - 1                                               KM   642
      DO 100 I=2,NOSTM1                                                 KM   643
      SCALOW(I-1) = DFINIT(I)/DELZ(I-1)/HTCELL(I)                       KM   644
      SCALUP(I) = DFINIT(I+1)/DELZ(I)/HTCELL(I)                         KM   645
      DCOF(I) = SCALOW(I-1) + SCALUP(I)                                 KM   646
  100 CONTINUE                                                          KM   647
      RETURN                                                            KM   648
      END                                                               KM   649



      SUBROUTINE DIFFUN (N,T,Y,YDOT)                                    KM   650
C                                                                       KM   651
C     ------       DIFFUN DEFINES THE DIFFERENTIAL EQUATIONS  ----------KM   652
C                                                                       KM   653
C                          YDOT = F(Y,T)                                KM   654
C                                                                       KM   655
      INTEGER BCFLAG                                                    KM   656
      COMMON/CHEM1/ NUSTAT,            NOSTM1,          NOREAC,         KM   657
     1              NOSPEC,            NSTDY,           NK              KM   658
      COMMON/CHEM2/ CONIN(40,5),       WTMOLE(40),      RATKON(55),     KM   659
     1              RATEFF(55),        RATEV(2,5),      QRATE,          KM   660
     2              NVRATE,            LOCVRT(2)                        KM   661
      COMMON/CHEM3/ ZEE(5),            DELZ(4),         HTCELL(6),      KM   662
     1              TDELZ(2) ,         DFINIT(6),       SCALOW(4),      KM   663
     2              DCOF(5),           FLXWAL(40),      FLXDGE(40),     KM   664
     3              BCFLAG(40),        DPRATE(10),      DEPOWR(10),     KM   665
     4              LOCDPF(10),        NDPFLX ,         SCALUP(4)       KM   666
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT   KM   667
      COMMON /CHEM5/  REACT(20,55),SPEC(40),LOCFLX(10),                 KM   668
     1               NASFLX,FLXW1(30),FLXW2(30)                         KM   669
      COMMON /PS1/   TPASS(200), PS(7,5,75), FRACT(3),                  KM   670
     1              NPTSR,NPSFLX,LOCPSF(7)                              KM   671
      DIMENSION  Y(N),     YDOT(N),     SINK(30)                       KM   672
C                                                                       KM   673
      CALL XMIT (-NK, 0,0, SINK)                                        KM   674
C                                                                       KM   675
      CALL RATES(Y,YDOT)                                                KM   676
C                                                                       KM   677
      IF(NDPFLX.LT.1)  GO TO 50                                         KM   678
C     ----    UPDATE FLUXES OF DEPOSITING SPECIES                       KM   679
      DO 40 K=1,NDPFLX                                                  KM   680
      ID = LOCDPF(K)                                                    KM   681
      IF(Y(ID).LT.1.E-16)  GO TO 40                                     KM   682
      SINK(ID) = - DPRATE(K)*(Y(ID)**DEPOWR(K))                         KM   683
   40 CONTINUE                                                          KM   684
   50 CONTINUE                                                          KM   685
C                                                                       KM   686
C       INTERPOLATE EMISSION FLUXES TO THE CURRENT TIME                 KM   687
C                                                                       KM   688
```

```
      FDELT = (T - TLAST)/UPDINT                                        KM   689
      DO 60 I = 1,NASFLX                                                KM   690
      K = LOCFLX(I)                                                     KM   691
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                  KM   692
60    CONTINUE                                                          KM   693
      K = 7                                                             KM   694
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                  KM   695
      K = 2                                                             KM   696
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                  KM   697
      DO 75 J =1,NOSTAT                                                 KM   698
      DO 70 I =1,NPSFLX                                                 KM   699
      K =LOCPSF(I)                                                      KM   700
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT             KM   701
70    CONTINUE                                                          KM   702
      K = 2                                                             KM   703
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT             KM   704
      K = 7                                                             KM   705
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT             KM   706
75    CONTINUE                                                          KM   707
C                                                                       KM   708
      NM1 = NK*NOSTM1                                                   KM   709
      NM2 = NK*(NOSTAT-2)                                               KM   710
C     ------- EQUATIONS FOR SURFACE AND TOP EDGE MESH POINTS            KM   711
      DO 100  I=1,NK                                                    KM   712
      IL=NM1+I                                                          KM   713
      YDOT(I) = DCOF(1)*(Y(I+NK)-Y(I)) + YDOT(I)                        KM   714
     1                  +  TDELZ(1)*(FLXWAL(I)+SINK(I)) + PSRATE(I,1)  KM   715
      IF(BCFLAG(I) .EQ. 2)  YDOT(I) = 0.0                               KM   716
      YDOT(IL) = DCOF(NOSTAT)*(Y(I+NM2)-Y(IL)) + YDOT(IL)               KM   717
     1                        +  PSRATE(I,NOSTAT)                      KM   718
  100 CONTINUE                                                          KM   719
C     ------- EQUATIONS FOR INTERMEDIATE MESH POINTS                    KM   720
      DO 200 J=2,NOSTM1                                                 KM   721
      NM1= (J-2)*NK                                                     KM   722
      NN = (J-1)*NK                                                     KM   723
      NP1 = J*NK                                                        KM   724
      JM1 = J- 1                                                        KM   725
      DO 150 I = 1,NK                                                   KM   726
      LN = I + NN                                                       KM   727
      YDOT(LN) = SCALOW(JM1)*Y(I+NM1) - DCOF(J)*Y(LN) +                 KM   728
     1           SCALUP(J)*Y(I+NP1) + YDOT(LN) + PSRATE(I,J)            KM   729
  150 CONTINUE                                                          KM   730
  200 CONTINUE                                                          KM   731
      RETURN                                                            KM   732
      END                                                               KM   733
```

```
      SUBROUTINE DRIVE(N,T0,H0,Y0,TOUT,EPS,IERROR,MF,INDEX,BIGSTP,KOK)  KM   734
C                                                                       KM   735
C-----------------------------------------------------------------------KM   736
C     THIS IS A SPECIAL VERSION OF EPISODE  (ERT CODE DATE 7.8.77)      KM   737
C     FOR SOLVING A BLOCK TRI-DIAGONAL SYSTEM OF ODE*S FOR KEMOD2.      KM   738
C     THIS VERSION OF EPISODE SHOULD ONLY BE USED WITH MF = 21          KM   739
C     MODIFICATIONS BY F.W. LURMANN   (7.8.77).                         KM   740
C-----------------------------------------------------------------------KM   741
C                                                                       KM   742
C THIS IS BASICALLY THE JUNE 24, 1975 VERSION OF                        KM   743
C EPISODE..  EXPERIMENTAL PACKAGE FOR INTEGRATION OF                    KM   744
C SYSTEMS OF ORDINARY DIFFERENTIAL EQUATIONS,                           KM   745
C    DY/DT = F(Y,T),  Y = (Y(1),Y(2),...,Y(N)) TRANSPOSE,               KM   746
C GIVEN THE INITIAL VALUE OF Y.                                         KM   747
C                  REFERENCES                                           KM   748
C 1.  G. D. BYRNE AND A. C. HINDMARSH, A POLYALGORITHM FOR THE          KM   749
C       NUMERICAL SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS,          KM   750
C       UCRL-75652, LAWRENCE LIVERMORE LABORATORY, P. O. BOX 808,       KM   751
C       LIVERMORE, CA 94550, APRIL 1974. ALSO IN ACM TRANSACTIONS       KM   752
C       ON MATHEMATICAL SOFTWARE, 1 (1975), PP. 71-96.                  KM   753
C                                                                       KM   754
C 2.  A. C. HINDMARSH AND G. D. BYRNE, EPISODE., AN EXPERIMENTAL        KM   755
C       PACKAGE FOR THE INTEGRATION OF SYSTEMS OF ORDINARY              KM   756
C       DIFFERENTIAL EQUATIONS, UCID-30112, L.L.L., MAY, 1975.          KM   757
C                                                                       KM   758
C 3.  A. C. HINDMARSH, GEAR., ORDINARY DIFFERENTIAL EQUATION            KM   759
C       SYSTEM SOLVER, UCID-30001, REV. 3, L.L.L., DECEMBER, 1974.      KM   760
C                                                                       KM   761
C-----------------------------------------------------------------------KM   762
C DRIVE IS A DRIVER SUBROUTINE FOR THE EPISODE PACKAGE.                 KM   763
C DRIVE IS TO BE CALLED ONCE FOR EACH OUTPUT VALUE OF T,                KM   764
C IT THEN MAKES REPEATED CALLS TO THE CORE INTEGRATOR                   KM   765
C SUBROUTINE, TSTEP.                                                    KM   766
C                                                                       KM   767
C THE INPUT PARAMETERS ARE AS FOLLOWS.                                  KM   768
C   N      =  THE NUMBER OF DIFFERENTIAL EQUATIONS (USED ONLY ON        KM   769
C               FIRST CALL, UNLESS INDEX = -1).  N MUST NEVER BE        KM   770
C               INCREASED DURING A GIVEN PROBLEM.                       KM   771
C   T0     =  THE INITIAL VALUE OF T, THE INDEPENDENT VARIABLE          KM   772
C               (USED FOR INPUT ONLY ON FIRST CALL).                    KM   773
C   H0     =  THE STEP SIZE IN T (USED FOR INPUT ONLY ON THE            KM   774
C               FIRST CALL, UNLESS INDEX = 3 ON INPUT).  WHEN           KM   775
C               INDEX = 3, H0 IS THE MAXIMUM ABSOLUTE VALUE OF          KM   776
C               THE STEP SIZE TO BE USED.                               KM   777
C   Y0     =  A VECTOR OF LENGTH N CONTAINING THE INITIAL VALUES OF     KM   778
C               Y (USED FOR INPUT ONLY ON FIRST CALL).                  KM   779
C   TOUT   =  THE VALUE OF T AT WHICH OUTPUT IS DESIRED NEXT.           KM   780
C               INTEGRATION WILL NORMALLY GO BEYOND TOUT AND            KM   781
C               INTERPOLATE TO T = TOUT.  (USED ONLY FOR INPUT.)        KM   782
C   EPS    =  THE RELATIVE ERROR BOUND (USED ONLY ON FIRST CALL,        KM   783
C               UNLESS INDEX = -1).  THIS BOUND IS USED AS FOLLOWS.     KM   784
C               LET R(I) DENOTE THE ESTIMATED RELATIVE LOCAL ERROR      KM   785
C               IN Y(I), I.E. THE ERROR RELATIVE TO YMAX(I), AS         KM   786
C               MEASURED PER STEP (OF SIZE H) OR PER SS UNITS OF T.     KM   787
C               THEN EPS IS A BOUND ON THE ROOT-MEAN-SQUARE NORM        KM   788
```

```
C               OF THE VECTOR R, I.E.                                   KM   789
C                     N                                                 KM   790
C               SQRT ( SUM ( R(I)**2 )/N ) .LT. EPS,                    KM   791
C                    I=1                                                KM   792
C               THE VECTOR YMAX IS COMPUTED IN DRIV AS DESCRIBED        KM   793
C               UNDER IERROR BELOW.                                     KM   794
C               IF ERROR CONTROL PER SS UNITS OF T IS DESIRED, SET SS   KM   795
C               TO A POSITIVE NUMBER AFTER STATEMENT 10 (WHERE IT IS    KM   796
C               NOW SET TO ZERO), AND UPDATE IT AFTER STATEMENT 60.     KM   797
C               SEE ALSO THE COMMENTS ON SS AND YMAX BELOW.             KM   798
C  IERROR =   THE ERROR FLAG WITH VALUES AND MEANINGS AS FOLLOW.        KM   799
C         1     ABSOLUTE ERROR IS CONTROLLED.  YMAX(I) = 1.0.         KM   800
C         2     ERROR RELATIVE TO ABS(Y) IS CONTROLLED.  IF Y(I) = 0.0 KM   801
C               A DIVIDE ERROR WILL OCCUR.  YMAX(I) = ABS(Y(I)).      KM   802
C         3     ERROR RELATIVE TO THE LARGEST VALUE OF ABS(Y(I)) SEEN KM   803
C               SO FAR IS CONTROLLED.  IF THE INITIAL VALUE OF Y(I) IS KM   804
C               0.0, THEN YMAX(I) IS SET TO 1.0 INITIALLY AND REMAINS KM   805
C               AT LEAST 1.0.                                         KM   806
C  MF     =   THE METHOD FLAG (USED ONLY ON FIRST CALL, UNLESS        KM   807
C               INDEX = -1).  ALLOWED VALUES ARE 10, 11, 12, 13,      KM   808
C               20, 21, 22, 23.  MF IS AN INTEGER WITH TWO DECIMAL    KM   809
C               DIGITS, METH AND MITER (MF = 10*METH + MITER).  (MF   KM   810
C               CAN BE THOUGHT OF AS THE ORDERED PAIR (METH,MITER).)  KM   811
C               METH IS THE BASIC METHOD INDICATOR.                   KM   812
C                 METH = 1 INDICATES VARIABLE-STEP SIZE, VARIABLE-    KM   813
C                          ORDER ADAMS METHOD, SUITABLE FOR NON-      KM   814
C                          STIFF PROBLEMS.                            KM   815
C                 METH = 2 INDICATES VARIABLE-STEP SIZE, VARIABLE-    KM   816
C                          ORDER BACKWARD DIFFERENTIATION METHOD,     KM   817
C                          SUITABLE FOR STIFF PROBLEMS.               KM   818
C               MITER INDICATES THE METHOD OF ITERATIVE CORRECTION    KM   819
C                 (NONLINEAR SYSTEM SOLUTION).                        KM   820
C                 MITER = 0 INDICATES FUNCTIONAL ITERATION (NO        KM   821
C                           PARTIAL DERIVATIVES NEEDED).              KM   822
C                 MITER = 1 INDICATES A CHORD OR SEMI-STATIONARY      KM   823
C                           NEWTON METHOD WITH CLOSED FORM (EXACT)    KM   824
C                           JACOBIAN, WHICH IS COMPUTED IN THE        KM   825
C                           USER SUPPLIED SUBROUTINE                  KM   826
C                           PEDERV(N,T,Y,PD,NO) DESCRIBED BELOW.      KM   827
C                 MITER = 2 INDICATES A CHORD  OR SEMI-STATIONARY     KM   828
C                           NEWTON METHOD WITH AN INTERNALLY          KM   829
C                           COMPUTED FINITE DIFFERENCE APPROXIMATION  KM   830
C                           TO THE JACOBIAN.                          KM   831
C                 MITER = 3 INDICATES A CHORD OR SEMI-STATIONARY      KM   832
C                           NEWTON METHOD WITH AN INTERNALLY          KM   833
C                           COMPUTED DIAGONAL MATRIX APPROXIMATION    KM   834
C                           TO THE JACOBIAN, BASED ON A DIRECTIONAL   KM   835
C                           DERIVATIVE.                               KM   836
C  INDEX =    INTEGER USED ON INPUT TO INDICATE TYPE OF CALL,         KM   837
C               WITH THE FOLLOWING VALUES AND MEANINGS..              KM   838
C         1     THIS IS THE FIRST CALL FOR THIS PROBLEM.              KM   839
C         0     THIS IS NOT THE FIRST CALL FOR THIS PROBLEM,          KM   840
C               AND INTEGRATION IS TO CONTINUE.                       KM   841
C        -1     THIS IS NOT THE FIRST CALL FOR THE PROBLEM,           KM   842
C               AND THE USER HAS RESET N, EPS, AND/OR MF.             KM   843
```

```
C          2    SAME AS 0 EXCEPT THAT TOUT IS TO BE HIT                    KM   844
C               EXACTLY (NO INTERPOLATION IS DONE).                        KM   845
C               ASSUMES TOUT .GE. THE CURRENT T.                           KM   846
C          3    SAME AS 0 EXCEPT CONTROL RETURNS TO CALLING                KM   847
C               PROGRAM AFTER ONE STEP.  TOUT IS IGNORED.                  KM   848
C             SINCE THE NORMAL OUTPUT VALUE OF INDEX IS 0,                 KM   849
C             IT NEED NOT BE RESET FOR NORMAL CONTINUATION.                KM   850
C                                                                          KM   851
C     BIGSTP = MAXIMUM STEP SIZE ALLOWED    (J.R. MARTINEZ -- 6.15.77)     KM   852
C                                                                          KM   853
C-------------------------------------------------------------------------KM   854
C                                                                          KM   855
      INTEGER IERROR, INDEX, MF, N                                         KM   856
      INTEGER IPIV, JSTART, KFLAG, MFC, NC, NFE, NJE,                      KM   857
     1        NQUSED, NSQ, NSTEP                                           KM   858
      INTEGER I, KGO, NHCUT, N0                                            KM   859
      INTEGER LOUT                                                         KM   860
C(6                                                                        KM   861
C/14                                                                       KM   862
      DIMENSION Y(150,6)                                                   KM   863
      DIMENSION Y0(N)                                                      KM   864
C                                                                          KM   865
      COMMON /EPCOM1/T,H,HMIN,HMAX,EPSC,SS,UROUND,NC,MFC,KFLAG,JSTART      KM   866
      COMMON /EPCOM2/ YMAX(150)                                            KM   867
      COMMON /EPCOM3/ ERROR(150)                                           KM   868
      COMMON /EPCOM4/ SAVE1(150)                                           KM   869
      COMMON /EPCOM5/ SAVE2(150)                                           KM   870
      COMMON /EPCOM6/ PW(4500)                                             KM   871
      COMMON /EPCOM7/ IPIV(150)                                            KM   872
      COMMON /EPCOM8/ EPSJ,NSQ                                             KM   873
      COMMON /EPCOM9/ HUSED,NQUSED,NSTEP,NFE,NJE                           KM   874
      COMMON /EPCM12/ PWL(3600),  WORKR(30), DUPPER(4), DLOWER(4)          KM   875
      INTEGER BCFLAG                                                       KM   876
      COMMON/CHEM1/ NOSTAT,              NOSTM1,             NOREAC,       KM   877
     1              NOSPEC,              NSTOY,              NK            KM   878
      COMMON/CHEM2/ CONIN(40,5),         WTMOLE(40),         RATKON(55),   KM   879
     1              RATEFF(55),          RATEV(2,5),         QRATE,        KM   880
     2              NVRATE,              LOCVRT(2)                         KM   881
      COMMON/CHEM3/ ZEE(5),              DELZ(4),            HTCELL(6),    KM   882
     1              TDELZ(2) ,           DFINIT(6),          SCALOW(4),    KM   883
     3              BCFLAG(40),          DPRATE(10),         DEPOWR(10),   KM   884
     2              OCOF(5),             FLXWAL(40),         FLXDGE(40),   KM   885
     4              LOCDPF(10),          NDPFLX ,            SCALUP(4)     KM   886
C                                                                          KM   887
      DATA LOUT /6/                                                        KM   888
C(3                                                                        KM   889
      DATA HCUT /0.100/                                                    KM   890
      DATA FOUR /4.000/, HUNDRD /1.0E2/, ONE  /1.000/,                     KM   891
     1     TEN  /1.0E1/, ZERO   /0.000/                                    KM   892
      DATA ERLIMT , KKKER  /1.0E-2,  0/                                    KM   893
C)3                                                                        KM   894
      IF (INDEX .EQ. 0) GO TO 20                                           KM   895
      IF (INDEX .EQ. 2) GO TO 25                                           KM   896
      IF (INDEX .EQ. -1) GO TO 30                                          KM   897
      IF (INDEX .EQ. 3) GO TO 40                                           KM   898
```

```
      IF (INDEX .NE. 1) GO TO 430                                       KM   899
      IF (EPS .LE. ZERO) GO TO 400                                      KM   900
      IF (N .LE. 0) GO TO 410                                           KM   901
      IF (N .GT.150) GO TO 440                                          KM   902
      IF ((T0-TOUT)*H0 .GE. ZERO) GO TO 420                             KM   903
C-----------------------------------------------------------------------KM   904
C IF INITIAL VALUES FOR YMAX OTHER THAN THOSE BELOW ARE DESIRED,        KM   905
C THEY SHOULD BE SET HERE.  ALL YMAX(I) MUST BE POSITIVE.  IF           KM   906
C VALUES FOR HMIN OR HMAX, THE BOUNDS ON THE ABSOLUTE VALUE OF H,       KM   907
C OTHER THAN THOSE BELOW, ARE DESIRED, THEY ALSO SHOULD BE SET HERE.    KM   908
C IF ERROR PER SS UNITS OF T IS TO BE CONTROLLED, SS SHOULD BE SET      KM   909
C TO A POSITIVE VALUE BELOW.  ERROR PER UNIT STEP IS CONTROLLED         KM   910
C WHEN SS = 1.  THE DEFAULT VALUE FOR SS IS 0 AND YIELDS CONTROL        KM   911
C OF ERROR PER STEP.                                                    KM   912
C-----------------------------------------------------------------------KM   913
C SET UROUND, THE MACHINE ROUNDOFF CONSTANT, HERE.                      KM   914
C USE STATEMENT BELOW FOR SHORT PRECISION ON IBM 360 OR 370.            KM   915
C     UROUND = 9.53674E-7                                               KM   916
C USE STATEMENT BELOW FOR SINGLE PRECISION ON CDC 7600 OR 6600.         KM   917
C     UROUND = 7.105427406E-15                                          KM   918
C USE STATEMENT BELOW FOR SHORT PRECISION ON UNIVAC 1110.               KM   919
C     UROUND = 1.490116112E-8                                           KM   920
C-----------------------------------------------------------------------KM   921
      UROUND=7.105427406E-15                                            KM   922
      DO 10 I = 1,N                                                     KM   923
        GO TO (5, 6, 7), IERROR                                         KM   924
C IERROR    =   1, 2, 3  ----------------------------------------------KM   925
  5     YMAX(I) = ONE                                                   KM   926
        GO TO 10                                                        KM   927
  6     YMAX(I) =  ABS(Y0(I))                                           KM   928
        GO TO 10                                                        KM   929
  7     YMAX(I) =  ABS(Y0(I))                                           KM   930
      YMAX(I) = AMAX1( YMAX(I) , ERLIMT)                                KM   931
        IF (YMAX(I) .EQ. ZERO) YMAX(I) = ONE                            KM   932
 10     Y(I,1) = Y0(I)                                                  KM   933
      NC = N                                                            KM   934
      T = T0                                                            KM   935
      H = H0                                                            KM   936
      IF ((T+H) .EQ. T) WRITE(LOUT,15) T                                KM   937
 15   FORMAT(/46H*** MESSAGE FROM SUBROUTINE DRIVE IN EPISODE,,         KM   938
     1       24H THE O.D.E. SOLVER, ***/22H WARNING.. T + H = T =,      KM   939
     2       E18.8,18H IN THE NEXT STEP./)                              KM   940
C(2                                                                     KM   941
      HMIN =  ABS(H0)                                                   KM   942
      HMAX =  ABS(T0 - TOUT)*TEN                                        KM   943
      HMAX=BIGSTP                                                       KM   944
      EPSC = EPS                                                        KM   945
      MFC = MF                                                          KM   946
      JSTART = 0                                                        KM   947
      SS = ZERO                                                         KM   948
      N0 = N                                                            KM   949
      NSQ = N0*N0                                                       KM   950
C(1                                                                     KM   951
      EPSJ =  SQRT(UROUND)                                              KM   952
      NHCUT = 0                                                         KM   953
```

```
      GO TO 50                                                          KM   954
C TOP IS THE PREVIOUS OUTPUT VALUE OF T0 FOR USE IN HMAX, ------------KM   955
C(3                                                                     KM   956
  20  HMAX =  ABS(TOUT - TOP)*TEN                                       KM   957
      HMAX=BIGSTP                                                       KM   958
      GO TO 80                                                          KM   959
  25  HMAX =  ABS(TOUT - TOP)*TEN                                       KM   960
      HMAX=BIGSTP                                                       KM   961
C                                                                       KM   962
       IF ((T-TOUT)*H .GE. ZERO) GO TO 460                              KM   963
       GO TO 85                                                         KM   964
C                                                                       KM   965
 30   IF ((T-TOUT)*H .GE. ZERO) GO TO 450                               KM   966
      IF (MF .NE. MFC) JSTART = -1                                      KM   967
      NC = N                                                            KM   968
      EPSC = EPS                                                        KM   969
      MFC = MF                                                          KM   970
      GO TO 45                                                          KM   971
C                                                                       KM   972
 40   HMAX = H0                                                         KM   973
C                                                                       KM   974
 45   IF ((T+H) .EQ. T) WRITE(LOUT,15) T                                KM   975
C                                                                       KM   976
 50   CALL TSTEP (Y, N0)                                                KM   977
C                                                                       KM   978
      KGO = 1 - KFLAG                                                   KM   979
      GO TO (60, 100, 200, 300), KGO                                    KM   980
C KFLAG  =   0,  -1,  -2,  -3 ----------------------------------------KM   981
C                                                                       KM   982
 60   CONTINUE                                                          KM   983
C-----------------------------------------------------------------------KM   984
C NORMAL RETURN FROM TSTEP.                                             KM   985
C                                                                       KM   986
C THE WEIGHTS YMAX(I) ARE UPDATED.  IF DIFFERENT VALUES ARE DESIRED,    KM   987
C THEY SHOULD BE SET HERE.  IF SS IS TO BE UPDATED FOR CONTROL OF       KM   988
C ERROR PER SS UNITS OF T, IT SHOULD ALSO BE DONE HERE.  A TEST IS      KM   989
C MADE TO DETERMINE IF EPS IS TOO SMALL FOR MACHINE PRECISION.          KM   990
C                                                                       KM   991
C ANY OTHER TESTS OR CALCULATIONS THAT ARE REQUIRED AFTER EACH STEP     KM   992
C SHOULD BE INSERTED HERE.                                              KM   993
C                                                                       KM   994
C IF INDEX = 3, Y0 IS SET TO THE CURRENT Y VALUES ON RETURN.            KM   995
C IF INDEX = 2, H IS CONTROLLED TO HIT TOUT (WITHIN ROUNDOFF            KM   996
C ERROR), AND THEN THE CURRENT Y VALUES ARE PUT IN Y0 ON                KM   997
C RETURN.  FOR ANY OTHER VALUE OF INDEX, CONTROL RETURNS TO             KM   998
C THE INTEGRATOR UNLESS TOUT HAS BEEN REACHED.  THEN                    KM   999
C INTERPOLATED VALUES OF Y ARE COMPUTED AND STORED IN Y0 ON             KM  1000
C RETURN.                                                               KM  1001
C IF INTERPOLATION IS NOT DESIRED, THE CALL TO INTERP SHOULD            KM  1002
C BE DELETED AND CONTROL TRANSFERRED TO STATEMENT 500 INSTEAD           KM  1003
C OF 520.                                                               KM  1004
C-----------------------------------------------------------------------KM  1005
C..........CHECK FOR NEGATIVE VALUES OF Y                               KM  1006
      CALL CHECKY(T,Y,Y0,N,KOK)                                         KM  1007
      IF(KOK .LT. 0) RETURN                                             KM  1008
```

```
C..........COMPUTE STEADY STATE SPECIES                                 KM  1009
      CALL STEADY(Y,N)                                                  KM  1010
      KKKER = KKKER + 1                                                 KM  1011
      D = ZERO                                                          KM  1012
      DO 70 I = 1,N                                                     KM  1013
        AYI =  ABS(Y(I,1))                                              KM  1014
        GO TO (70, 66, 67), IERROR                                      KM  1015
C IERROR  =     1,  2,  3 ***************************************************KM  1016
 66     YMAX(I) = AYI                                                   KM  1017
        GO TO 70                                                        KM  1018
  67    YMAX(I) = AMAX1(YMAX(I), AYI)                                   KM  1019
      IF(KKKER.LT.20)  GO TO 70                                         KM  1020
C .........UPDATE YMAX EVERY 20 STEPS USING AYI AND ERLIMT              KM  1021
      KKKER = 0                                                         KM  1022
      YMAX(I) = AMAX1(AYI,ERLIMT)                                       KM  1023
  70    D = D + (AYI/YMAX(I))**2                                        KM  1024
      D = D*(UROUND/EPS)**2                                             KM  1025
      IF (D .GT.  FLOAT(N)) GO TO 250                                   KM  1026
      IF (INDEX .EQ. 3) GO TO 500                                       KM  1027
      IF (INDEX .EQ. 2) GO TO 85                                        KM  1028
 80   IF ((T-TOUT)*H .LT. ZERO) GO TO 45                                KM  1029
      CALL INTERP (TOUT, Y, NO, YO)                                     KM  1030
      TO = TOUT                                                         KM  1031
      GO TO 520                                                         KM  1032
 85   IF (((T+H)-TOUT)*H .LE. ZERO) GO TO 45                            KM  1033
C(1                                                                     KM  1034
      IF ( ABS(T-TOUT) .LE. HUNDRD*UROUND*HMAX) GO TO 500               KM  1035
      IF ((T-TOUT)*H .GE. ZERO) GO TO 500                               KM  1036
      H = (TOUT - T)*(ONE - FOUR*UROUND)                                KM  1037
      JSTART = -1                                                       KM  1038
      GO TO 45                                                          KM  1039
C*****************************************************************************KM  1040
C ON AN ERROR RETURN FROM TSTEP, AN IMMEDIATE RETURN OCCURS IF          KM  1041
C KFLAG = -2, AND RECOVERY ATTEMPTS ARE MADE OTHERWISE.                 KM  1042
C TO RECOVER, H AND HMIN ARE REDUCED BY A FACTOR OF .1 UP TO 10         KM  1043
C TIMES BEFORE GIVING UP.                                               KM  1044
C*****************************************************************************KM  1045
 100  WRITE (LOUT,101)                                                 KM  1046
 101  FORMAT (/46H***  MESSAGE FROM SUBROUTINE DRIVE IN EPISODE,,       KM  1047
     1         24H THE O.D.E. SOLVER,  ***/)                          KM  1048
      WRITE(LOUT,105) T,HMIN                                            KM  1049
 105  FORMAT(//35H KFLAG = -1 FROM INTEGRATOR AT T = ,E18.8/            KM  1050
     1       40H  ERROR TEST FAILED WITH ABS(H) = HMIN =,E18.8/)        KM  1051
 110  IF (NHCUT .EQ. 10) GO TO 150                                      KM  1052
      NHCUT = NHCUT + 1                                                 KM  1053
      HMIN = HCUT*HMIN                                                  KM  1054
      H = HCUT*H                                                        KM  1055
      WRITE (LOUT,115) H                                                KM  1056
 115  FORMAT(24H  H HAS BEEN REDUCED TO ,E18.8,                         KM  1057
     1       26H  AND STEP WILL BE RETRIED//)                           KM  1058
      JSTART = -1                                                       KM  1059
      GO TO 45                                                          KM  1060
C                                                                       KM  1061
 150  WRITE (LOUT,155)                                                  KM  1062
 155  FORMAT(//44H PROBLEM APPEARS UNSOLVABLE WITH GIVEN INPUT//)       KM  1063
```

```
      GO TO 500                                                          KM  1064
C                                                                        KM  1065
 200  WRITE (LOUT,101)                                                   KM  1066
      WRITE (LOUT,205) T,H,EPS                                           KM  1067
 205  FORMAT(//16H KFLAG = -2  T =,E18.8,4H H =,E18.8,6H EPS =,E18.8/    KM  1068
     1       50H  THE REQUESTED ERROR IS TOO SMALL FOR INTEGRATOR.//)    KM  1069
      GO TO 500                                                          KM  1070
C                                                                        KM  1071
 250  WRITE (LOUT,101)                                                   KM  1072
      WRITE (LOUT,255) T,EPS                                             KM  1073
 255  FORMAT(//46H INTEGRATION HALTED BY SUBROUTINE DRIVE AT T =,        KM  1074
     1       E18.8/43H EPS IS TOO SMALL FOR MACHINE PRECISION AND/       KM  1075
     2       29H PROBLEM BEING SOLVED.  EPS =,E18.8//)                   KM  1076
      KFLAG = -2                                                         KM  1077
      GO TO 500                                                          KM  1078
C                                                                        KM  1079
 300  WRITE (LOUT,101)                                                   KM  1080
      WRITE (LOUT,305) T                                                 KM  1081
 305  FORMAT(//38H KFLAG = -3 FROM INTEGRATOR AT T =,E18.8/              KM  1082
     1       45H  CORRECTOR CONVERGENCE COULD NOT BE ACHIEVED/)          KM  1083
      GO TO 110                                                          KM  1084
C                                                                        KM  1085
 400  WRITE (LOUT,101)                                                   KM  1086
      WRITE (LOUT,405) EPS                                               KM  1087
 405  FORMAT(//35H ILLEGAL INPUT.. EPS .LE. 0. EPS = ,E18.8//)           KM  1088
      INDEX = -4                                                         KM  1089
      RETURN                                                             KM  1090
C                                                                        KM  1091
 410  WRITE (LOUT,101)                                                   KM  1092
      WRITE (LOUT,415) N                                                 KM  1093
 415  FORMAT(//31H ILLEGAL INPUT.. N .LE. 0. N = ,I8//)                  KM  1094
      INDEX = -4                                                         KM  1095
      RETURN                                                             KM  1096
C                                                                        KM  1097
 420  WRITE (LOUT,101)                                                   KM  1098
      WRITE (LOUT,425) T0,TOUT,H0                                        KM  1099
 425  FORMAT(//39H ILLEGAL INPUT.. (T0 - TOUT)*H0 .GE. 0./               KM  1100
     1       5H T0 =,E18.8,7H TOUT =,E18.8,5H H0 =,E18.8//)             KM  1101
      INDEX = -4                                                         KM  1102
      RETURN                                                             KM  1103
C                                                                        KM  1104
 430  WRITE (LOUT,101)                                                   KM  1105
      WRITE (LOUT,435) INDEX                                             KM  1106
 435  FORMAT(//24H ILLEGAL INPUT.. INDEX =,I8//)                         KM  1107
      INDEX = -4                                                         KM  1108
      RETURN                                                             KM  1109
C                                                                        KM  1110
 440  WRITE (LOUT,101)                                                   KM  1111
      WRITE (LOUT,445) N                                                 KM  1112
 445  FORMAT (//39H ILLEGAL INPUT.  THE NUMBER OF ORDINARY/              KM  1113
     1        43H DIFFERENTIAL EQUATIONS BEING SOLVED IS N =, I6/       KM  1114
     2        42H STORAGE ALLOCATION IN SUBROUTINE DRIVE IS/            KM  1115
     3        46H TOO SMALL.  SEE COMMENTS IN SUBROUTINE DRIVE./)       KM  1116
      INDEX = -4                                                         KM  1117
      RETURN                                                             KM  1118
```

```
C                                                                       KM  1119
 450  WRITE (LOUT,101)                                                  KM  1120
      WRITE (LOUT,455) T,TOUT,H                                         KM  1121
 455  FORMAT(//46H INDEX = -1 ON INPUT WITH (T - TOUT)*H .GE. 0./       KM  1122
     1       44H INTERPOLATION WAS DONE AS ON NORMAL RETURN./           KM  1123
     2       41H DESIRED PARAMETER CHANGES WERE NOT MADE./              KM  1124
     3       4H T =,E18.8,7H TOUT =,E18.8,4H H =,E18.8//)               KM  1125
      CALL INTERP (TOUT, Y, N0, Y0)                                     KM  1126
      T0 = TOUT                                                         KM  1127
      INDEX = -5                                                        KM  1128
      RETURN                                                            KM  1129
C                                                                       KM  1130
 460  WRITE (LOUT,101)                                                  KM  1131
      WRITE (LOUT,465) T,TOUT,H                                         KM  1132
 465  FORMAT(//45H INDEX = 2 ON INPUT WITH (T - TOUT)*H .GE. 0./        KM  1133
     1       4H T =,E18.8,7H TOUT =,E18.8,4H H =,E18.8//)               KM  1134
      INDEX = -6                                                        KM  1135
      RETURN                                                            KM  1136
C                                                                       KM  1137
 500  T0 = T                                                            KM  1138
      DO 510 I = 1,N                                                    KM  1139
 510    Y0(I) = Y(I,1)                                                  KM  1140
 520  INDEX = KFLAG                                                     KM  1141
      TOP = T0                                                          KM  1142
      H0 = HUSED                                                        KM  1143
      IF (KFLAG .NE. 0) H0 = H                                          KM  1144
      RETURN                                                            KM  1145
C----------------------  END OF SUBROUTINE DRIVE  ----------------------KM  1146
      END                                                               KM  1147



      SUBROUTINE FMAX(A,N,B)                                            KM  1148
C                                                                       KM  1149
      DIMENSION A(1)                                                    KM  1150
C                                                                       KM  1151
      C = A(1)                                                          KM  1152
      DO 10 I = 2,N                                                     KM  1153
      C = AMAX1(C,A(I))                                                 KM  1154
   10 CONTINUE                                                          KM  1155
      B = C                                                             KM  1156
      RETURN                                                            KM  1157
      END                                                               KM  1158



      SUBROUTINE INTERP (TOUT, Y, N0, Y0)                               KM  1159
C-----------------------------------------------------------------------KM  1160
C SUBROUTINE INTERP COMPUTES INTERPOLATED VALUES OF THE DEPENDENT       KM  1161
C VARIABLE Y AND STORES THEM IN Y0.  THE INTERPOLATION IS TO THE        KM  1162
C POINT T = TOUT AND USES THE NORDSIECK HISTORY ARRAY Y AS FOLLOWS..    KM  1163
C                          N0                                           KM  1164
C                Y0(I)  =  SUM  Y(I,J+1)*S**J ,                         KM  1165
C                          J=0                                          KM  1166
C WHERE S = -(T-TOUT)/H.                                                KM  1167
```

```
C-----------------------------------------------------------------------KM  1168
      INTEGER N0                                                        KM  1169
      INTEGER JSTART, KFLAG, MF, N                                      KM  1170
      INTEGER I, J, L                                                   KM  1171
      DIMENSION Y0(N0),Y(N0, 6)                                         KM  1172
C                                                                       KM  1173
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART     KM  1174
C(1                                                                     KM  1175
      DATA ONE /1.000/                                                  KM  1176
      DO 10 I = 1,N                                                     KM  1177
 10     Y0(I) = Y(I,1)                                                 KM  1178
      L = JSTART + 1                                                    KM  1179
      S = (TOUT - T)/H                                                  KM  1180
      S1 = ONE                                                          KM  1181
      DO 30 J = 2,L                                                     KM  1182
        S1 = S1*S                                                       KM  1183
        DO 20 I = 1,N                                                   KM  1184
 20       Y0(I) = Y0(I) + S1*Y(I,J)                                     KM  1185
 30     CONTINUE                                                        KM  1186
      RETURN                                                            KM  1187
C----------------------  END OF SUBROUTINE INTERP  --------------------KM  1188
      END                                                               KM  1189



      SUBROUTINE ISTATE                                                 KM  1190
C                                                                       KM  1191
C     ----   SUBROUTINE ISTATE CALCULATES THE INITIAL CONCENTRATIONS     KM  1192
C            OF THE FOLLOWING SPECIES USING THE QUASI-STEADY STATE      KM  1193
C            APPROXIMATION.   THIS VERSION OF ISTATE IS TO BE USED      KM  1194
C            ONLY FOR THE ERT 30 SPECIES X 51 REACTION MODEL OF 4.1.78. KM  1195
C                                                                       KM  1196
C            HONO    NO2     O       NO3     N2O5    AROH              KM  1197
C            ARCO    AO2     PA32    ARO     AO      PAN               KM  1198
C            RCO3    RO2     PAO     HNO4    O                         KM  1199
C                                                                       KM  1200
      COMMON/CHEM1/ NOSTAT,             NOSTM1,             NOREAC,     KM  1201
     1              NOSPEC,             NSTDY,              NK          KM  1202
      COMMON/CHEM2/ CONIN(40,5),        WTMOLE(40),         RATKON(55), KM  1203
     1              RATEFF(55),         RATEV(2,5),         QRATE,      KM  1204
     2              NVRATE,             LOCVRT(2)                       KM  1205
      DIMENSION  C(40),  R(55),  A(2,2),  B(2)                          KM  1206
      EQUIVALENCE (RATEFF, R)                                           KM  1207
      DATA YES /3HYES/                                                  KM  1208
C                                                                       KM  1209
      DO 200 K=1,NOSTAT                                                 KM  1210
C                                                                       KM  1211
      CALL XMIT(NOSPEC,CONIN(1,K),C)                                    KM  1212
      IF(QRATE.EQ.YES)  CALL RATEHI(K)                                  KM  1213
      CALL UNMIXR(K)                                                    KM  1214
C                                                                       KM  1215
C     ------   NO2    ------                                            KM  1216
      C(2) = C(1)*C(3)*R(1)/R(3)                                        KM  1217
C     ------    O     ------                                            KM  1218
      C(25) = R(1)*C(2)/( R(2) + R(24)*C(8) )                           KM  1219
```

```
C     ------   HONO   ---------                                          KM  1220
      AA = 2.*R(5)                                                      KM  1221
      BB = R(6)                                                         KM  1222
      CC = - 2.*R(4)*C(1)*C(2)*C(30) - R(7)*C(1)*C(24)                  KM  1223
      C(4) = (-BB +(BB*BB -4.*AA*CC)**.5)/2./AA                         KM  1224
C     ------   D    -------                                             KM  1225
      C(19) = (.5*R(22)*C(3)*C(8))/(R(23)*C(2) + R(25)*C(1))            KM  1226
C     ------  NO3 + N2O5   ------                                       KM  1227
      A(1,1) = R(14)*C(1) + R(15) *C(2)                                 KM  1228
      A(1,2) = -R(17)                                                   KM  1229
      A(2,1) = -R(15)*C(2)                                              KM  1230
      A(2,2) = R(16)*C(30) + R(17)                                      KM  1231
      B(1) = R(13)*C(2)*C(3) + R(23)*C(2)*C(19)                         KM  1232
      B(2) = 0.0                                                        KM  1233
      CALL SOL2B2 (A,B)                                                 KM  1234
      C(10) = B(1)                                                      KM  1235
      C(27) = B(2)                                                      KM  1236
C     ------    AROH    ------                                          KM  1237
      C(22) = R(41)*C(13)/R(42)                                         KM  1238
C     ------    ARCO    ------                                          KM  1239
      C(14) = R(42)*C(22)*C(24)/(R(43)*C(1))                            KM  1240
C     ------    AO2    ------                                           KM  1241
      C(17) = R(18)*C(8)*C(24)/(R(19)*C(1))                             KM  1242
C     ------     PAO2                                                   KM  1243
      C(18) = R(26)*C(9)*C(24)/((R(27) + R(28))*C(1))                   KM  1244
C     ------   HNO4 ------                                              KM  1245
      C(20) = R(11)*C(2)*C(11)/R(21)                                    KM  1246
C     ------    ARO    ------                                          KM  1247
      C(21) = R(44)*C(13)*C(24)/(R(45)*C(1))                            KM  1248
C     ------    AO    ------                                           KM  1249
      C(23) = R(19)*C(1)*C(17)/R(20)                                    KM  1250
C     ------    PAN + RCO3    -----                                     KM  1251
      A(1,1) = R(37)                                                    KM  1252
      A(1,2) = -R(35)*C(2)                                              KM  1253
      B(1) = 0.0                                                        KM  1254
      A(2,1) = - R(37)                                                  KM  1255
      A(2,2) = R(35)*C(1) + R(36)*C(2)                                  KM  1256
      B(2) = .25*R(22)*C(3)*C(8) + R(33)*C(7)*C(24)                     KM  1257
      CALL SOL2B2(A,B)                                                  KM  1258
      C(12) = B(1)                                                      KM  1259
      C(16) = B(2)                                                      KM  1260
C     ------   RO2 + PAO    ------                                      KM  1261
      A(1,1) = R(30)*C(1) + R(40)*C(11)                                 KM  1262
      A(1,2) = - R(29)                                                  KM  1263
      B(1) = .4*R(24)*C(8)*C(25) + .15*R(27)*C(1)*C(18) + R(34)*C(7)    KM  1264
     1                                          + R(35)*C(1)*C(16)     KM  1265
      A(2,1) = -R(30)*C(1)                                              KM  1266
      A(2,2) = R(29) + R(31) + R(32)*C(2)                               KM  1267
      B(2) = .85*R(27)*C(1)*C(18)                                       KM  1268
      CALL SOL2B2 (A,B)                                                 KM  1269
      C(15) = B(1)                                                      KM  1270
      C(26) = B(2)                                                      KM  1271
      CALL XMIT (NOSPEC, C, CONIN(1,K)  )                               KM  1272
200   CONTINUE                                                          KM  1273
      RETURN                                                            KM  1274
```

```
      END                                                                KM  1275


      FUNCTION ITHOUR(TMIN)                                              KM  1276
C                                                                        KM  1277
C     CONVERTS TIME IN MINUTES FROM MIDNIGHT TO HOURS ON 24-HOUR CLOCK   KM  1278
C                                                                        KM  1279
      IA = IFIX(TMIN/.600 + 1.E-04)                                      KM  1280
      IB = IA/100                                                        KM  1281
      IM = IFIX(TMIN+1.E-04) - 60*IB                                     KM  1282
      IF(IM.LT.60) GO  TO 10                                             KM  1283
      IM = IM - 60                                                       KM  1284
      IB = IB + 1                                                        KM  1285
 10   ITHOUR = IB*100 + IM                                               KM  1286
      RETURN                                                             KM  1287
      END                                                                KM  1288


      SUBROUTINE JACOB(A,C,R,N)                                          KM  1289
C                                                                        KM  1290
C     JACOB COMPUTES THE JACOBIAN OF THE CHEMICAL RATE EQUATIONS.        KM  1291
C                                                                        KM  1292
C     ERT PHOTOCHEMICAL MECHANISM (4.1.78)  (30 SPECIES X 51 REACTIONS)  KM  1293
C                                                                        KM  1294
      DIMENSION    A(N,N),       C(1),       R(1)                        KM  1295
C                                                                        KM  1296
C     ZERO THE A ARRAY                                                   KM  1297
      NN2 = N*N                                                          KM  1298
      CALL XMIT(-NN2,0.0,A)                                              KM  1299
C                                                                        KM  1300
      A(  1,  1) = - R(  3)*C(  3) - R(  4)*C(  2)*C( 30) - R(  7)       KM  1301
     *    *C( 24) - R( 10)*C( 11) - R( 14)*C( 10) - R( 19)*C( 17) -      KM  1302
     *    R( 25)*C( 19) - R( 27)*C( 18) - R( 28)*C( 18) - R( 30)*C( 15)  KM  1303
     *     - R( 35)*C( 16) - R( 43)*C( 14) - R( 45)*C( 21)               KM  1304
      A(  1,  2) = + R(  1) - R(  4)*C(  1)*C( 30)                       KM  1305
      A(  1,  3) = - R(  3)*C(  1)                                       KM  1306
      A(  1,  4) = +    2.00*R(  5)*C(  4) + R(  6)                      KM  1307
      A(  1, 10) = - R( 14)*C(  1)                                       KM  1308
      A(  1, 11) = - R( 10)*C(  1)                                       KM  1309
      A(  1, 14) = - R( 43)*C(  1)                                       KM  1310
      A(  1, 15) = - R( 30)*C(  1)                                       KM  1311
      A(  1, 16) = - R( 35)*C(  1)                                       KM  1312
      A(  1, 17) = - R( 19)*C(  1)                                       KM  1313
      A(  1, 18) = - R( 27)*C(  1) - R( 28)*C(  1)                       KM  1314
      A(  1, 19) = - R( 25)*C(  1)                                       KM  1315
      A(  1, 21) = - R( 45)*C(  1)                                       KM  1316
      A(  1, 24) = - R(  7)*C(  1)                                       KM  1317
C     A(  1, 30) = - R(  4)*C(  1)*C(  2)                                KM  1318
      A(  2,  1) = + R(  3)*C(  3) - R(  4)*C(  2)*C( 30) + R( 10)       KM  1319
     *    *C( 11) +    2.00*R( 14)*C( 10) + R( 19)*C( 17) + R( 25)       KM  1320
     *    *C( 19) + R( 27)*C( 18) + R( 30)*C( 15) + R( 43)*C( 14) +      KM  1321
     *    R( 45)*C( 21)                                                  KM  1322
      A(  2,  2) = - R(  1) - R(  4)*C(  1)*C( 30) - R(  8)*C( 24) -     KM  1323
```

```
     *    R( 11)*C( 11) - R( 13)*C(  3) - R( 15)*C( 10) - R( 23)*C( 19) KM  1324
     *     - R( 32)*C( 26) - R( 36)*C( 16)                              KM  1325
      A(  2,  3) = + R(  3)*C(  1) - R( 13)*C(  2)                      KM  1326
      A(  2,  4) = +    2.00*R(  5)*C(  4)                              KM  1327
      A(  2, 10) = +    2.00*R( 14)*C(  1) - R( 15)*C(  2)              KM  1328
      A(  2, 11) = + R( 10)*C(  1) - R( 11)*C(  2)                      KM  1329
      A(  2, 12) = + R( 37)                                             KM  1330
      A(  2, 14) = + R( 43)*C(  1)                                      KM  1331
      A(  2, 15) = + R( 30)*C(  1)                                      KM  1332
      A(  2, 16) = - R( 36)*C(  2)                                      KM  1333
      A(  2, 17) = + R( 19)*C(  1)                                      KM  1334
      A(  2, 18) = + R( 27)*C(  1)                                      KM  1335
      A(  2, 19) = - R( 23)*C(  2) + R( 25)*C(  1)                      KM  1336
      A(  2, 20) = + R( 21)                                             KM  1337
      A(  2, 21) = + R( 45)*C(  1)                                      KM  1338
      A(  2, 24) = - R(  8)*C(  2)                                      KM  1339
      A(  2, 26) = - R( 32)*C(  2)                                      KM  1340
      A(  2, 27) = + R( 17)                                             KM  1341
C     A(  2, 30) = - R(  4)*C(  1)*C(  2)                               KM  1342
      A(  3,  1) = - R(  3)*C(  3)                                      KM  1343
      A(  3,  2) = - R( 13)*C(  3)                                      KM  1344
      A(  3,  3) = - R(  3)*C(  1) - R( 13)*C(  2) - R( 22)*C(  8)      KM  1345
      A(  3,  8) = - R( 22)*C(  3)                                      KM  1346
      A(  3, 25) = + R(  2)                                             KM  1347
      A(  4,  1) = +    2.00*R(  4)*C(  2)*C( 30) + R(  7)*C( 24)       KM  1348
      A(  4,  2) = +    2.00*R(  4)*C(  1)*C( 30) +    0.15*R( 32)      KM  1349
     *    *C( 26)                                                      KM  1350
      A(  4,  4) = -    4.00*R(  5)*C(  4) - R(  6)                     KM  1351
      A(  4, 24) = + R(  7)*C(  1)                                      KM  1352
      A(  4, 26) = +    0.15*R( 32)*C(  2)                              KM  1353
C     A(  4, 30) = +    2.00*R(  4)*C(  1)*C(  2)                       KM  1354
      A(  5,  5) = - R(  9)*C( 24)                                      KM  1355
      A(  5,  6) = + R( 38) + R( 39)*C( 24)                             KM  1356
      A(  5,  7) = + R( 34)                                             KM  1357
      A(  5, 24) = - R(  9)*C(  5) + R( 39)*C(  6)                      KM  1358
      A(  6,  1) = +    0.50*R( 25)*C( 19)                              KM  1359
      A(  6,  2) = +    0.50*R( 23)*C( 19)                              KM  1360
      A(  6,  3) = +    0.50*R( 22)*C(  8)                              KM  1361
      A(  6,  6) = - R( 38) - R( 39)*C( 24)                             KM  1362
      A(  6,  8) = +    0.50*R( 22)*C(  3)                              KM  1363
      A(  6, 19) = +    0.50*R( 23)*C(  2) +    0.50*R( 25)*C(  1)      KM  1364
      A(  6, 23) = + R( 20)                                             KM  1365
      A(  6, 24) = - R( 39)*C(  6)                                      KM  1366
      A(  6, 26) = +    0.50*R( 29)                                     KM  1367
      A(  7,  1) = +    0.50*R( 25)*C( 19) + R( 43)*C( 14)              KM  1368
      A(  7,  2) = +    0.50*R( 23)*C( 19) +    0.15*R( 32)*C( 26)      KM  1369
      A(  7,  3) = +    0.50*R( 22)*C(  8)                              KM  1370
      A(  7,  7) = - R( 33)*C( 24) - R( 34)                             KM  1371
      A(  7,  8) = +    0.50*R( 22)*C(  3) +    0.30*R( 24)*C( 25)      KM  1372
      A(  7, 14) = + R( 43)*C(  1)                                      KM  1373
      A(  7, 19) = +    0.50*R( 23)*C(  2) +    0.50*R( 25)*C(  1) +    KM  1374
     *    R( 49)*C( 28)                                                KM  1375
      A(  7, 23) = + R( 20)                                             KM  1376
      A(  7, 24) = - R( 33)*C(  7)                                      KM  1377
      A(  7, 25) = +    0.30*R( 24)*C(  8)                              KM  1378
```

```
 A(  7, 26) = +    0.50*R( 29) +    0.50*R( 31) +    0.15*R( 32)   KM  1379
*    *C(  2)                                                       KM  1380
 A(  7, 28) = + R( 49)*C( 19)                                      KM  1381
 A(  8,  3) = - R( 22)*C(  8)                                      KM  1382
 A(  8,  8) = - R( 18)*C( 24) - R( 22)*C(  3) - R( 24)*C( 25)      KM  1383
 A(  8, 24) = - R( 18)*C(  8)                                      KM  1384
 A(  8, 25) = - R( 24)*C(  8)                                      KM  1385
 A(  9,  9) = - R( 26)*C( 24)                                      KM  1386
 A(  9, 24) = - R( 26)*C(  9)                                      KM  1387
 A( 10,  1) = - R( 14)*C( 10)                                      KM  1388
 A( 10,  2) = + R( 13)*C(  3) - R( 15)*C( 10) + R( 23)*C( 19)      KM  1389
 A( 10,  3) = + R( 13)*C(  2)                                      KM  1390
 A( 10, 10) = - R( 14)*C(  1) - R( 15)*C(  2)                      KM  1391
 A( 10, 19) = + R( 23)*C(  2)                                      KM  1392
 A( 10, 27) = + R( 17)                                             KM  1393
 A( 11,  1) = - R( 10)*C( 11) + R( 45)*C( 21)                      KM  1394
 A( 11,  2) = - R( 11)*C( 11)                                      KM  1395
 A( 11,  3) = +    0.25*R( 22)*C(  8)                              KM  1396
 A( 11,  5) = + R(  9)*C( 24)                                      KM  1397
 A( 11,  6) = +    0.67*R( 38) + R( 39)*C( 24)                     KM  1398
 A( 11,  7) = + R( 34)                                             KM  1399
 A( 11,  8) = +    0.25*R( 22)*C(  3) +    0.40*R( 24)*C( 25)      KM  1400
 A( 11, 11) = - R( 10)*C(  1) - R( 11)*C(  2) -    4.00*R( 12)     KM  1401
*    *C( 11) - R( 40)*C( 15) - R( 48)*C( 28)                       KM  1402
 A( 11, 15) = + R( 41)*C( 24)                                      KM  1403
 A( 11, 15) = - R( 40)*C( 11)                                      KM  1404
 A( 11, 20) = + R( 21)                                             KM  1405
 A( 11, 21) = + R( 45)*C(  1)                                      KM  1406
 A( 11, 23) = + R( 20)                                             KM  1407
 A( 11, 24) = + R(  9)*C(  5) + R( 39)*C(  6) + R( 41)*C( 13)      KM  1408
 A( 11, 25) = +    0.40*R( 24)*C(  8)                              KM  1409
 A( 11, 26) = + R( 31)                                             KM  1410
 A( 11, 28) = - R( 48)*C( 11)                                      KM  1411
 A( 12,  2) = + R( 36)*C( 16)                                      KM  1412
 A( 12, 12) = - R( 37)                                             KM  1413
 A( 12, 16) = + R( 36)*C(  2)                                      KM  1414
 A( 13, 13) = - R( 41)*C( 24) - R( 44)*C( 24)                      KM  1415
 A( 13, 24) = - R( 41)*C( 13) - R( 44)*C( 13)                      KM  1416
 A( 14,  1) = - R( 43)*C( 14)                                      KM  1417
 A( 14, 14) = - R( 43)*C(  1)                                      KM  1418
 A( 14, 22) = + R( 42)*C( 24)                                      KM  1419
 A( 14, 24) = + R( 42)*C( 22)                                      KM  1420
 A( 15,  1) = +    0.15*R( 27)*C( 18) - R( 30)*C( 15) + R( 35)     KM  1421
*    *C( 16)                                                       KM  1422
 A( 15,  7) = + R( 34)                                             KM  1423
 A( 15,  8) = +    0.40*R( 24)*C( 25)                              KM  1424
 A( 15, 11) = - R( 40)*C( 15)                                      KM  1425
 A( 15, 15) = - R( 30)*C(  1) - R( 40)*C( 11) - R( 50)*C( 28)      KM  1426
 A( 15, 16) = + R( 35)*C(  1)                                      KM  1427
 A( 15, 18) = +    0.15*R( 27)*C(  1)                              KM  1428
 A( 15, 25) = +    0.40*R( 24)*C(  8)                              KM  1429
 A( 15, 26) = + R( 29)                                             KM  1430
 A( 15, 28) = - R( 50)*C( 15)                                      KM  1431
 A( 16,  1) = - R( 35)*C( 16)                                      KM  1432
 A( 16,  2) = - R( 36)*C( 16)                                      KM  1433
```

```
A( 16,  3) = +    0.25*R( 22)*C(  8)                                        KM  1434
A( 16,  7) = + R( 33)*C( 24)                                               KM  1435
A( 16,  8) = +    0.25*R( 22)*C(  3)                                        KM  1436
A( 16, 12) = + R( 37)                                                      KM  1437
A( 16, 16) = - R( 35)*C(  1) - R( 36)*C(  2)                               KM  1438
A( 16, 24) = + R( 33)*C(  7)                                               KM  1439
A( 17,  1) = - R( 19)*C( 17)                                               KM  1440
A( 17,  8) = + R( 18)*C( 24)                                               KM  1441
A( 17, 17) = - R( 19)*C(  1) - R( 51)*C( 28)                               KM  1442
A( 17, 24) = + R( 18)*C(  8)                                               KM  1443
A( 17, 28) = - R( 51)*C( 17)                                               KM  1444
A( 18,  1) = - R( 27)*C( 18) - R( 28)*C( 18)                               KM  1445
A( 18,  9) = + R( 26)*C( 24)                                               KM  1446
A( 18, 18) = - R( 27)*C(  1) - R( 28)*C(  1)                               KM  1447
A( 18, 24) = + R( 26)*C(  9)                                               KM  1448
A( 19,  1) = - R( 25)*C( 19)                                               KM  1449
A( 19,  2) = - R( 23)*C( 19)                                               KM  1450
A( 19,  3) = +    0.50*R( 22)*C(  8)                                       KM  1451
A( 19,  8) = +    0.50*R( 22)*C(  3)                                       KM  1452
A( 19, 19) = - R( 23)*C(  2) - R( 25)*C(  1) - R( 49)*C( 28)               KM  1453
A( 19, 28) = - R( 49)*C( 19)                                               KM  1454
A( 20,  2) = + R( 11)*C( 11)                                               KM  1455
A( 20, 11) = + R( 11)*C(  2)                                               KM  1456
A( 20, 20) = - R( 21)                                                      KM  1457
A( 21,  1) = - R( 45)*C( 21)                                               KM  1458
A( 21, 13) = + R( 44)*C( 24)                                               KM  1459
A( 21, 21) = - R( 45)*C(  1)                                               KM  1460
A( 21, 24) = + R( 44)*C( 13)                                               KM  1461
A( 22, 13) = + R( 41)*C( 24)                                               KM  1462
A( 22, 22) = - R( 42)*C( 24)                                               KM  1463
A( 22, 24) = + R( 41)*C( 13) - R( 42)*C( 22)                               KM  1464
A( 23,  1) = + R( 19)*C( 17)                                               KM  1465
A( 23, 17) = + R( 19)*C(  1) + R( 51)*C( 28)                               KM  1466
A( 23, 23) = - R( 20)                                                      KM  1467
A( 23, 28) = + R( 51)*C( 17)                                               KM  1468
A( 24,  1) = - R(  7)*C( 24) + R( 10)*C( 11)                               KM  1469
A( 24,  2) = - R(  8)*C( 24)                                               KM  1470
A( 24,  3) = +    0.50*R( 22)*C(  8)                                       KM  1471
A( 24,  4) = + R(  6)                                                      KM  1472
A( 24,  5) = - R(  9)*C( 24)                                               KM  1473
A( 24,  6) = - R( 39)*C( 24)                                               KM  1474
A( 24,  7) = - R( 33)*C( 24)                                               KM  1475
A( 24,  8) = - R( 18)*C( 24) +    0.50*R( 22)*C(  3)                       KM  1476
A( 24,  9) = - R( 26)*C( 24)                                               KM  1477
A( 24, 11) = + R( 10)*C(  1) + R( 48)*C( 28)                               KM  1478
A( 24, 13) = - R( 41)*C( 24) - R( 44)*C( 24)                               KM  1479
A( 24, 22) = - R( 42)*C( 24)                                               KM  1480
A( 24, 24) = - R(  7)*C(  1) - R(  8)*C(  2) - R(  9)*C(  5) -             KM  1481
*     R( 18)*C(  8) - R( 26)*C(  9) - R( 33)*C(  7) - R( 39)*C(  6)       KM  1482
*      - R( 41)*C( 13) - R( 42)*C( 22) - R( 44)*C( 13) - R( 47)            KM  1483
*     *C( 28)                                                              KM  1484
A( 24, 28) = - R( 47)*C( 24) + R( 48)*C( 11)                               KM  1485
A( 25,  2) = + R(  1)                                                      KM  1486
A( 25,  8) = - R( 24)*C( 25)                                               KM  1487
A( 25, 25) = - R(  2) - R( 24)*C(  8) - R( 46)*C( 28)                      KM  1488
```

```
      A( 25, 28) = - R( 46)*C( 25)                                          KM  1489
      A( 26,  1) = +    0.85*R( 27)*C( 18) + R( 30)*C( 15)                  KM  1490
      A( 26,  2) = - R( 32)*C( 26)                                          KM  1491
      A(26,15) = R(30)*C(1) + R(50)*C(28)                                   KM  1492
      A( 26, 18) = +    0.85*R( 27)*C(  1)                                  KM  1493
      A( 26, 26) = - R( 29) - R( 31) - R( 32)*C(  2)                        KM  1494
      A(26,28) = R(50)*C(15)                                                KM  1495
      A( 27,  2) = + R( 15)*C( 10)                                          KM  1496
      A( 27, 10) = + R( 15)*C(  2)                                          KM  1497
      A( 27, 27) = - R( 16)*C( 30) - R( 17)                                 KM  1498
C     A( 27, 30) = - R( 16)*C( 27)                                          KM  1499
      A( 28, 11) = - R( 48)*C( 28)                                          KM  1500
      A( 28, 15) = - R( 50)*C( 28)                                          KM  1501
      A( 28, 17) = - R( 51)*C( 28)                                          KM  1502
      A( 28, 19) = - R( 49)*C( 28)                                          KM  1503
      A( 28, 24) = - R( 47)*C( 28)                                          KM  1504
      A( 28, 25) = - R( 46)*C( 28)                                          KM  1505
      A( 28, 28) = - R( 46)*C( 25) - R( 47)*C( 24) - R( 48)*C( 11) -        KM  1506
     *    R( 49)*C( 19) - R( 50)*C( 15) - R( 51)*C( 17)                     KM  1507
      A( 29, 11) = + R( 48)*C( 28)                                          KM  1508
      A( 29, 15) = + R( 50)*C( 28)                                          KM  1509
      A( 29, 17) = + R( 51)*C( 28)                                          KM  1510
      A( 29, 19) = + R( 49)*C( 28)                                          KM  1511
      A( 29, 24) = + R( 47)*C( 28)                                          KM  1512
      A( 29, 25) = + R( 46)*C( 28)                                          KM  1513
      A( 29, 28) = + R( 46)*C( 25) + R( 47)*C( 24) + R( 48)*C( 11) +        KM  1514
     *    R( 49)*C( 19) + R( 50)*C( 15) + R( 51)*C( 17)                     KM  1515
C     A( 30,  1) = - R(  4)*C(  2)*C( 30)                                   KM  1516
C     A( 30,  2) = - R(  4)*C(  1)*C( 30)                                   KM  1517
C     A( 30,  4) = +    2.00*R(  5)*C(  4)                                  KM  1518
C     A( 30,  6) = + R( 39)*C( 24)                                          KM  1519
C     A( 30,  7) = + R( 33)*C( 24)                                          KM  1520
C     A( 30,  9) = + R( 26)*C( 24)                                          KM  1521
C     A( 30, 13) = + R( 44)*C( 24)                                          KM  1522
C     A( 30, 24) = + R( 26)*C(  9) + R( 33)*C(  7) + R( 39)*C(  6) +        KM  1523
C    *    R( 44)*C( 13)                                                     KM  1524
C     A( 30, 27) = - R( 16)*C( 30)                                          KM  1525
C     A( 30, 30) = - R(  4)*C(  1)*C(  2) - R( 16)*C( 27)                   KM  1526
      RETURN                                                                KM  1527
      END                                                                   KM  1528



      SUBROUTINE KEMOD2(INTIM,TSTOP,NUMFLX,FLXUNT)                          KM  1529
C                                                                           KM  1530
C     -------  KEMOD2 IS THE SECONDARY DRIVER FOR THE CHEMICAL/             KM  1531
C              DIFFUSION MODULE OF THE ERT TRAJECTORY MODEL.                KM  1532
C                                                                           KM  1533
C              KEMOD2 CONTROLS ALL I/O EXCEPT INITIAL EMISSION INPUTS.      KM  1534
C              KEMOD2 CONTROLS THE UPDATING PROCESSES.                      KM  1535
C                                                                           KM  1536
C     -------  FEATURES OF KEMOD2                                           KM  1537
C                  INTEGRATION BY EPISODE PACKAGE                           KM  1538
C                  VARIABLE SIZE VERTICAL MESH                              KM  1539
C                  GENERALIZED BOUNDARY CONDITIONS                          KM  1540
```

```
C                  TIME + HEIGHT DEPENDENT PHOTODISSOCIATION RATES,      KM  1541
C                   WITH SKY CLEARNESS RATIOS                            KM  1542
C                  TEMPERATURE DEPENDENT REACTION RATE CONSTANTS          KM  1543
C                                                                        KM  1544
      REAL INTIM                                                         KM  1545
      INTEGER BCFLAG                                                     KM  1546
      COMMON/CHEM1/ NOSTAT,              NOSTM1,             NOREAC,     KM  1547
     1              NOSPEC,              NSTDY,              NK          KM  1548
      COMMON/CHEM2/ CONIN(40,5),         WTMOLE(40),         RATKON(55), KM  1549
     1              RATEFF(55),          RATEV(2,5),         QRATE,      KM  1550
     2              NVRATE,              LOCVRT(2)                       KM  1551
      COMMON/CHEM3/ ZEE(5),              DELZ(4),            HTCELL(6),  KM  1552
     1              TDELZ(2) ,           DFINIT(6),          SCALOW(4),  KM  1553
     2              DCOF(5),             FLXWAL(40),         FLXDGE(40), KM  1554
     3              BCFLAG(40),          DPRATE(10),         DEPOWR(10), KM  1555
     4              LOCOPF(10),          NOPFLX ,            SCALUP(4)   KM  1556
      COMMON/CHEM4/ RATK1(100,5),        RATK2(100,5),       RLAT,       KM  1557
     1              RLONG,               TMZONE,             SUNTIM,     KM  1558
     2              HIRATE,              JDATE,              NRATE       KM  1559
      COMMON /CHEM5/  REACT(20,55),SPEC(40),LOCFLX(10),                  KM  1560
     1                NASFLX,FLXW1(30),FLXW2(30)                         KM  1561
      COMMON/INPUTS/TITLE(20),           IDATE(10),          NCURV       KM  1562
      COMMON/FLUXES/FLXIN( 7,200),       FLXTIM(200),        NFLUX       KM  1563
      COMMON /PS1/  TPASS(200),          PS(7,5,75),         FRACT(3),   KM  1564
     1              NPTSR,               NPSFLX,             LOCPSF(7)   KM  1565
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT    KM  1566
      COMMON/EPCOM6/PW(4500)                                             KM  1567
      COMMON/PRPLOT/KPS(5) ,             KSYM(5),            GRCON(5,100)KM  1568
     1     ,      TOUT(100),   VALMAX(5)                                 KM  1569
      COMMON/EPCOM9/HUSED,               NQUSED,             NSTEP,      KM  1570
     1              NFE,                 NJE                             KM  1571
      DIMENSION     YO(150),             PPM (40,5),         TEMP(25),   KM  1572
     1              TMTEMP(25)  ,        HTINV(100,8),       WORK(200),  KM  1573
     2              CLOUDT(25),          CLOUDF(25)         ,VFR(5,200), KM  1574
     3              PTSR(7,200)                                           KM  1575
C                                                                        KM  1576
      EQUIVALENCE (YES,IES),  (RNEG,NEG)                                 KM  1577
      EQUIVALENCE (PPM ,CONIN)                                           KM  1578
      EQUIVALENCE (VFR,PW),  (PTSR,PW(1001))                             KM  1579
C                                                                        KM  1580
      DATA LUP, LIN, LOUT, YES, RNEG  /1, 3, 6, 3HYES, 2HNO/            KM  1581
      DATA KSYM /1HN, 1H2, 1HO, 1HS, 1HC  /                              KM  1582
      DATA RMOLE /4HMOLE/                                                KM  1583
      DATA NROW,MAXR,MAXMSH,MAXNK,NRHT,KTMAX /40,55,5,30,100,100/        KM  1584
      DATA MF,IERROR /21,3/                                              KM  1585
C                                                                        KM  1586
C------------ READ INPUTS FOR COMPUTING POLLUTANT CONCENTRATIONS         KM  1587
C                                                                        KM  1588
      READ(LIN,30) DELT                                                  KM  1589
      READ(LIN,30) BIGSTP                                                KM  1590
      READ(LIN,30) EPS                                                   KM  1591
      READ(LIN,30) UPDINT                                                KM  1592
      READ(LIN,30) PRNTIN                                                KM  1593
      READ(LIN,29) TIMOUT                                                KM  1594
      READ(LIN,29) SUNGEN                                                KM  1595
```

```
      READ(LIN,29) HIRATE                                                KM  1596
      READ(LIN,29) QCLOUD                                                KM  1597
      READ(LIN,29) VTEMP                                                 KM  1598
      READ(LIN,29) SPUNCH                                                KM  1599
      READ(LIN,31) NUMSTP                                                KM  1600
      READ(LIN,31) NOREAC                                                KM  1601
      READ(LIN,31) NOSPEC                                                KM  1602
      READ(LIN,31) NSTDY                                                 KM  1603
      READ(LIN,31) NOSTAT                                                KM  1604
      READ(LIN,30) (ZEE(I), I=1,NOSTAT)                                  KM  1605
C                                                                        KM  1606
      NK = NOSPEC -NSTDY                                                 KM  1607
      NPUNCH = NK                                                        KM  1608
      SUNTIM = UPDINT                                                    KM  1609
      RDCHEM = YES                                                       KM  1610
      QRATE = YES                                                        KM  1611
      F = YES                                                            KM  1612
      TSTOP2 = TSTOP                                                     KM  1613
      KOK = 100                                                          KM  1614
      NASFLX = NUMFLX                                                    KM  1615
C                                                                        KM  1616
C     ---- NOSPEC IS THE TOTAL NUMBER OF SPECIE    (MAX = 40)           KM  1617
C     ---- NSTDY IS THE NUMBER OF SPECIE HELD CONSTANT OR IN STEADY STATKM  1618
C     ---- NK IS THE NUMBER OF SPECIE INTEGRATED    (MAXNK = 30)        KM  1619
C     ---- NOSTAT IS THE NUMBER OF VERTICAL STATIONS (MIN=4,MAX=5)      KM  1620
C     -----LOCFLX IS LOCATION  INDEX FOR AREA SOURCE EMISSION FLUXES     KM  1621
C     -----NUMFLX IS NUMBER OF AREA SOURCE EMISSION FLUXES               KM  1622
C     -----LOCPSF IS LOCATION  INDEX OF POINT SOURCE EMISSION FLUXES     KM  1623
C     -----NPSFLX IS NUMBER OF POINT SOURCE EMMISION FLUXES              KM  1624
C     -----LOCDPF IS LOCATION  INDEX FOR DEPOSITING FLUXES               KM  1625
C     -----NDPFLX IS NUMBER OF DEPOSITING FLUXES                         KM  1626
C                                                                        KM  1627
C     -------------BOUNDARY CONDITION CODE FOR WALL--------------------KM  1628
C                                                                        KM  1629
C     BCFLAG = 0   DC/DZ = PHI/KZ   WHERE   PHI = 0.  (DEFAULT CASE)    KM  1630
C                                   OR      PHI = AREA SOURCE EMISSION FKM  1631
C                                   AND     KZ  = DIFFUSION COEFFICIENT KM  1632
C                                                                        KM  1633
C     BCFLAG = 1   DC/DZ = -DPRATE*(PPM**DEPOWR)/KZ  DEPOSITING SPECIE  KM  1634
C     -----WHERE                                                         KM  1635
C     -----DPRATE(I) IS THE DEPOSITION VELOCITY OF THE I*TH  SPECIE      KM  1636
C     -----IN METERS/MINUTE/(PPM**(DEPOWR-1))                           KM  1637
C     -----INPUT AS POSITIVE QUANTITY FOR UPTAKE AT WALL                 KM  1638
C                                                                        KM  1639
C     BCFLAG = 2    C = INITIAL CONCENTRATION FOR ALL TIME               KM  1640
C                                                                        KM  1641
C     ------------------------------------------------------------------KM  1642
C                                                                        KM  1643
C     ------  ZERO PPM ARRAY  AND CHECK NK AND NOSTAT INPUTS             KM  1644
      NRC = MAXMSH*NROW                                                  KM  1645
      CALL XMIT (-NRC,0.,CONIN)                                          KM  1646
      IF(NOSTAT.GE.4)  GO TO 200                                         KM  1647
      WRITE(LOUT,77) NOSTAT                                              KM  1648
      GO TO 800                                                          KM  1649
  200 CONTINUE                                                           KM  1650
```

```
      IF(NK.LE.MAXNK)  GO TO 210                                        KM  1651
      WRITE(LOUT,76) NOSPEC ,NSTDY, NK                                  KM  1652
      GO TO 800                                                         KM  1653
  210 CONTINUE                                                          KM  1654
      IF(RDCHEM.NE.YES)  GO TO 219                                      KM  1655
      NOPFLX = 0                                                        KM  1656
      DO 215  I=1,NOSPEC                                                KM  1657
      READ(LIN,51)  SPEC(I),WTMOLE(I),BCFLAG(I)                         KM  1658
      IF(BCFLAG(I) .EQ. 1)  NOPFLX = NOPFLX + 1                         KM  1659
  215 CONTINUE                                                          KM  1660
      IF(NUMFLX.GT.0) READ(LIN,31) (LOCFLX(I),I=1,NUMFLX)               KM  1661
      IF(NPSFLX.GT.0) READ(LIN,31) (LOCPSF(I),I=1,NPSFLX)               KM  1662
      IF(NOPFLX.GT.0) READ(LIN,53) (LOCOPF(I),OPRATE(I),DEPOWR(I),      KM  1663
     1  I=1,NOPFLX)                                                     KM  1664
  219 CONTINUE                                                          KM  1665
      DO 220 I = 1,NOSPEC                                               KM  1666
      READ(LIN,41) (PPM(I,J),J=1,NOSTAT)                                KM  1667
220   CONTINUE                                                          KM  1668
      IF(RDCHEM.NE.YES)  GO TO 230                                      KM  1669
      DO 222 I=1,NOREAC                                                 KM  1670
      READ(LIN,30) RATKON(I)                                            KM  1671
222   CONTINUE                                                          KM  1672
      DO 224 J=1,NOREAC                                                 KM  1673
      READ(LIN,40) (REACT(I,J),I=1,20)                                  KM  1674
224   CONTINUE                                                          KM  1675
C                                                                       KM  1676
230   CONTINUE                                                          KM  1677
C                                                                       KM  1678
      TIME = INTIM                                                      KM  1679
C                                                                       KM  1680
C                                                                       KM  1681
C     -------- READ OR GENERATE THE PHOTODISSOCIATION RATES             KM  1682
C                                                                       KM  1683
C     ------- THE PHOTODISSOCIATION RATES ARE GENERATED IF SUNGEN=YES    KM  1684
C     ------- THE PHOTODISSOCIATION RATES VARY WITH HEIGHT IF HIRATE=YESKM  1685
C     ------- LOCVRT IS THE SPECIES INDEX FOR THE RATES (INPUT)         KM  1686
C                                                                       KM  1687
      READ(LIN,30) RLAT                                                 KM  1688
      READ(LIN,30) RLONG                                                KM  1689
      READ(LIN,30) TMZONE                                               KM  1690
      READ(LIN,31) JDATE                                                KM  1691
      READ(LIN,31)  NVRATE                                              KM  1692
      READ(LIN,31) (LOCVRT(I),I=1,NVRATE)                               KM  1693
C                                                                       KM  1694
      IF(SUNGEN.EQ.YES)  GO TO 244                                      KM  1695
      DO 243 KK=1,NVRATE                                                KM  1696
      NHT=NOSTAT                                                        KM  1697
      IF(HIRATE.EQ.RNEG)  NHT=1                                         KM  1698
      DO 240  I=1,101                                                   KM  1699
      IF(KK.GT.1)  GO TO 236                                            KM  1700
      READ(LIN,54) (RATK1(I,IZ),IZ=1,NHT)                               KM  1701
      IF(RATK1(I,1) .LT.  0.0)  GO TO 243                               KM  1702
      GO TO 240                                                         KM  1703
  236 IF(KK.GT.2)  GO TO 242                                            KM  1704
      READ(LIN,54) (RATK2(I,IZ),IZ=1,NHT)                               KM  1705
```

```
      IF(RATK2(I,1) .LT.  0.0)  GO TO 243                              KM  1706
  240 CONTINUE                                                         KM  1707
  242 WRITE(LOUT,15)                                                   KM  1708
      GO TO 800                                                        KM  1709
  243 CONTINUE                                                         KM  1710
      NRATE = I - 1                                                    KM  1711
      GO TO 245                                                        KM  1712
  244 CONTINUE                                                         KM  1713
C                                                                      KM  1714
      JSTART = ITHOUR( INTIM )                                         KM  1715
      JSTOP  = ITHOUR( FLXTIM(NFLUX) )                                 KM  1716
      CALL PHOTOD (JSTART,JSTOP,NOSTAT,ZEE)                            KM  1717
  245 CONTINUE                                                         KM  1718
      IK1 = 1                                                          KM  1719
C                                                                      KM  1720
      IF(QCLOUD.EQ.RNEG)  GO TO 249                                    KM  1721
C     ------- INPUT SKY COVER FACTORS AND TIMES                        KM  1722
C     ------- LAST SKY COVER UPDATE TIME (CLOUDT) MUST BE A NEGATIVE   KM  1723
      DO 246  I=1,26                                                   KM  1724
      READ(LIN,59)  A, B                                               KM  1725
      IF(A.LT.0.0)  GO TO 248                                          KM  1726
      IF(I.GT.25)  GO TO 247                                           KM  1727
      CLOUDT(I) = A                                                    KM  1728
      CLOUDF(I) = B                                                    KM  1729
      NCLOUD = I                                                       KM  1730
  246 CONTINUE                                                         KM  1731
      GO TO 248                                                        KM  1732
  247 WRITE(LOUT,57)                                                   KM  1733
      GO TO 800                                                        KM  1734
  248 CONTINUE                                                         KM  1735
      ICLOUD = 1                                                       KM  1736
      CLOUDY = CLOUDF(1)                                               KM  1737
  249 IF(QCLOUD.EQ.RNEG)  CLOUDY= 1.0                                  KM  1738
C     ------      THIS CALL TO UPRAT2 INITIALIZES THE VARIABLE RATES   KM  1739
      CALL UPRAT2(TIME,IK1,RATEV,NOSTAT,NVRATE,CLOUDY)                 KM  1740
      CALL RATEHI(1)                                                   KM  1741
260   CONTINUE                                                         KM  1742
C                                                                      KM  1743
C     ---------- READ TEMPERATURES IF VTEMP = YES                      KM  1744
      ITEMP = 1                                                        KM  1745
      NTEMP = 1                                                        KM  1746
      TEMP(1) = 298.2                                                  KM  1747
      IF(VTEMP.NE.YES)  GO TO 265                                      KM  1748
      DO 261 I = 1,26                                                  KM  1749
      READ(LIN,59)  A, B                                               KM  1750
      IF(I.GT.25)   GO TO 262                                          KM  1751
      IF(A.LT.0.)  GO TO 265                                           KM  1752
      TMTEMP(I) = A                                                    KM  1753
      TEMP(I) = B + 273.2                                              KM  1754
      NTEMP = I                                                        KM  1755
  261 CONTINUE                                                         KM  1756
  262 CONTINUE                                                         KM  1757
      WRITE(LOUT,79)                                                   KM  1758
      GO TO 800                                                        KM  1759
  265 CONTINUE                                                         KM  1760
```

```
      CALL TEMPR (ITEMP,INTIM,TEMP)                                     KM  1761
C                                                                       KM  1762
C     -------- READ TABLE OF DIFFUSIVITY UPDATE TIMES, MIXING HEIGHTS,  KM  1763
C              AND DIFFUSIVITY COEFFICIENTS.                            KM  1764
C              MIXING HEIGHT INFORMATION IS OPTIONAL.                   KM  1765
      K = NOSTAT+3                                                      KM  1766
      DO 270 I = 1,101                                                  KM  1767
      READ(LIN,41) A,B,(HTINV(I,J),J=3,K)                               KM  1768
      IF (A.LT.0.) GO TO 280                                            KM  1769
      HTINV(I,1) = A                                                    KM  1770
      HTINV(I,2) = B                                                    KM  1771
270   CONTINUE                                                          KM  1772
      PRINT 45                                                          KM  1773
      GO TO 800                                                         KM  1774
280   NHT = I-1                                                         KM  1775
      IHT = 1                                                           KM  1776
      KHT = 1                                                           KM  1777
290   CONTINUE                                                          KM  1778
C                                                                       KM  1779
C     INITIALIZE PLOT PARAMETERS                                        KM  1780
C      FOR                                                              KM  1781
      KPS(1) = 1                                                        KM  1782
      KPS(2) = 2                                                        KM  1783
      KPS(3) = 3                                                        KM  1784
      KPS(4) = 28                                                       KM  1785
      KPS(5) = 5                                                        KM  1786
      NCURV = 5                                                         KM  1787
      CALL XMIT(-NCURV,0,0,VALMAX)                                      KM  1788
C                                                                       KM  1789
C      BANNER PAGE                                                      KM  1790
C                                                                       KM  1791
      WRITE(LOUT,99)                                                    KM  1792
C                                                                       KM  1793
C     ----- REGULATE THE AREA SOURCE EMISSION FLUX SCHEDULES            KM  1794
      IF(NUMFLX.EQ.0)  GO TO 299                                        KM  1795
      DO 297 I=1,NUMFLX                                                 KM  1796
      DO 293 J=1,NFLUX                                                  KM  1797
      WORK(J) = FLXIN(I,J)                                              KM  1798
  293 CONTINUE                                                          KM  1799
      CALL SKEDUL(WORK,FLXTIM,NFLUX,NUPDAT,UPDINT,INTIM,TSTOP,-1.,F,-1) KM  1800
      IF(F.EQ. YES)  GO TO 800                                          KM  1801
C     -----   CONVERT FLUXES FROM MASS OR MOLE FRACTION TO PPM          KM  1802
      ID = LOCFLX(I)                                                    KM  1803
      WT = 1.E6                                                         KM  1804
      IF(FLXUNT.NE.RMOLE)   WT = 28.97E6 /WTMOLE(ID)                    KM  1805
      DO 295 J=1,NUPDAT                                                 KM  1806
      FLXIN(I,J) = WORK(J)*WT                                           KM  1807
  295 CONTINUE                                                          KM  1808
  297 CONTINUE                                                          KM  1809
C     ----------  THE NEW SCHEDULE OF FLUXES IS STORED IN FLXIN         KM  1810
      NFLUX = NUPDAT                                                    KM  1811
      IFLXTM = 1                                                        KM  1812
  299 CONTINUE                                                          KM  1813
C     ------- INITIALIZE FLUXES                                         KM  1814
      NPMX = 5*MAXNK                                                    KM  1815
```

```
      CALL  XMIT(-NPMX,0,0,PSR1)                                        KM  1816
      CALL  XMIT(-NPMX,0,0,PSR2)                                        KM  1817
      CALL  XMIT(-MAXNK,0,0,FLXW1)                                      KM  1818
      CALL  XMIT(-MAXNK,0,0,FLXW2)                                      KM  1819
      CALL XMIT(-NROW,0.,FLXWAL)                                        KM  1820
      CALL XMIT(-NROW,0.,FLXOGE)                                        KM  1821
      CALL UPFLX1(TIME,IFLXTM)                                          KM  1822
C                                                                       KM  1823
C     *****   GENERATE VERTICAL MESH PARAMETERS                         KM  1824
C                                                                       KM  1825
      NOSTM1 = NOSTAT - 1                                               KM  1826
      DELZ(1) = ZEE(2) - ZEE(1)                                         KM  1827
      HTCELL(1) = DELZ(1)                                               KM  1828
      DO 300 K = 2,NOSTM1                                               KM  1829
      DELZ(K) = ZEE(K+1) - ZEE(K)                                       KM  1830
      HTCELL(K) = .5*(DELZ(K) + DELZ(K-1))                              KM  1831
300   CONTINUE                                                          KM  1832
      HTCELL(NOSTAT) = DELZ(NOSTM1)                                     KM  1833
C                                                                       KM  1834
C      ***** CONVERT POINT SOURCE EMISSIONS FROM ABSOLUTE MASS           KM  1835
C            OR MOLES TO RATES ON A REGULAR UPDATE SCHEDULE.            KM  1836
C                                                                       KM  1837
C      ***** POINT SOURCE MASS IS BLED IN OVER (UP TO) THREE            KM  1838
C            UPDATE INTERVALS. THE FRACTION IN EACH INTERVAL            KM  1839
C            IS SET BY THE FRACT ARRAY (SEE KEMOD DATA STATEMENT).      KKM 1840
C                                                                       KM  1841
      IF(NPSFLX.LE.0)  GO TO 307                                        KM  1842
      CALL XMIT(-2625,0,0,PS)                                           KM  1843
      A = 1.E6/UPDINT                                                   KM  1844
      IF(FLXUNT.EQ.RMOLE)  GO TO 303                                    KM  1845
      DO 302 K = 1,NPSFLX                                               KM  1846
      ID =LOCPSF(K)                                                     KM  1847
      WORK(K) = 28.97/WTMOLE(ID)                                        KM  1848
302   CONTINUE                                                          KM  1849
303   CONTINUE                                                          KM  1850
      B = INTIM - 0.499*UPDINT                                          KM  1851
      K4 = 0                                                            KM  1852
      DO 305 K = 1,NPTSR                                                KM  1853
      K1 = IFIX( (TPASS(K)-B)/UPDINT) + 1                               KM  1854
      K2 = K1 + 1                                                       KM  1855
      K3 = K1 + 2                                                       KM  1856
      K4 = MAX0(K4,K3)                                                  KM  1857
      DO 305 J = 1,NOSTAT                                               KM  1858
      VF = A*VFR(J,K)/HTCELL(J)                                         KM  1859
      IF(J.EQ.1  .OR.  J.EQ.NOSTAT)  VF = 2.*VF                         KM  1860
      DO 305 I = 1,NPSFLX                                               KM  1861
      C = VF*PTSR(I,K)                                                  KM  1862
      IF(FLXUNT.NE.RMOLE) C = C*WORK(I)                                 KM  1863
      PS(I,J,K1) = PS(I,J,K1) + FRACT(1)*C                              KM  1864
      PS(I,J,K2) = PS(I,J,K2) + FRACT(2)*C                              KM  1865
      PS(I,J,K3) = PS(I,J,K3) + FRACT(3)*C                              KM  1866
305   CONTINUE                                                          KM  1867
      NPTSR = K4                                                        KM  1868
      TPASS(1) = INTIM                                                  KM  1869
      DO 306 K = 2,NPTSR                                                KM  1870
```

```
      TPASS(K) = TPASS(K-1) + UPDINT                                    KM  1871
306   CONTINUE                                                          KM  1872
      N = MAXNK*NOSTAT                                                  KM  1873
      CALL XMIT(-N,0,0,PSRATE)                                          KM  1874
      CALL XMIT(-2400,0,0,PW)                                           KM  1875
307   IPS = 1                                                           KM  1876
      CALL UPSORC(TIME,IPS,NOSTAT,SPEC)                                 KM  1877
C                                                                       KM  1878
C     ------- PRINT INPUTS                                            KM  1879
C                                                                       KM  1880
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  1881
      WRITE(LOUT,16) INTIM,TSTOP,DELT,BIGSTP,NOREAC,NOSPEC,NSTDY,NOSTAT KM  1882
      IF(NUMFLX.EQ.0)  GO TO 311                                        KM  1883
      DO 308 I= 1,NUMFLX                                                KM  1884
      ID = LOCFLX(I)                                                    KM  1885
      WORK(I) = SPEC(ID)                                                KM  1886
308   CONTINUE                                                          KM  1887
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  1888
      WRITE(LOUT,68)  (WORK(I) , I=1,NUMFLX)                            KM  1889
      DO 310  J = 1, NFLUX                                              KM  1890
      IF(MOD(J,50).NE.0)  GO TO 309                                     KM  1891
      CALL NEWPAG (TITLE,0,IDATE)                                       KM  1892
      WRITE(LOUT,68)  (WORK(I), I=1,NUMFLX)                             KM  1893
309   WRITE(LOUT,69) FLXTIM(J), (FLXIN(I,J), I=1,NUMFLX)                KM  1894
      FLXTIM(J+1)=FLXTIM(J) + UPDINT                                    KM  1895
310   CONTINUE                                                          KM  1896
311   CONTINUE                                                          KM  1897
      IF(NPSFLX.EQ.0)  GO TO 317                                        KM  1898
      DO 312  K=1,NPSFLX                                                KM  1899
      ID = LOCPSF(K)                                                    KM  1900
      WORK(K) = SPEC(ID)                                                KM  1901
312   CONTINUE                                                          KM  1902
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  1903
      WRITE(LOUT,83)  (WORK(K),K=1,NPSFLX)                              KM  1904
      DO 314 K = 1,NPTSR                                                KM  1905
      IF(MOD(K,9).NE.0)  GO TO 313                                      KM  1906
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  1907
      WRITE(LOUT,83)  (WORK(KK),KK=1,NPSFLX)                            KM  1908
313   WRITE(LOUT,84)  TPASS(K), (PS(I,1,K),I=1,NPSFLX)                  KM  1909
      DO 314 J =2,NOSTAT                                                KM  1910
      WRITE(LOUT,85)  (PS(I,J,K),I=1,NPSFLX)                            KM  1911
314   CONTINUE                                                          KM  1912
317   CONTINUE                                                          KM  1913
C                                                                       KM  1914
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  1915
      WRITE(LOUT,37)                                                    KM  1916
      DO 320 I=1,NOREAC                                                 KM  1917
319   WRITE(LOUT,38) I, (REACT(J,I),J=1,20), RATKON(I)                  KM  1918
320   CONTINUE                                                          KM  1919
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  1920
      WRITE(LOUT,28)                                                    KM  1921
      DO 330 J = 1,NOSPEC                                               KM  1922
 325  WRITE(LOUT,27)  J,  SPEC(J), WTMOLE(J), BCFLAG(J)                 KM  1923
 330  CONTINUE                                                          KM  1924
C                                                                       KM  1925
```

```
C      ------  SCHEDULE AND INITIALIZE DIFFUSIVITIES                    KM  1926
C                                                                       KM  1927
       IF(NHT.EQ.1)   GO TO 360                                         KM  1928
       K = NOSTAT + 3                                                   KM  1929
       TSTOP = HTINV(NHT,1)                                             KM  1930
       CALL XMIT (NHT,HTINV(1,1),WORK(1))                               KM  1931
       DO 350 J =2,K                                                    KM  1932
       CALL SKEDUL(HTINV(1,J),WORK,NHT,NINT,UPDINT,INTIM,TSTOP,-1.,F,1) KM  1933
       IF(F.EQ.YES)  GO TO 800                                          KM  1934
350    CONTINUE                                                         KM  1935
       NHT = NINT                                                       KM  1936
       HTINV(1,1) = INTIM                                               KM  1937
       DO 355 I =2,NHT                                                  KM  1938
       HTINV(I,1) = HTINV(I-1,1) + UPDINT                               KM  1939
355    CONTINUE                                                         KM  1940
       TSTOP = TSTOP2                                                   KM  1941
360    CONTINUE                                                         KM  1942
       K = NOSTAT + 1                                                   KM  1943
       DO 370 I =1,K                                                    KM  1944
       J = I + 2                                                        KM  1945
       DFINIT(I) = HTINV(1,J)                                           KM  1946
370    CONTINUE                                                         KM  1947
       CALL DIFCOF(NOSTAT)                                              KM  1948
C       ----    INITIALIZE RATEFF                                       KM  1949
       CALL XMIT(NOREAC,RATKON,RATEFF)                                  KM  1950
C                                                                       KM  1951
410    IPROD = NK*NOSTAT                                                KM  1952
       DELZ(1) = 2./HTCELL(1)                                           KM  1953
       DELZ(2) = 2./HTCELL(NOSTAT)                                      KM  1954
       CALL NEWPAG(TITLE,0,IDATE)                                       KM  1955
       WRITE(LOUT,14)                                                   KM  1956
       HTCELL(1) = HTCELL(1)/2.                                         KM  1957
       HTCELL(NOSTAT) = HTCELL(NOSTAT)/2.                               KM  1958
       I = 1                                                            KM  1959
       WRITE(LOUT,44) I,ZEE(I),DFINIT(I),HTCELL(I)                      KM  1960
       DO 420 I = 2,NOSTAT                                              KM  1961
       WRITE(LOUT,43) DFINIT(I), DELZ(I-1)                              KM  1962
       IF(I.EQ.NOSTAT)  GO TO 415                                       KM  1963
       WRITE(LOUT,42)  I, ZEE(I), HTCELL(I)                             KM  1964
       GO TO 420                                                        KM  1965
415    WRITE(LOUT,44) I,ZEE(I),DFINIT(I),HTCELL(I)                      KM  1966
420    CONTINUE                                                         KM  1967
       HTCELL(1) = HTCELL(1)*2.                                         KM  1968
       HTCELL(NOSTAT) = HTCELL(NOSTAT)*2.                               KM  1969
C                                                                       KM  1970
C      ----- INITIALIZE INTERNALLY COMPUTED CONCENTRATIONS              KM  1971
       CALL ISTATE                                                      KM  1972
C      ----------  INITIALIZE YO VECTOR OF CONCENTRATIONS FOR DRIVE      KM  1973
       DO 425 I=1,NOSTAT                                                KM  1974
       K= NK*(I-1) + 1                                                  KM  1975
       CALL XMIT(NK,CONIN(1,I),YO(K))                                   KM  1976
  425 CONTINUE                                                          KM  1977
       CALL NEWPAG(TITLE,0,IDATE)                                       KM  1978
       ITIM = ITHOUR(TIME)                                              KM  1979
       WRITE(LOUT,97)   INTIM, ITIM                                     KM  1980
```

```
      WRITE(LOUT,18)  (SPEC(J),J=1,NOSPEC)                                KM  1981
      DO 450 K=1,NOSTAT                                                   KM  1982
      WRITE(LOUT,32)  ZEE(K), ( PPM(J,K),J=1,NOSPEC)                      KM  1983
450   CONTINUE                                                            KM  1984
C                                                                         KM  1985
      IF(SPUNCH,NE,YES)  GO TO 454                                        KM  1986
C     ------  WRITE I.C. ON TAPE LUP FOR PUNCH OR FILE STORAGE            KM  1987
      WRITE(LUP,80) TITLE                                                 KM  1988
      WRITE(LUP,81) TIME,  (ZEE(K),K=1,NOSTAT)                            KM  1989
      NOELV = 1                                                           KM  1990
      DO 453 K=1,NOELV                                                    KM  1991
      WRITE(LUP,81) ( PPM(J,K),J=1,NPUNCH)                                KM  1992
453   CONTINUE                                                            KM  1993
454   CONTINUE                                                            KM  1994
      IF(NCURV.EQ.0) GO TO 459                                            KM  1995
      KT = 1                                                              KM  1996
      DO 455 I = 1,NCURV                                                  KM  1997
      J = KPS(I)                                                          KM  1998
      IF(J.NE.5)  GO TO 456                                               KM  1999
C     SCALE CO FOR PLOTTING                                               KM  2000
      GRCON(I,KT) = .10*PPM(J,1)                                          KM  2001
      IF(GRCON(I,KT).GT.VALMAX(I)) VALMAX(I)=GRCON(I,KT)                  KM  2002
      GO TO 455                                                           KM  2003
456   CONTINUE                                                            KM  2004
      GRCON(I,KT) = PPM(J,1)                                              KM  2005
      IF(PPM(J,1).GT.VALMAX(I)) VALMAX(I) = PPM(J,1)                      KM  2006
  455 CONTINUE                                                            KM  2007
      TOUT(KT) = INTIM                                                    KM  2008
  459 CONTINUE                                                            KM  2009
C     ----------   INITIALIZE ADDITIONAL PARAMETERS FOR INTEGRATION        KM  2010
      NSTEP = 0                                                           KM  2011
      N = IPROD                                                           KM  2012
      KNTER = 0                                                           KM  2013
      TPRINT = INTIM + PRNTIN -.01                                        KM  2014
      RECALL = YES                                                        KM  2015
      UPD4 = UPDINT/4.0                                                   KM  2016
C                                                                         KM  2017
C     ------------ THE INTEGRATION TIME CYCLE BEGINS HERE ------------   KM  2018
C                                                                         KM  2019
      CALL TIMEX(0,A,B)                                                   KM  2020
  460 CONTINUE                                                            KM  2021
      OLDTIM = TIME                                                       KM  2022
      IF(RECALL .NE. YES)  GO TO 470                                      KM  2023
      INDEX = 1                                                           KM  2024
      TOUTEP = OLDTIM + UPDINT/10.                                        KM  2025
      IF(KNTER .GT.1)   DELT = .01                                        KM  2026
  470 IF(NSTEP .GT. NUMSTP)  GO TO 800                                    KM  2027
      TLAST = TIME                                                        KM  2028
C                                                                         KM  2029
C                     INTEGRATE BY EPISODE                                KM  2030
C                                                                         KM  2031
      CALL DRIVE(N,TIME,DELT,YO,TOUTEP,EPS,IERROR,MF,INDEX,BIGSTP,KOK)    KM  2032
C                                                                         KM  2033
C                                                                         KM  2034
      IF(KOK.GE.0)  GO TO 475                                             KM  2035
```

```
      ISTOP = IES                                                       KM  2036
      WRITE(LOUT,60)                                                    KM  2037
      GO TO 480                                                         KM  2038
475   IF(INDEX .LT. 0)  GO TO 800                                       KM  2039
      INDEX = 2                                                         KM  2040
      KNTER = KNTER + 1                                                 KM  2041
      IF(RECALL.NE. YES)   GO TO 480                                    KM  2042
C     MAKE SHORT CALL TO EPISODE (DRIVE) IF RECALL EQUALS YES           KM  2043
C     THEN RESET TOUTEP TO ACTUAL TOUT ON NEXT CALL                     KM  2044
      TOUTEP = OLDTIM + UPDINT                                          KM  2045
      RECALL = RNEG                                                     KM  2046
      GO TO 470                                                         KM  2047
480   TOUTEP = OLDTIM + 2.*UPDINT                                       KM  2048
C     ---------- UNPACK YO FOR CHECKING AND OUTPUT                      KM  2049
      DO 500 I=1,NOSTAT                                                 KM  2050
      K = NK*(I-1) + 1                                                  KM  2051
      CALL XMIT(NK,YO(K),CONIN(1,I)  )                                  KM  2052
500   CONTINUE                                                          KM  2053
```

```
620   CONTINUE                                                          KM  2058
      CALL PRODUK(UPDINT)                                               KM  2059
C                                                                       KM  2060
      IF(TIME.LT.TPRINT)  GO TO 694                                     KM  2061
      TPRINT = TPRINT + PRNTIN                                          KM  2062
 650  CONTINUE                                                          KM  2063
C                                                                       KM  2064
C     ------- OUTPUT                                                    KM  2065
C                                                                       KM  2066
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  2067
      IF(NDPFLX. EQ. 0)  GO TO 682                                      KM  2068
      DO 681 J=1,NDPFLX                                                 KM  2069
      I= LOCDPF(J)                                                      KM  2070
      SINK = -DPRATE(J)*(PPM(I,1)**DEPOWR(J))                           KM  2071
 681  WRITE(10,88)  SPEC(I),  SINK                                      KM  2072
 682  CONTINUE                                                          KM  2073
      ITIM = ITHOUR(TIME)                                               KM  2074
      WRITE(LOUT,91)  TIME, ITIM, RATEV(1,1), DFINIT(1)                 KM  2075
      IF(NUMFLX.LE.0)  GO TO 656                                        KM  2076
      DO 655 I = 1,NUMFLX                                               KM  2077
      K = LOCFLX(I)                                                     KM  2078
      WORK(I) = SPEC(K)                                                 KM  2079
      WORK(I+10) = FLXWAL(K)*TDELZ(1)                                   KM  2080
655   CONTINUE                                                          KM  2081
656   IF(NPSFLX.LE.0)  GO TO 661                                        KM  2082
      DO 660 I = 1,NPSFLX                                               KM  2083
      K = LOCPSF(I)                                                     KM  2084
      WORK(I+20) = SPEC(K)                                              KM  2085
      WORK(I+30) = PSRATE(K,1)                                          KM  2086
660   CONTINUE                                                          KM  2087
661   CONTINUE                                                          KM  2088
      NMX = MAX0(NUMFLX,NPSFLX)                                         KM  2089
      NMV = NMX + 1                                                     KM  2090
```

```
      NMX = NMX + 2                                                    KM  2091
      K = 2                                                            KM  2092
      DO 662 I = NMV,NMX                                               KM  2093
      WORK(I) = SPEC(K)                                                KM  2094
      WORK(I+10) = FLXWAL(K)*TDELZ(1)                                  KM  2095
      WORK(I+20) = SPEC(K)                                             KM  2096
      WORK(I+30) = PSRATE(K,1)                                         KM  2097
      K = 7                                                            KM  2098
662   CONTINUE                                                         KM  2099
      WRITE(LOUT,92)                                                   KM  2100
      WRITE(LOUT,93)  (WORK(I),WORK(I+10)                              KM  2101
     1 ,              WORK(I+20), WORK(I+30),I=1,NMX)                  KM  2102
      WRITE(LOUT,94)                                                   KM  2103
      WRITE(LOUT,9) (SPEC(J),J=1,NOSPEC)                               KM  2104
      DO 690 K = 1,NOSTAT                                              KM  2105
      WRITE(LOUT,32) ZEE(K) , (PPM(J,K),J=1,NOSPEC)                    KM  2106
  690 CONTINUE                                                         KM  2107
      IF (TIMOUT.NE.YES)  GO TO 691                                    KM  2108
      WRITE(LOUT,66)  INDEX,HUSED,NQUSED,NSTEP,NFE,NJE                 KM  2109
      NTSTP = NSTEP                                                    KM  2110
      CALL TIMEX(NTSTP,A,B)                                            KM  2111
  691 CONTINUE                                                         KM  2112
      IF(SPUNCH.NE.YES)  GO TO 694                                     KM  2113
C     ------  WRITE OUTPUT ON TAPE LUP FOR PUNCH OR FILE STORAGE       KM  2114
      WRITE(LUP,81) TIME                                               KM  2115
      DO 693 K=1,NOELV                                                 KM  2116
      WRITE(LUP,81) (PPM(J,K),J=1,NPUNCH)                              KM  2117
  693 CONTINUE                                                         KM  2118
  694 CONTINUE                                                         KM  2119
C                                                                      KM  2120
      IF(NCURV.EQ.0) GO TO 700                                         KM  2121
      KT = KT + 1                                                      KM  2122
      IF(KT.GT.KTMAX) GO TO 700                                        KM  2123
      DO 695 I = 1,NCURV                                               KM  2124
      J = KPS(I)                                                       KM  2125
      IF(J.NE.5)  GO TO 696                                            KM  2126
C     SCALE CO FOR PLOTTING                                            KM  2127
      GRCON(I,KT) = .10*PPM(J,1)                                       KM  2128
      IF(GRCON(I,KT).GT.VALMAX(I)) VALMAX(I)=GRCON(I,KT)               KM  2129
      GO TO 695                                                        KM  2130
696   CONTINUE                                                         KM  2131
      GRCON(I,KT) = PPM(J,1)                                           KM  2132
      IF(PPM(J,1).GT.VALMAX(I)) VALMAX(I) = PPM(J,1)                   KM  2133
  695 CONTINUE                                                         KM  2134
      TOUT(KT) = TIME                                                  KM  2135
  700 CONTINUE                                                         KM  2136
      IF(TIME.GE.(TSTOP-.01)) ISTOP = IES                              KM  2137
      IF(ISTOP.EQ. IES)  GO TO 800                                     KM  2138
C                                                                      KM  2139
C     ------------ UPDATE EMISSION FLUXES                              KM  2140
      IFLXTM = IFLXTM+1                                                KM  2141
      IF (IFLXTM.GT.NFLUX) IFLXTM = NFLUX                              KM  2142
      CALL UPFLX1(TIME,IFLXTM)                                         KM  2143
      IPS = IPS + 1                                                    KM  2144
      CALL UPSORC(TIME,IPS,NOSTAT,SPEC)                                KM  2145
```

```
720   CONTINUE                                                          KM  2146
C                                                                       KM  2147
C     ******* UPDATE SKY COVER FACTOR                                   KM  2148
      IF (RRATE.EQ.RNEG) GO TO 730                                      KM  2149
      IF(RCLOUD.EQ.RNEG)  GO TO 725                                     KM  2150
      IF(ICLOUD.GE.NCLOUD)  GO TO 725                                   KM  2151
      BB= CLOUDT(ICLOUD+1)  - UPD4                                      KM  2152
      IF(TIME.LT. BB)  GO TO 725                                        KM  2153
      ICLOUD = ICLOUD + 1                                               KM  2154
      CLOUDY = CLOUDF(ICLOUD)                                           KM  2155
      WRITE(10,75)  TIME ,  CLOUDY                                      KM  2156
725   CONTINUE                                                          KM  2157
C                                                                       KM  2158
C     ********** UPDATE THE VARIABLE PHOTODISSOCIATION RATE CONSTANTS   KM  2159
C                                                                       KM  2160
      IK1 = IK1+1                                                       KM  2161
      IF (IK1.GT.NRATE) IK1 = NRATE                                     KM  2162
      ROLDK1 = RATEV(1,1)                                               KM  2163
      CALL UPRAT2(TIME,IK1,RATEV,NOSTAT,NVRATE,CLOUDY)                  KM  2164
      IF(RATEV(1,1).LT..2*ROLDK1.OR.RATEV(1,1).GT.5.*ROLDK1)RECALL=YES  KM  2165
730   CONTINUE                                                          KM  2166
C                                                                       KM  2167
C     ****          UPDATE DIFFUSION COEFFICIENTS                     KM  2168
C                                                                       KM  2169
      IF(NHT.EQ.1)  GO TO 770                                           KM  2170
      AA= HTINV(IHT,1) - UPD4                                           KM  2171
      IF(TIME.LT.AA)  GO TO 770                                         KM  2172
      IF(KHT.GE.NHT)  GO TO 770                                         KM  2173
      IHT = IHT + 1                                                     KM  2174
      KHT = KHT + 1                                                     KM  2175
      WRITE (11,46) TIME,HTINV(IHT,1),HTINV(IHT,2)                      KM  2176
      K = NOSTAT+1                                                      KM  2177
      DO 750 I = 1,K                                                    KM  2178
      DFINIT(I) = HTINV(IHT,I+2)                                        KM  2179
750   CONTINUE                                                          KM  2180
      WRITE(11,96) (DFINIT(I),I=1,K)                                    KM  2181
      CALL DIFCOF(NOSTAT)                                               KM  2182
      IF (IHT.GT.NHT) IHT = NHT                                         KM  2183
770   CONTINUE                                                          KM  2184
C                                                                       KM  2185
C      *****  UPDATE TEMPERATURE DEPENDENT REACTION RATES               KM  2186
      IF(VTEMP.NE.YES)  GO TO 780                                       KM  2187
      IF(ITEMP.GE.NTEMP)  GO TO 780                                     KM  2188
      IP1 = ITEMP + 1                                                   KM  2189
      BB = TMTEMP(IP1) - UPD4                                           KM  2190
      IF(TIME.LT.BB)  GO TO 780                                         KM  2191
      ITEMP = IP1                                                       KM  2192
      CALL TEMPR(ITEMP,TIME,TEMP)                                       KM  2193
780   CONTINUE                                                          KM  2194
C                                                                       KM  2195
      GO TO 460                                                         KM  2196
C                                                                       KM  2197
C     ************  EXIT FROM INTEGRATION TIME CYCLE ***************** KM  2198
C                                                                       KM  2199
800   CONTINUE                                                          KM  2200
```

```
      IF(F.EQ.YES)  RETURN                                                  KM  2201
      IF(INDEX.LT.0)  WRITE(LOUT,74) TIME, INDEX                            KM  2202
      IF(NCURV.EQ.0) GO TO 805                                              KM  2203
      IF(KT.GT.KTMAX) KT = KTMAX                                            KM  2204
      CALL COPLOT(KT)                                                       KM  2205
805   CONTINUE                                                              KM  2206
      IF(NSTEP .GT. NUMSTP)  WRITE(LOUT,263)                                KM  2207
      RETURN                                                                KM  2208
C                                                                           KM  2209
C     FORMATS                                                               KM  2210
C                                                                           KM  2211
8     FORMAT (1H ,5X,A4,8H FLUX = ,2X,G15.8)                                KM  2212
9     FORMAT (1H0,4X,9HHEIGHT-M.,3X,A4,7(11X,A4)/,6X,8(11X,A4),/,           KM  2213
     1  6X,8(11X,A4),/,6X,8(11X,A4),/,6X,8(11X,A4),/,6X,8(11X,A4),/   ) KM  2214
13    FORMAT (1H0,  /,  34X,48HCONCENTRATION PROFILES-PARTS PER MILLION KM  2215
     1         //45X,5HTIME=,E12.5,8H MINUTES//10X,20HGROUND DIFFUSIVITY KM  2216
     2=,G15.8,10X, 1HK,I1,3H = ,G12.4,10X,1HK,I2,3H = ,G12.4)              KM  2217
14    FORMAT (1H0,10X,30HINITIAL DIFFUSION COEFFICIENTS//10X,52HVALUES SKM  2218
     1HOWN ARE FOR POINTS HALFWAY BETWEEN STATIONS///10X,7HSTATION,10X,1KM  2219
     25HHEIGHT (METERS),10X,11HDIFFUSIVITY,16X,7HDELTA Z,11X,11HCELL HEIKM  2220
     3GHT  )                                                               KM  2221
15    FORMAT (1H1,10X,84HNUMBER OF NO2 RATE CONSTANTS EXCEEDED THE DIMENKM  2222
     1SIONS OF VECTOR -RATK1-, JOB ABORTED.)                               KM  2223
16    FORMAT (1H0,//,55H  INPUT DATA FOR VERTICAL DIFFUSION WITH PHOTOCHKM  2224
     1EMISTRY  ///8X,15HINITIAL TIME IS,F25.2,8H MINUTES//8X13HFINAL TIMKM  2225
     1E IS,F27.2,8H MINUTES//8X25HINITIAL TIME STEP SIZE IS,F15.6,8H MINKM  2226
     3UTES  //,8X25HMAXIMUM TIME STEP SIZE IS ,F15.2,8H MINUTES ,///,8X  KM  2227
     322HNUMBER OF REACTIONS IS ,I18,                                       KM  2228
     4    //8X,20HNUMBER OF SPECIES IS ,I20,//,8X, 36HNUMBER OF SPECIES KM  2229
     5IN STEADY STATE IS,I4,                                               KM  2230
     6                 //8X33HNUMBER OF VERTICAL MESH POINTS IS,I7,4X,33HKM  2231
     7INCLUDING THE GROUND AND THE EDGE )                                  KM  2232
17    FORMAT (1H0,//13X,8(8X,A4),6X,13HRATE CONSTANT,/,13X,8(8X,A4),       KM  2233
     1 /,13X,8(8X,A4),/,13X,8(8X,A4),/,13X,8(8X,A4) )                      KM  2234
18    FORMAT(1H0,                                                           KM  2235
     1/////4X9HHEIGHT-M.,5X,A4,7(11X,A4)/,7X,8(11X,A4),/,7X,8(11X,A4),     KM  2236
     2  /,7X,8(11X,A4),/,7X,8(11X,A4)  )                                   KM  2237
19    FORMAT (1H0,5X8HREACTION,25X55H- - - R E A C T A N T   S T O I C HKM  2238
     1 I O M E T R Y - - -,24X,15H(1/(PPM--MIN))  )                        KM  2239
20    FORMAT(1H0,5X,I3,2X,8F12.2,10X,G12.5)                                 KM  2240
21    FORMAT (1H ,10X,8F12.2,/, 3(11X,8F12.2,/)  )                          KM  2241
22    FORMAT (1H0,//6X8HREACTION,26X53H- - - P R O D U C T   S T O I C HKM  2242
     1 I O M E T R Y - - -)                                                KM  2243
26    FORMAT (1H1,25(1H /),10X,36HNUMBER OF INTEGRATION STEPS EXCEEDED,2KM  2244
     14H LIMIT.  JOB TERMINATED.)                                          KM  2245
27    FORMAT(1H ,I3,3H.  ,A4,14X,F7.1,16X,I2)                               KM  2246
28    FORMAT(1H0,/6X,7HSPECIES,7X,16HMOLECULAR WEIGHT,5X,9HBOUNDARY         KM  2247
     1  11HCONDITION    ,/)                                                KM  2248
29    FORMAT (40X,A4)                                                       KM  2249
30    FORMAT (40X,F10.0)                                                    KM  2250
31    FORMAT (40X,I10)                                                      KM  2251
32    FORMAT (1H0,2X,F7.2,8E15.5,/, 4(10X,8E15.5,/) )                       KM  2252
33    FORMAT (1H1,25(/),25X,20A4//25X,8HDATE -- ,A10,12X,10HCP TIME = ,FKM  2253
     110.2,5H  SEC)                                                        KM  2254
37    FORMAT(1H0,    20X 17HR E A C T I O N S ,55X,30HRATE CONSTANTS (1/KM  2255
```

```
     1(PPM--MIN))  /)                                                    KM  2256
38    FORMAT(1H ,3X,I3,2H.  ,20A4,8X,E12.4)                              KM  2257
39    FORMAT (11X,A4,5X,11HAT STATION ,I3,5X,25HNEGATIVE OR ZERO AT TIMEKM  2258
     1 ,E10.3,5X,18HCONCENTRATION IS  ,E10.3,5X,8HDELT IS ,E10.3)       KM  2259
40    FORMAT (20A4)                                                      KM  2260
41    FORMAT (20X,5E10.4)                                                KM  2261
42    FORMAT(1H0,10X,I5,15X,F10.2,55X,F10.2)                             KM  2262
43    FORMAT(1H0,50X,G15.8,10X,F10.2)                                    KM  2263
44    FORMAT(1H0,10X,I5,15X,F10.2,10X,G15.8,30X,F10.2)                   KM  2264
45    FORMAT (1H1,10X,57HTOO MANY INVERSION HEIGHT ENTRIES IN TABLE.  JOKM  2265
     18 ABORTED.)                                                       KM  2266
46    FORMAT (1H0,10X,50HINVERSION HEIGHT AND DIFFUSIVITY UPDATED AT TIMKM  2267
     1E ,F10.2,5X,14HUPDATE TIME = ,F10.2,5X,9HINV HT = ,F10.2)         KM  2268
47    FORMAT(1H0,10X,I5,15X,F10.2,10X,G15.8)                             KM  2269
51    FORMAT (40X,A4,6X,F10.2,I10)                                       KM  2270
52    FORMAT(20X,5(I2,8X)/20X,5(I2,8X))                                  KM  2271
53    FORMAT (40X,I10,2F10.0)                                            KM  2272
54    FORMAT (20X,5E12.4)                                                KM  2273
55    FORMAT (20X,10F6.2)                                                KM  2274
57    FORMAT (1H1,10X,45HTOO MANY CLOUD COVER ENTRIES  -- JOB ABORTED) KM  2275
58    FORMAT(10(/),10X,73HDEPLETION OF  NO  HAS CAUSED FAILURE OF CHEMICKM  2276
     1AL MODEL.  CASE TERMINATED. )                                     KM  2277
59    FORMAT (40X,2F10.0)                                                KM  2278
60    FORMAT(1H1,/////,20X,41HJOB TERMINATED -- NEGATIVE CONCENTRATIONS KM  2279
     1   30H OR DEPLETION OF NITRIC OXIDE       )                      KM  2280
66    FORMAT(1H09X20HINTEGRATION INDEX = ,I4,10X,18HLAST STEP SIZE =    KM  2281
     1 E10.3,20X,17HLAST ORDER USED = ,I3,//,10X,18HNUMBER OF STEPS = , KM  2282
     2 I6,10X,33HNUMBER OF FUNCTION EVALUATIONS = ,I4,10X,34H NUMBER OF KM  2283
     3JACOBIAN EVALUATIONS = ,I4)                                       KM  2284
68    FORMAT(1H0, 5X,70HREVISED AREA SOURCE FLUX SCHEDULE      ---       KM  2285
     1    PPM-METERS/MINUTE  ,//,6X,4HTIME,7(11X,A4),/,10X,,3(11X,A4)) KM  2286
69    FORMAT(1H ,F10.1,1X,7E15.4)                                        KM  2287
74    FORMAT (1H0,///,8H  TIME =, F8.2,5X,9HINDEX =   ,I2,               KM  2288
     1  36H  INTEGRATION TERMINATED BY EPISODE   )                      KM  2289
75    FORMAT (1H0,10X,33HCLEARNESS RATIO UPDATED AT TIME =   ,F8.2,5X,   KM  2290
     1    20HSKY COVER FACTOR  =   , F8.4)                             KM  2291
76    FORMAT(1H1,9X,41HJOB ABORTED BECAUSE OF TOO MANY SPECIES  , //     KM  2292
     1/,9H NOSPEC =  ,I3,5X,9H  NSTOY=   ,I3,5X,5H NK=   ,I3)          KM  2293
77    FORMAT(1H1,58HJOB ABORTED BECAUSE OF TOO FEW VERTICAL STATIONS  ,/KM  2294
     1//,10X,8HNOSTAT = , I2)                                           KM  2295
79    FORMAT (47H0  TOO MANY TEMPERATURES INPUT -- JOB ABORTED       )   KM  2296
80    FORMAT (20A4)                                                      KM  2297
81    FORMAT ( 7(6E12.4,/))                                              KM  2298
83    FORMAT (1H0,5X,50HPOINT SOURCE EMISSION RATES BY VERTICAL CELL     KM  2299
     1    10X,10H(PPM/MIN)  ,//,6X,4HTIME,2X,8(8X,A4))                  KM  2300
84    FORMAT (1H0,F9.1,4X,8E12.3)                                        KM  2301
85    FORMAT (1H ,13X,8E12.3)                                            KM  2302
88    FORMAT(1H ,5X,A4,8H SINK = ,2X,G15.8)                              KM  2303
91    FORMAT(1H0,/,40X,6HTIME =,F8.2,14H MINUTES     ( ,  I4,            KM  2304
     1 17H ON 2400-CLOCK)   //,40X,24HSURFACE CELL PARAMETERS           KM  2305
     2 40X,30HNO2 PHOTODISSOCIATION RATE  =   ,F8.3,10H   (/MIN)  ,/,   KM  2306
     3 40X,30HDIFFUSIVITY COEFFICIENT     =   ,F8.1,13H   (M**2/MIN))   KM  2307
92    FORMAT(1H0,39X,12HAREA SOURCES,2X,9H(PPM/MIN)3X,13HPOINT SOURCES   KM  2308
     1  2X,9H(PPM/MIN) )                                                KM  2309
93    FORMAT(  10(/,43X,A4,3H = ,E11.3,8X,A4,3H = ,E11.3))               KM  2310
```

```
94    FORMAT(1H ,//,40X,43HCONCENTRATION PROFILES IN PARTS PER MILLION )KM  2311
96    FORMAT(1H ,10X,6F15.2)                                             KM  2312
97    FORMAT(1H0,/,40X,6HTIME =,F8.2,14H MINUTES     ( ,  I4,            KM  2313
     1 17H ON 2400-CLOCK)   //,                                          KM  2314
     1/,40X,51HINITIAL CONCENTRATION PROFILES IN PARTS PER MILLION/)     KM  2315
99    FORMAT(11(//),57X,24HTHE ERT TRAJECTORY MODEL ,/57X6(4H====),      KM  2316
     1  .//,59X,20H(CODE DATE   4,1,78) ////63X12HDEVELOPED BY  ,//      KM  2317
     1  48X,41HENVIRONMENTAL RESEARCH & TECHNOLOGY, INC.  //             KM  2318
     1  58X,22HSANTA BARBARA DIVISION   )                                KM  2319
      END                                                                KM  2320


      SUBROUTINE MATMUL(A,B,R,N,M,L)                                     KM  2321
      DIMENSION A(1),B(1),R(1)                                           KM  2322
C                                                                        KM  2323
      IR = 0                                                             KM  2324
      IK = -M                                                            KM  2325
      DO 10 K=1,L                                                        KM  2326
       IK = IK + M                                                       KM  2327
        DO 10 J=1,N                                                      KM  2328
         IR = IR + 1                                                     KM  2329
         JI = J-N                                                        KM  2330
         IB = IK                                                         KM  2331
         R(IR) = 0.                                                      KM  2332
          DO 10 I = 1,M                                                  KM  2333
           JI = JI+N                                                     KM  2334
           IB = IB + 1                                                   KM  2335
           R(IR) = R(IR) + A(JI)*B(IB)                                   KM  2336
   10 CONTINUE                                                           KM  2337
      RETURN                                                             KM  2338
      END                                                                KM  2339


      SUBROUTINE PEDERV(NO,Y,Q,NKM)                                      KM  2340
C                                                                        KM  2341
C           THIS SUBROUTINE CALCULATES THE BLOCK DIAGONAL JACOBIAN       KM  2342
C           MATRACIES BY CALLING JACOB FOR EACH STATION                  KM  2343
C                                                                        KM  2344
      INTEGER BCFLAG                                                     KM  2345
      COMMON/CHEM1/ NOSTAT,            NOSTM1,          NOREAC,          KM  2346
     1              NOSPEC,            NSTDY,           NK               KM  2347
      COMMON/CHEM2/ CONIN(40,5),       WTMOLE(40),      RATKON(55),      KM  2348
     1              RATEFF(55),        RATEV(2,5),      QRATE,           KM  2349
     2              NVRATE,            LOCVRT(2)                         KM  2350
      COMMON/CHEM3/ ZEE(5),            DELZ(4),         HTCELL(6),       KM  2351
     1              TDELZ(2) ,         DFINIT(6),       SCALOW(4),       KM  2352
     2              DCOF(5),           FLXWAL(40),      FLXDGE(40),      KM  2353
     3              BCFLAG(40),        DPRATE(10),      DEPOWR(10),      KM  2354
     4              LOCDPF(10),        NDPFLX ,         SCALUP(4)        KM  2355
      DIMENSION     Y(NO,1),           Q(NKM,NKM,1)                      KM  2356
      DATA YES /3HYES/                                                   KM  2357
      DO 20 I = 1,NOSTAT                                                 KM  2358
      KS = NK*(I-1) + 1                                                  KM  2359
```

```
      IF(GRATE .EQ. YES)  CALL RATEHI(I)                                KM  2360
      CALL XMIT(NK,Y(KS,1), CONIN(1,I))                                 KM  2361
      CALL UNMIXR(I)                                                    KM  2362
      CALL JACOB(Q(1,1,I) , CONIN(1,I), RATEFF, NK )                    KM  2363
      DO 10 J=1,NK                                                      KM  2364
      Q(J,J,I) = Q(J,J,I) - DCOF(I)                                     KM  2365
   10 CONTINUE                                                          KM  2366
      IF(I .NE. 1)  GO TO 20                                            KM  2367
      IF(NDPFLX.LT.1)  GO TO 20                                         KM  2368
      DO 15 K=1,NDPFLX                                                  KM  2369
      ID = LOCDPF(K)                                                    KM  2370
      IF(DEPOWR(K).LT. .99)  GO TO 12                                   KM  2371
      IF(DEPOWR(K).GT.1.01)  GO TO 12                                   KM  2372
      Q(ID,ID,1) = Q(ID,ID,1) -TDELZ(1)*DPRATE(K)                       KM  2373
      GO TO 15                                                          KM  2374
   12 Q(ID,ID,1) = Q(ID,ID,1) -TDELZ(1)*DPRATE(K)*DEPOWR(K)*            KM  2375
     1                                    (Y(ID)**(DEPOWR(K)-1.))      KM  2376
   15 CONTINUE                                                          KM  2377
   20 CONTINUE                                                          KM  2378
      RETURN                                                            KM  2379
      END                                                               KM  2380



      SUBROUTINE PHOTOD(JSTART,JSTOP,NOSTAT,ZEE)                        KM  2381
C                                                                       KM  2382
C        PHOTOD GENERATES THE PHOTODISSOCIATION RATES OF NO2 AND HCHO   KM  2383
C        AS A FUNCTION OF TIME AND HEIGHT.                              KM  2384
C                                                                       KM  2385
      DIMENSION ZINP(10),  HTINP(11),  ZEE(1),  HTD(6),  NAM(2),        KM  2386
     1          RA(11,10), RB(11,10)                                    KM  2387
      COMMON /INPUTS/ TITLE(20),  IDATE(10),   NCURV                    KM  2388
      COMMON /CHEM4/  RATK1(100,5),      RATK2(100,5),   RLAT,          KM  2389
     1                RLONG,             TMZONE,         SUNTIM,       KM  2390
     2                HIRATE,            JDATE,          NRATE         KM  2391
      EQUIVALENCE (KYES,YES)                                            KM  2392
C                                                                       KM  2393
      DATA ZINP /0.,10.,20.,30.,40.,50.,60.,70.,78.,86./                KM  2394
      DATA HTINP /0.,150.,360.,640.,980.,1380.,1840.,2350.,2910.,       KM  2395
     1           3530.,4210./                                           KM  2396
C                                                                       KM  2397
C        RA CONTAINS SURFACE AND ELEVATED NO2 CLEAR-SKY NORMAL          KM  2398
C           AEROSOL PHOTODISSOCIATION RATES                             KM  2399
C                                                                       KM  2400
      DATA RA /.579,.614,.645,.675,.703,.729,.752,.772,.790,.808,.824,  KM  2401
     1         .574,.609,.640,.671,.700,.725,.748,.768,.787,.805,.821,  KM  2402
     2         .560,.596,.628,.659,.688,.715,.737,.758,.777,.795,.812,  KM  2403
     3         .535,.572,.605,.637,.667,.694,.717,.738,.758,.776,.794,  KM  2404
     4         .496,.534,.568,.601,.631,.659,.683,.704,.725,.745,.764,  KM  2405
     5         .438,.477,.512,.545,.576,.604,.628,.650,.673,.694,.713,  KM  2406
     6         .352,.391,.426,.459,.490,.517,.541,.564,.583,.608,.631,  KM  2407
     7         .231,.264,.295,.325,.353,.378,.400,.421,.443,.466,.489,  KM  2408
     8         .114,.133,.153,.174,.194,.212,.229,.246,.264,.283,.303,  KM  2409
     9         .025,.027,.030,.034,.037,.040,.043,.046,.050,.055,.060/  KM  2410
C                                                                       KM  2411
```

```
C      RB CONTAINS THE SURFACE HCHO RATES AND THE PERCENTAGE CHANGE      KM  2412
C         ABOVE THE SURFACE RATES AS A FUNCTION OF ELEVATION.            KM  2413
C                                                                        KM  2414
       DATA RB /2.18,8.00,15.8,23.7,31.6,38.3,46.0,51.8,3*58.3,          KM  2415
      1          2.15,8.00,15.9,23.8,31.7,38.5,46.3,52.2,3*58.7,        KM  2416
      2          2.05,8.30,16.4,24.6,32.8,39.8,47.9,54.0,3*60.8,        KM  2417
      3          1.88,8.80,17.4,26.1,34.6,42.0,50.6,57.2,3*64.5,        KM  2418
      4          1.64,9.50,18.8,28.2,37.5,45.5,54.8,62.1,3*70.2,        KM  2419
      5          1.32,10.2,20.9,31.3,41.7,50.7,61.1,69.6,3*78.9,        KM  2420
      6          .934,12.0,23.8,35.9,47.9,58.5,70.8,81.1,3*92.5,        KM  2421
      7          .507,13.6,27.4,41.7,56.3,69.5,84.6,98.6,3*114.,        KM  2422
      8          .211,13.5,27.7,42.8,58.7,73.3,90.2,107.,3*126.,        KM  2423
      9          .040,11.6,22.7,34.1,45.6,55.5,66.9,77.6,3*89.3/        KM  2424
       DATA NHTI /11/                                                    KM  2425
       DATA NZINP /10/                                                   KM  2426
       DATA NAM /4HNO2 , 4HHCHO /                                        KM  2427
       DATA KYES /4HYES /                                                KM  2428
C                                                                        KM  2429
C      SPECIFY THE RA & RB MULTIPLIERS                                   KM  2430
C                                                                        KM  2431
       DATA RADJST, RBEXP /0.950,.00285/                                 KM  2432
       DATA NV /5/                                                       KM  2433
C                                                                        KM  2434
C                                                                        KM  2435
       IINC = IFIX(SUNTIM)                                               KM  2436
       TIME = FLOAT(JSTART)                                              KM  2437
       NTMSTP = (JSTOP - JSTART)*6/10/IINC + 1                           KM  2438
       IY = JDATE/10000                                                  KM  2439
       IM = JDATE/100 - IY*100                                           KM  2440
       ID = JDATE - IY*10000 - IM*100                                    KM  2441
       IY = IY + 1900                                                    KM  2442
       IRITE = NV                                                        KM  2443
       NSTAT = NOSTAT                                                    KM  2444
       CALL XMIT(NSTAT,ZEE,HTD)                                          KM  2445
       IF(HIRATE.NE.YES)  NSTAT = 1                                      KM  2446
       DO 5 K =1,10                                                      KM  2447
       RB(1,K) = RB(1,K)*RBEXP                                           KM  2448
       RA(1,K) = RADJST*RA(1,K)                                          KM  2449
       DO 5 J =2,11                                                      KM  2450
       RB(J,K) = RB(1,K)*(1.0 + 0.01*RB(J,K))                            KM  2451
       RA(J,K) = RADJST*RA(J,K)                                          KM  2452
  5    CONTINUE                                                          KM  2453
C                                                                        KM  2454
       DO 50 IS = 1,NTMSTP                                               KM  2455
       CALL SOLAR(RLAT,RLONG,TMZONE,IY,IM,ID,TIME,D,NV)                  KM  2456
       ZEN = 90. - D                                                     KM  2457
       IF(ZEN.LE.ZINP(1))  GO TO 30                                      KM  2458
       IF(ZEN.GE.ZINP(NZINP))  GO TO 30                                  KM  2459
C         FIND NEIGHBORING ZENITH ANGLE INDICIES                         KM  2460
       DO 10 I = 1,NZINP                                                 KM  2461
       IF(ZINP(I).LT.ZEN)   GO TO 10                                     KM  2462
       Z1 = ZINP(I-1)                                                    KM  2463
       Z2 = ZINP(I)                                                      KM  2464
       I1 = I - 1                                                        KM  2465
       I2 = I                                                            KM  2466
```

```
      GO TO 15                                                          KM  2467
   10 CONTINUE                                                          KM  2468
   15 CONTINUE                                                          KM  2469
C                                                                       KM  2470
C        INTERPOLATE ON ZENITH ANGLE THEN ELEVATION                     KM  2471
C                                                                       KM  2472
      J = 1                                                             KM  2473
      DO 20 JJ = 1,NSTAT                                                KM  2474
      P = (ZEN - Z1)/(Z2 - Z1)                                          KM  2475
      K = J + 1                                                         KM  2476
      R1 = RA(J,I1) + P*(RA(J,I2) - RA(J,I1))                           KM  2477
      R2 = RA(K,I1) + P*(RA(K,I2) - RA(K,I1))                           KM  2478
      PH = (HTO(JJ) - HTINP(J))/(HTINP(K) - HTINP(J))                   KM  2479
      RATK1(IS,JJ) = R1 + PH*(R2 - R1)                                  KM  2480
      R1 = RB(J,I1) + P*(RB(J,I2) - RB(J,I1))                           KM  2481
      R2 = RB(K,I1) + P*(RB(K,I2) - RB(K,I1))                           KM  2482
      RATK2(IS,JJ) = R1 + PH*(R2 - R1)                                  KM  2483
      IF(JJ.EQ.NSTAT)   GO TO 20                                        KM  2484
      DO 18 KK = 1,5                                                    KM  2485
      IF(HTO(JJ+1) .GT. HTINP(J+1))  J = J + 1                          KM  2486
   18 CONTINUE                                                          KM  2487
      IF(J.LE.NHTI)  GO TO 20                                           KM  2488
      J = NHTI                                                          KM  2489
      HTO(JJ+1) = HTINP(NHTI)                                           KM  2490
   20 CONTINUE                                                          KM  2491
C                                                                       KM  2492
      GO TO 40                                                          KM  2493
   30 CONTINUE                                                          KM  2494
      DO 35 JJ = 1,NSTAT                                                KM  2495
      RATK1(IS,JJ) = 5.E-03                                             KM  2496
      RATK2(IS,JJ) = 1.E-05                                             KM  2497
   35 CONTINUE                                                          KM  2498
   40 CONTINUE                                                          KM  2499
C                                                                       KM  2500
C        UPDATE TIME ON 2400 HR CLOCK                                   KM  2501
C                                                                       KM  2502
      IHR = TIME/100                                                    KM  2503
      MIN = TIME - IHR*100                                              KM  2504
      NM = MIN + IINC                                                   KM  2505
      IF(NM.LT.60)  GO TO 45                                            KM  2506
      NM = NM - 60                                                      KM  2507
      IHR = IHR + 1                                                     KM  2508
   45 TIME = IHR*100 + NM                                               KM  2509
   50 CONTINUE                                                          KM  2510
C                                                                       KM  2511
C       WRITE PHOTODISSOCIATION RATE ARRAYS  (IF IRITE EQUALS YES)      KM  2512
C                                                                       KM  2513
      IF(IRITE.NE.KYES)  GO TO 80                                       KM  2514
      K = 1                                                             KM  2515
   55 CONTINUE                                                          KM  2516
      IHR = JSTART/100                                                  KM  2517
      MIN = JSTART - IHR*100                                            KM  2518
      TIME = FLOAT( IHR*60 + MIN )                                      KM  2519
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  2520
      WRITE(6,120)  NAM(K) ,  (HTO(I),I=1,NSTAT)                        KM  2521
```

```
      DO 70 J = 1,NTMSTP                                                KM  2522
      IF(MOD(J,50) .NE. 0)  GO TO 60                                    KM  2523
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  2524
      WRITE(6,120)  NAM(K) ,  (HTD(I),I=1,NSTAT)                        KM  2525
   60 CONTINUE                                                          KM  2526
      IF(K.EQ.1)  WRITE(6,130) TIME,(RATK1(J,I),I=1,NSTAT)              KM  2527
      IF(K.EQ.2)  WRITE(6,130) TIME,(RATK2(J,I),I=1,NSTAT)              KM  2528
      TIME = TIME + SUNTIM                                              KM  2529
   70 CONTINUE                                                          KM  2530
      K = K + 1                                                         KM  2531
      IF(K.LE.2)  GO TO 55                                              KM  2532
   80 NRATE = NTMSTP                                                    KM  2533
      RETURN                                                            KM  2534
C                                                                       KM  2535
  120 FORMAT (1H0,A4,1X,24HPHOTODISSOCIATION RATES ,20X,10HELEVATIONS,  KM  2536
     1        //,6X,4HTIME,10X,5(F10.1,2H M) ,/)                       KM  2537
  130 FORMAT (1H ,4X,F5.0,10X,5F12.5)                                   KM  2538
      END                                                               KM  2539



      SUBROUTINE PRODUK(UPDINT)                                         KM  2540
C                                                                       KM  2541
C        *PRODUK* CAN BE MODIFIED TO CALCULATE                          KM  2542
C         APPROXIMATE PRODUCT SPECIES CONCENTRATIONS                    KM  2543
C                                                                       KM  2544
      COMMON/CHEM1/ NOSTAT,NOSTM1,NOREAC,NOSPEC,NSTDY,NK                KM  2545
      COMMON/CHEM2/ CONIN(40,5),WTMOLE(40),RATKON(55),RATEFF(55),       KM  2546
     1  RATEV(2,5),QRATE,NVRATE,LOCVRT(2)                               KM  2547
C                                                                       KM  2548
C     ----- CURRENTLY *PRODUK* IS A DUMMY SUBROUTINE  ----             KM  2549
C                                                                       KM  2550
      RETURN                                                            KM  2551
      END                                                               KM  2552



      SUBROUTINE PSET (Y,NO,CON,MITER,IER)                              KM  2553
C-----------------------------------------------------------------------KM  2554
C     THIS VERSION OF PSET IS DESIGNED FOR A BLOCK TRI-DIAGONAL         KM  2555
C     SYSTEM OF ODE*S AND IS ONLY TO BE USED WITH MITER = 1.            KM  2556
C     MODIFICATIONS BY F.W. LURMANN       (7.8.77).                     KM  2557
C-----------------------------------------------------------------------KM  2558
C PSET IS CALLED BY TSTEP TO COMPUTE AND TO PROCESS THE MATRIX          KM  2559
C P = I - (H/EL(2))*J, WHERE J IS AN APPROXIMATION TO THE               KM  2560
C JACOBIAN.  J IS COMPUTED BY THE USER SUPPLIED SUBROUTINE PEDERV.      KM  2561
C-----------------------------------------------------------------------KM  2562
      INTEGER IER, MITER, NO                                            KM  2563
      INTEGER IPIV, JSTART, KFLAG, MF, N, NSQ                           KM  2564
      INTEGER I, IER, J, J1, N                                          KM  2565
      DIMENSION Y(NO,1)                                                 KM  2566
C                                                                       KM  2567
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART     KM  2568
      COMMON /EPCOM2/ YMAX(1)                                           KM  2569
      COMMON /EPCOM4/ SAVE1(1)                                          KM  2570
```

```
      COMMON /EPCOM5/ SAVE2(1)                                          KM  2571
      COMMON /EPCOM6/ PW(1)                                             KM  2572
      COMMON /EPCOM7/ IPIV(1)                                           KM  2573
      COMMON /EPCOM8/ EPSJ,NSQ                                          KM  2574
      COMMON /EPCM12/ PWL(3600),  WORKR(30), DUPPER(4), DLOWER(4)       KM  2575
      INTEGER BCFLAG                                                    KM  2576
      COMMON/CHEM1/ NOSTAT,             NOSTM1,           NOREAC,       KM  2577
     1              NOSPEC,             NSTDY,            NK            KM  2578
      COMMON/CHEM2/ CONIN(40,5),        WTMOLE(40),       RATKON(55),   KM  2579
     1              RATEFF(55),         RATEV(2,5),       QRATE,        KM  2580
     2              NVRATE,             LOCVRT(2)                       KM  2581
      COMMON/CHEM3/ ZEE(5),             DELZ(4),          HTCELL(6),    KM  2582
     1              TDELZ(2) ,          DFINIT(6),        SCALOW(4),    KM  2583
     2              DCOF(5),            FLXWAL(40),       FLXOGE(40),   KM  2584
     3              BCFLAG(40),         OPRATE(10),       DEPOWR(10),   KM  2585
     4              LOCOPF(10),         NOPFLX ,          SCALUP(4)     KM  2586
      DATA ONE /1.0E0/, REP /1.0E-3/, ZERO /0.0E0/                      KM  2587
      NKM = NK                                                          KM  2588
      IF (MITER.NE.1)  GO TO 9                                          KM  2589
C IF MITER = 1, CALL PEDERV AND MULTIPLY BY A SCALAR. ****************KM  2590
      CALL PEDERV(NO,Y,PW,NKM)                                          KM  2591
C     SCALE BLOCK DIAGONAL MATRICIES Q BY CON                           KM  2592
      NK2N = NK*NK*NOSTAT                                               KM  2593
      DO 5 I=1,NK2N                                                     KM  2594
      PW(I) = PW(I)*CON                                                 KM  2595
   5  CONTINUE                                                          KM  2596
C     ADD IDENTITY MATRIX TO Q MATRICES                                 KM  2597
      NKP1 = NK + 1                                                     KM  2598
      NK2 = NK*NK                                                       KM  2599
      DO 6 I=1,NOSTAT                                                   KM  2600
      K = (I-1)*NK2 + 1                                                 KM  2601
      DO 6 J=1,NK                                                       KM  2602
      PW(K) = PW(K) + ONE                                               KM  2603
      K = K + NKP1                                                      KM  2604
   6  CONTINUE                                                          KM  2605
C     SCALE UPPER AND  LOWER OFF-DIAGONAL VECTORS BY CON                KM  2606
      DO 7 K=1,NOSTM1                                                   KM  2607
      DUPPER(K) = SCALUP(K)*CON                                         KM  2608
      DLOWER(K) = SCALOW(K)*CON                                         KM  2609
   7  CONTINUE                                                          KM  2610
      CALL BLKDEC(PW,NK,NOSTAT,IPIV,DLOWER,DUPPER,PWL,WORKR,NOGO,BCFLAG)KM  2611
C     NOW TL AND Q CONTAIN THE LOWER AND UPPER COMPOSITION OF Q         KM  2612
C     AND IPIV CONTAINS THE PIVOTING INFORMATION FOR BLKSOL             KM  2613
      IER = NOGO                                                        KM  2614
      RETURN                                                            KM  2615
   9  CONTINUE                                                          KM  2616
20    WRITE(6,21)                                                       KM  2617
21    FORMAT (1H0,90H THIS VERSION OF PSET IS RESTRICTED TO MITER = 1   KM  2618
     1  ***    JOB ABORTED                                           ) KM  2619
      STOP                                                              KM  2620
22    RETURN                                                            KM  2621
C*********************  END OF SUBROUTINE PSET **********************KM  2622
      END                                                               KM  2623
```

```
      SUBROUTINE RATEHI(K)                                                  KM  2624
C                                                                           KM  2625
C     -------- THIS SUBROUTINE TRANSFERS THE VARIABLE RATE                  KM  2626
C              CONSTANTS AT HIGHER ELEVATIONS  IN  THE ARRAY                KM  2627
C              RATEV TO RATKON AND RATEFF                                   KM  2628
C     -------- PHOTODISSOCIATION RATES WHICH ARE PRPORTIONAL                KM  2629
C              TO  RATEV(1,N) OR RATEV(2,N)ARE ASSIGNED                     KM  2630
C     -------- HONO AND RCHO PHOTOLYSIS RATES ARE ASSUMED                   KM  2631
C              PROPORTIONAL TO NO2 AND HCHO.                                KM  2632
C     -------- THIS ROUTINE IS SPECFIC TO THE ERT                           KM  2633
C              30 SPECIE X 51 REACTION MECHANISM 4.1.78                     KM  2634
C                                                                           KM  2635
      COMMON/CHEM2/ CONIN(40,5),          WTMOLE(40),          RATKON(55),  KM  2636
     1              RATEFF(55),           RATEV(2,5),          QRATE,       KM  2637
     2              NVRATE,               LOCVRT(2)                         KM  2638
      I1 = LOCVRT(1)                                                        KM  2639
      RATKON(I1) = RATEV( 1,K)                                              KM  2640
      RATEFF(I1) = RATKON(I1)                                               KM  2641
      RATKON( 6) = .280*RATKON(I1)                                          KM  2642
      RATEFF( 6) = RATKON( 6)                                               KM  2643
      IF(NVRATE.LT. 2)  RETURN                                              KM  2644
      I2 = LOCVRT(2)                                                        KM  2645
      RATKON(I2)  = RATEV(2,K)                                              KM  2646
      RATEFF(I2)  =  RATKON(I2)                                             KM  2647
C     ALDEHYDE PHOTOLYSIS CORRECTION                                        KM  2648
      RATKON(34) = RATKON(I2)/3.0                                           KM  2649
      RATEFF(34) = RATKON(34)                                               KM  2650
      RETURN                                                                KM  2651
      END                                                                   KM  2652



      SUBROUTINE RATES(Y,YDOT)                                              KM  2653
C                                                                           KM  2654
C          CALCULATION OF CHEMICAL RATES                                    KM  2655
C                                                                           KM  2656
      INTEGER BCFLAG                                                        KM  2657
      COMMON/CHEM1/ NOSTAT,               NOSTM1,              NOREAC,      KM  2658
     1              NOSPEC,               NSTDY,               NK           KM  2659
      COMMON/CHEM2/ CONIN(40,5),          WTMOLE(40),          RATKON(55),  KM  2660
     1              RATEFF(55),           RATEV(2,5),          QRATE,       KM  2661
     2              NVRATE,               LOCVRT(2)                         KM  2662
      COMMON/CHEM3/ ZEE(5),               DELZ(4),             HTCELL(6),   KM  2663
     1              TDELZ(2) ,            DFINIT(6),           SCALOW(4),   KM  2664
     2              DCOF(5),              FLXWAL(40),          FLXDGE(40),  KM  2665
     3              BCFLAG(40),           DPRATE(10),          DEPOWR(10),  KM  2666
     4              LOCDPF(10),           NDPFLX ,             SCALUP(4)    KM  2667
      DIMENSION Y(1), YDOT(1)  ,  RATE(40),  C(40),  R(55)                  KM  2668
      EQUIVALENCE (R,RATEFF)                                                KM  2669
      DATA  YES  /3HYES/                                                    KM  2670
      NKP1 = NK + 1                                                         KM  2671
      DO 130 K = 1,NOSTAT                                                   KM  2672
      KS = NK*(K-1)                                                         KM  2673
      DO 90  I=1,NK                                                         KM  2674
      C(I) = Y(I+KS)                                                        KM  2675
```

```
 90   CONTINUE                                                          KM  2676
      IF(NSTDY.EQ.0)  GO TO 105                                         KM  2677
      DO 100 I=NKP1,NOSPEC                                              KM  2678
      C(I) = CONIN(I,K)                                                 KM  2679
100   CONTINUE                                                          KM  2680
105   CONTINUE                                                          KM  2681
C                                                                       KM  2682
      IF(QRATE.EQ.YES)  CALL RATEHI(K)                                  KM  2683
      CALL UNMIXR(K)                                                    KM  2684
C                                                                       KM  2685
C      EXPLICIT CHEMICAL RATE EQUATIONS FOR                             KM  2686
C                                                                       KM  2687
C    ERT PHOTOCHEMICAL MECHANISM (4.1.78)  (30 SPECIES X 51 REACTIONS)  KM  2688
C                                                                       KM  2689
      RATE(  1) = + R(  1)*C(  2) - R(  3)*C(  1)*C(  3) - R(  4)*C(  1)KM  2690
     *     *C(  2)*C( 30) + R(  5)*C(  4)** 2 + R(  6)*C(  4) - R(  7)   KM  2691
     *     *C(  1)*C( 24) - R( 10)*C(  1)*C( 11) - R( 14)*C(  1)*C( 10)  KM  2692
     *      - R( 19)*C(  1)*C( 17) - R( 25)*C(  1)*C( 19) - R( 27)*C(  1)KM  2693
     *     *C( 18) - R( 28)*C(  1)*C( 18) - R( 30)*C(  1)*C( 15) - R( 35)KM  2694
     *     *C(  1)*C( 16) - R( 43)*C(  1)*C( 14) - R( 45)*C(  1)*C( 21)  KM  2695
      RATE(  2) = - R(  1)*C(  2) + R(  3)*C(  1)*C(  3) - R(  4)*C(  1)KM  2696
     *     *C(  2)*C( 30) + R(  5)*C(  4)** 2 - R(  8)*C(  2)*C( 24) +   KM  2697
     *     R( 10)*C(  1)*C( 11) - R( 11)*C(  2)*C( 11) - R( 13)*C(  2)   KM  2698
     *     *C(  3) +  2.00*R( 14)*C(  1)*C( 10) - R( 15)*C(  2)*C( 10) + KM  2699
     *     R( 17)*C( 27) + R( 19)*C(  1)*C( 17) + R( 21)*C( 20) - R( 23) KM  2700
     *     *C(  2)*C( 19) + R( 25)*C(  1)*C( 19) + R( 27)*C(  1)*C( 18)  KM  2701
     *      + R( 30)*C(  1)*C( 15) - R( 32)*C(  2)*C( 26) - R( 36)*C(  2)KM  2702
     *     *C( 16) + R( 37)*C( 12) + R( 43)*C(  1)*C( 14) + R( 45)*C(  1)KM  2703
     *     *C( 21)                                                      KM  2704
      RATE(  3) = + R(  2)*C( 25) - R(  3)*C(  1)*C(  3) - R( 13)*C(  2)KM  2705
     *     *C(  3) - R( 22)*C(  3)*C(  8)                               KM  2706
      RATE(  4) = +  2.00*R(  4)*C(  1)*C(  2)*C( 30) -  2.00*R(  5)    KM  2707
     *     *C(  4)** 2 - R(  6)*C(  4) + R(  7)*C(  1)*C( 24) +         KM  2708
     *      0.15*R( 32)*C(  2)*C( 26)                                   KM  2709
      RATE(  5) = - R(  9)*C(  5)*C( 24) + R( 34)*C(  7) + R( 38)*C(  6)KM  2710
     *      + R( 39)*C(  6)*C( 24)                                      KM  2711
      RATE(  6) = + R( 20)*C( 23) +  0.50*R( 22)*C(  3)*C(  8) +        KM  2712
     *      0.50*R( 23)*C(  2)*C( 19) +  0.50*R( 25)*C(  1)*C( 19) +    KM  2713
     *      0.50*R( 29)*C( 26) - R( 38)*C(  6) - R( 39)*C(  6)*C( 24)   KM  2714
      RATE(  7) = + R( 20)*C( 23) +  0.50*R( 22)*C(  3)*C(  8) +        KM  2715
     *      0.50*R( 23)*C(  2)*C( 19) +  0.30*R( 24)*C(  8)*C( 25) +    KM  2716
     *      0.50*R( 25)*C(  1)*C( 19) +  0.50*R( 29)*C( 26) +           KM  2717
     *      0.50*R( 31)*C( 26) +  0.15*R( 32)*C(  2)*C( 26) - R( 33)    KM  2718
     *     *C(  7)*C( 24) - R( 34)*C(  7) + R( 43)*C(  1)*C( 14) + R( 49)KM  2719
     *     *C( 19)*C( 28)                                               KM  2720
      RATE(  8) = - R( 18)*C(  8)*C( 24) - R( 22)*C(  3)*C(  8) - R( 24)KM  2721
     *     *C(  8)*C( 25)                                               KM  2722
      RATE(  9) = - R( 26)*C(  9)*C( 24)                                KM  2723
      RATE( 10) = + R( 13)*C(  2)*C(  3) - R( 14)*C(  1)*C( 10) - R( 15)KM  2724
     *     *C(  2)*C( 10) + R( 17)*C( 27) + R( 23)*C(  2)*C( 19)        KM  2725
      RATE( 11) = + R(  9)*C(  5)*C( 24) - R( 10)*C(  1)*C( 11) - R( 11)KM  2726
     *     *C(  2)*C( 11) -  2.00*R( 12)*C( 11)** 2 + R( 20)*C( 23) +   KM  2727
     *     R( 21)*C( 20) +  0.25*R( 22)*C(  3)*C(  8) +  0.40*R( 24)    KM  2728
     *     *C(  8)*C( 25) + R( 31)*C( 26) + R( 34)*C(  7) +  0.67*R( 38) KM  2729
     *     *C(  6) + R( 39)*C(  6)*C( 24) - R( 40)*C( 11)*C( 15) + R( 41)KM  2730
```

```
     *     *C( 13)*C( 24) + R( 45)*C(  1)*C( 21) - R( 48)*C( 11)*C( 28)  KM  2731
      RATE( 12) = + R( 36)*C(  2)*C( 16) - R( 37)*C( 12)                  KM  2732
      RATE( 13) = - R( 41)*C( 13)*C( 24) - R( 44)*C( 13)*C( 24)           KM  2733
      RATE( 14) = + R( 42)*C( 22)*C( 24) - R( 43)*C(  1)*C( 14)           KM  2734
      RATE( 15) = +  0.40*R( 24)*C(  8)*C( 25) +  0.15*R( 27)*C(  1)      KM  2735
     *     *C( 18) + R( 29)*C( 26) - R( 30)*C(  1)*C( 15) + R( 34)*C(  7)KM  2736
     *      + R( 35)*C(  1)*C( 16) - R( 40)*C( 11)*C( 15) - R( 50)*C( 15)KM  2737
     *     *C( 28)                                                       KM  2738
      RATE( 16) = +  0.25*R( 22)*C(  3)*C(  8) + R( 33)*C(  7)*C( 24) - KM  2739
     *     R( 35)*C(  1)*C( 16) - R( 36)*C(  2)*C( 16) + R( 37)*C( 12)   KM  2740
      RATE( 17) = + R( 18)*C(  8)*C( 24) - R( 19)*C(  1)*C( 17) - R( 51)KM  2741
     *     *C( 17)*C( 28)                                                KM  2742
      RATE( 18) = + R( 26)*C(  9)*C( 24) - R( 27)*C(  1)*C( 18) - R( 28)KM  2743
     *     *C(  1)*C( 18)                                                KM  2744
      RATE( 19) = +  0.50*R( 22)*C(  3)*C(  8) - R( 23)*C(  2)*C( 19) - KM  2745
     *     R( 25)*C(  1)*C( 19) - R( 49)*C( 19)*C( 28)                   KM  2746
      RATE( 20) = + R( 11)*C(  2)*C( 11) - R( 21)*C( 20)                  KM  2747
      RATE( 21) = + R( 44)*C( 13)*C( 24) - R( 45)*C(  1)*C( 21)           KM  2748
      RATE( 22) = + R( 41)*C( 13)*C( 24) - R( 42)*C( 22)*C( 24)           KM  2749
      RATE( 23) = + R( 19)*C(  1)*C( 17) - R( 20)*C( 23) + R( 51)*C( 17)KM  2750
     *     *C( 28)                                                       KM  2751
      RATE( 24) = + R(  6)*C(  4) - R(  7)*C(  1)*C( 24) - R(  8)*C(  2)KM  2752
     *     *C( 24) - R(  9)*C(  5)*C( 24) + R( 10)*C(  1)*C( 11) - R( 18)KM  2753
     *     *C(  8)*C( 24) +  0.50*R( 22)*C(  3)*C(  8) - R( 26)*C(  9)   KM  2754
     *     *C( 24) - R( 33)*C(  7)*C( 24) - R( 39)*C(  6)*C( 24) - R( 41)KM  2755
     *     *C( 13)*C( 24) - R( 42)*C( 22)*C( 24) - R( 44)*C( 13)*C( 24)  KM  2756
     *      - R( 47)*C( 24)*C( 28) + R( 48)*C( 11)*C( 28)                KM  2757
      RATE( 25) = + R(  1)*C(  2) - R(  2)*C( 25) - R( 24)*C(  8)*C( 25)KM  2758
     *      - R( 46)*C( 25)*C( 28)                                       KM  2759
      RATE( 26) = +  0.85*R( 27)*C(  1)*C( 18) - R( 29)*C( 26) + R( 30) KM  2760
     *     *C(  1)*C( 15) - R( 31)*C( 26) - R( 32)*C(  2)*C( 26)         KM  2761
     *     +R( 50)*C( 25)*C( 28)                                         KM  2762
      RATE( 27) = + R( 15)*C(  2)*C( 10) - R( 16)*C( 27)*C( 30) - R( 17)KM  2763
     *     *C( 27)                                                       KM  2764
      RATE( 28) = - R( 46)*C( 25)*C( 28) - R( 47)*C( 24)*C( 28) - R( 48)KM  2765
     *     *C( 11)*C( 28) - R( 49)*C( 19)*C( 28) - R( 50)*C( 15)*C( 28)  KM  2766
     *      - R( 51)*C( 17)*C( 28)                                       KM  2767
      RATE( 29) = + R( 46)*C( 25)*C( 28) + R( 47)*C( 24)*C( 28) + R( 48)KM  2768
     *     *C( 11)*C( 28) + R( 49)*C( 19)*C( 28) + R( 50)*C( 15)*C( 28)  KM  2769
     *      + R( 51)*C( 17)*C( 28)                                       KM  2770
C     RATE( 30) = - R(  4)*C(  1)*C(  2)*C( 30) + R(  5)*C(  4)** 2 -    KM  2771
C     *     R( 16)*C( 27)*C( 30) + R( 26)*C(  9)*C( 24) + R( 33)*C(  7)   KM  2772
C     *     *C( 24) + R( 39)*C(  6)*C( 24) + R( 44)*C( 13)*C( 24)         KM  2773
C                                                                         KM  2774
      DO 120 J = 1,NK                                                     KM  2775
      YDOT(J+KS) = RATE(J)                                                KM  2776
120   CONTINUE                                                            KM  2777
C                                                                         KM  2778
130   CONTINUE                                                            KM  2779
      RETURN                                                              KM  2780
      END                                                                 KM  2781
```

```
      SUBROUTINE SCALE(Y1,Y2,YB,YT)                                      KM  2782
C                                                                        KM  2783
C         ESTABLISH MINIMUM AND MAXIMUM VALUES FOR PRINTER-PLOT  Y  AXISKM  2784
C                                                                        KM  2785
      DIMENSION Y(2)                                                     KM  2786
      DATA ONE,TEN/1.,10./                                               KM  2787
C                                                                        KM  2788
      BOT = Y1                                                           KM  2789
      TOP = Y2                                                           KM  2790
      IF(Y1.LT.Y2) GO TO 10                                              KM  2791
      TOP = Y1                                                           KM  2792
      BOT = Y2                                                           KM  2793
   10 CONTINUE                                                           KM  2794
      LA = 0                                                             KM  2795
      Y(1) = TOP                                                         KM  2796
      Y(2) = BOT                                                         KM  2797
      Z = TOP                                                            KM  2798
      DO 50 I=1,2                                                        KM  2799
      IF(Z.EQ.0.) GO TO 45                                               KM  2800
      SN = SIGN(ONE,Z)                                                   KM  2801
      A = ABS(Z)                                                         KM  2802
      LA = IFIX(ALOG10(A))                                               KM  2803
      IF(SN.LT.0.) GO TO 30                                              KM  2804
   20 DO 25 N=2,10                                                       KM  2805
      XN = FLOAT(N)                                                      KM  2806
      YT = SN*XN*TEN**LA                                                 KM  2807
      IF(Z.LT.YT) GO TO 40                                               KM  2808
   25 CONTINUE                                                           KM  2809
      GO TO 40                                                           KM  2810
   30 DO 35 N=1,9                                                        KM  2811
      XN = TEN - FLOAT(N)                                                KM  2812
      YT = SN*XN*TEN**LA                                                 KM  2813
      IF(Z.LT.YT) GO TO 40                                               KM  2814
   35 CONTINUE                                                           KM  2815
   40 Y(I) = YT                                                          KM  2816
   45 IF(I.EQ.2) GO TO 50                                                KM  2817
      Z = BOT                                                            KM  2818
   50 CONTINUE                                                           KM  2819
      YT = Y(1)                                                          KM  2820
      YB = Y(2) - TEN**LA                                                KM  2821
C                                                                        KM  2822
      RETURN                                                             KM  2823
      END                                                                KM  2824
```

```
      SUBROUTINE SKEDUL(X,T,NX,NINT,UPDELT,INTIM,TSTOP,DELTIN,FAIL,INTP)KM  2825
C                                                                        KM  2826
C     SUBROUTINE SKEDUL CALCULATES MEAN VALUES FOR A REGULAR UPDATE      KM  2827
C     SCHEDULE  FROM AN IRREGULAR SCHEDULE OR DIFFERENT INTERVAL SCHEDULKM  2828
C                                                                        KM  2829
C     X IS AN INPUT STEP FUNCTION WITH ARBITARY UPDATE INTERVALS         KM  2830
C     X IS RETURNED AS A STEP FUNCTION WITH A FIXED UPDATE INTERVAL      KM  2831
C     T IS THE TIME SCHEDULE ASSOCIATED WITH X ON INPUT                  KM  2832
C       IF T IS AVAILIABLE SET DELTIN EQUAL TO A NEGATIVE NUMBER         KM  2833
```

```
C       IT MUST BE MONOTONICALLY INCREASING                                KM  2834
C     NX IS THE NUMBER OF X VALUES INPUT                                   KM  2835
C  NINT IS THE NUMBER OF UPDATE INTERVALS OF THE RETURNED FUNCTION X       KM  2836
C UPDELT IS THE DESIRED UPDATE INTERVAL INPUT                              KM  2837
C INTIM IS THE INITIAL TIME TO START UPDATING                              KM  2838
C TSTOP IS THE FINAL TIME                                                  KM  2839
C DELTIN IS A FIXED UPDATE INTERVAL OF X INPUT (DELTIN .NE. UPDELT         KM  2840
C         IS USED WHEN T IS NOT AVAILIABLE                                 KM  2841
C  FAIL IS RETURN AS 'YES' IF T IS NOT MONOTONICALLY INCREASING            KM  2842
C  INTP IS A FLAG POSITIVE FOR INTERPOLATION, NEGATIVE FOR STEP-WISE       KM  2843
C              INTEGRATION                                                 KM  2844
C                                                                          KM  2845
      REAL INTIM                                                           KM  2846
      DIMENSION X(1), T(1), XO(200), TO(200)                               KM  2847
      DATA  YES  /3HYES/                                                   KM  2848
      DATA  RNEG  /2HNO/                                                   KM  2849
      FAIL = RNEG                                                          KM  2850
      IF(INTP.GT.0)   GO TO 330                                            KM  2851
C     INITIALIZE T IF NOT INPUT                                            KM  2852
      IF(DELTIN.LT.0.0)  GO TO 10                                          KM  2853
      T(1) = INTIM                                                         KM  2854
      DO 5 I=2,NX                                                          KM  2855
      T(I) = T(I-1) + DELTIN                                               KM  2856
  5   CONTINUE                                                             KM  2857
C     INITIALIZE TIME OUT SCHEDULE                                         KM  2858
  10  NINT = (TSTOP+ .01 -INTIM)/UPDELT + 1                                KM  2859
      TO(1) = INTIM                                                        KM  2860
      NP1 = NINT + 1                                                       KM  2861
      TO(2) =INTIM + UPDELT/2.                                             KM  2862
      DO 15 I=3,NP1                                                        KM  2863
      TO(I) = TO(I-1) + UPDELT                                             KM  2864
  15  CONTINUE                                                             KM  2865
C     CHECK FOR FIRST T                                                    KM  2866
      IF(T(1) .GE. TO(1))  GO TO 60                                        KM  2867
      DO 50 J=2,NX                                                         KM  2868
      IF(T(J) .LT. TO(1))  GO TO 50                                        KM  2869
      KL = J-1                                                             KM  2870
      GO TO 70                                                             KM  2871
  50  CONTINUE                                                             KM  2872
      K = NX +1                                                            KM  2873
      GO TO 320                                                            KM  2874
  60  KL=1                                                                 KM  2875
  70  T(KL) = TO(1)                                                        KM  2876
      T(NX+1) = TO(NP1) + .01                                              KM  2877
      IF(INTP.GT.0)  GO TO 330                                             KM  2878
      S=0.0                                                                KM  2879
C     FORWARD INTEGRATIOON LOOP TO DETERMINE MEAN VALUES OF NEW INTERVAL   KM  2880
      DO 300 I=1,NINT                                                      KM  2881
      IF(I.EQ.1)  GO TO 80                                                 KM  2882
      IF(T(KL).GE.TO(I+1))  GO TO 280                                      KM  2883
      IF(KL.GT.1)  S=X(KL-1)*(T(KL)- TO(I))                                KM  2884
  80  DO 200 K=KL,NX                                                       KM  2885
      IF(T(K+1).LT.T(K))  GO TO 320                                        KM  2886
      IF(T(K+1).GE.TO(I+1))  GO TO 160                                     KM  2887
      S = S + X(K)*(T(K+1)-T(K))                                           KM  2888
```

```
      GO TO 200                                                          KM  2889
  160 S = S + X(K)*(TO(I+1)-T(K))                                        KM  2890
      GO TO 250                                                          KM  2891
  200 CONTINUE                                                           KM  2892
C                                                                        KM  2893
  250 XO(I) = S/UPDELT                                                   KM  2894
      KL = K + 1                                                         KM  2895
      GO TO 300                                                          KM  2896
  280 XO(I) = X(K)                                                       KM  2897
  300 CONTINUE                                                           KM  2898
C                                                                        KM  2899
      CALL XMIT(NINT,XO,X)                                               KM  2900
      RETURN                                                             KM  2901
C                                                                        KM  2902
  330 CONTINUE                                                           KM  2903
      SMALL = 1.E-3                                                      KM  2904
      TO(1) = INTIM                                                      KM  2905
      NINT = IFIX((TSTOP - INTIM + SMALL)/UPDELT) + 1                    KM  2906
      NP1 = NINT + 1                                                     KM  2907
      KL = 1                                                             KM  2908
      KLP = 2                                                            KM  2909
      DO 340 I = 2,NP1                                                   KM  2910
      TO(I) = TO(I-1) + UPDELT                                           KM  2911
340   CONTINUE                                                           KM  2912
      X(NX+1) = X(NX)                                                    KM  2913
      DO 350 I = 1,NINT                                                  KM  2914
      TMX = T(KLP) - SMALL                                               KM  2915
      IF(TO(I).GT.TMX)  KL = KL + 1                                      KM  2916
      KLP = KL + 1                                                       KM  2917
      SLOPE = (X(KLP)-X(KL)) / (T(KLP)-T(KL))                           KM  2918
      XO(I) = X(KL) + SLOPE*(TO(I)-T(KL))                               KM  2919
  350 CONTINUE                                                           KM  2920
      CALL XMIT(NINT,XO,X)                                               KM  2921
      RETURN                                                             KM  2922
C                                                                        KM  2923
  320 WRITE(6,1) T(K+1)                                                  KM  2924
  1   FORMAT(1H1,34HRESCHEDULING FAILED AT TIME EQUAL  ,F8.2)            KM  2925
      FAIL = YES                                                         KM  2926
      RETURN                                                             KM  2927
      END                                                                KM  2928
```

```
      SUBROUTINE SOL (N, NDIM, A, B, IP)                                 KM  2929
C-----------------------------------------------------------------------KM  2930
C SOLUTION OF LINEAR SYSTEM, A*X = B .                                   KM  2931
C INPUT..                                                                KM  2932
C   N = ORDER OF MATRIX.                                                 KM  2933
C   NDIM = DECLARED DIMENSION OF ARRAY  A .                              KM  2934
C   A = TRIANGULARIZED MATRIX OBTAINED FROM DEC.                         KM  2935
C   B = RIGHT HAND SIDE VECTOR.                                          KM  2936
C   IP = PIVOT VECTOR OBTAINED FROM DEC.                                 KM  2937
C DO NOT USE IF DEC HAS SET IER .NE. 0.                                  KM  2938
C OUTPUT..                                                               KM  2939
C   B = SOLUTION VECTOR, X .                                             KM  2940
```

```
C-----------------------------------------------------------------------KM  2941
      INTEGER IP, N, NDIM                                               KM  2942
      INTEGER I, K, KB, KM1, KP1, M, NM1                                KM  2943
      DIMENSION A(NDIM, N), B(N), IP(N)                                 KM  2944
C                                                                       KM  2945
      IF (N .EQ. 1) GO TO 50                                            KM  2946
      NM1 = N - 1                                                       KM  2947
      DO 20 K = 1,NM1                                                   KM  2948
        KP1 = K + 1                                                     KM  2949
        M = IP(K)                                                       KM  2950
        T = B(M)                                                        KM  2951
        B(M) = B(K)                                                     KM  2952
        B(K) = T                                                        KM  2953
        DO 10 I = KP1,N                                                 KM  2954
 10       B(I) = B(I) + A(I,K)*T                                        KM  2955
 20     CONTINUE                                                        KM  2956
      DO 40 KB = 1,NM1                                                  KM  2957
        KM1 = N - KB                                                    KM  2958
        K = KM1 + 1                                                     KM  2959
        B(K) = B(K)/A(K,K)                                              KM  2960
        T = -B(K)                                                       KM  2961
        DO 30 I = 1,KM1                                                 KM  2962
 30       B(I) = B(I) + A(I,K)*T                                        KM  2963
 40     CONTINUE                                                        KM  2964
 50   B(1) = B(1)/A(1,1)                                                KM  2965
      RETURN                                                            KM  2966
C---------------------  END OF SUBROUTINE SOL  -------------------------KM  2967
      END                                                               KM  2968


      SUBROUTINE SOL282(A,B)                                            KM  2969
C     THIS SUBROUTINE SOLVES A SYSTEM OF 2 LINEAR EQUATIONS             KM  2970
      DIMENSION A(2,2), B(2), X(2)                                      KM  2971
      DET = A(1,1)*A(2,2) - A(1,2)*A(2,1)                               KM  2972
      X(1) = ( A(2,2)*B(1) - A(1,2)*B(2) )/DET                          KM  2973
      X(2) = (-A(2,1)*B(1) + A(1,1)*B(2) )/DET                          KM  2974
      B(1) = X(1)                                                       KM  2975
      B(2) = X(2)                                                       KM  2976
      RETURN                                                            KM  2977
      END                                                               KM  2978


      SUBROUTINE STEADY (Y,N)                                           KM  2979
C                                                                       KM  2980
C         H2O IS TREATED AT A CONSTANT USING NSTDY = 1                  KM  2981
C         THERE ARE NO STEADY STATE APPROXIMATIONS INCLUDED             KM  2982
C         IN THIS VERSION OF STEADY.                                    KM  2983
C                                                                       KM  2984
      RETURN                                                            KM  2985
      END                                                               KM  2986
```

```
      SUBROUTINE TEMPR(IT,TIME,T)                                       KM  2987
C                                                                       KM  2988
C     DETERMINES TEMPERATURE DEPENDENT REACTION RATES                   KM  2989
C                                                                       KM  2990
      DIMENSION T(1)                                                    KM  2991
      COMMON/CHEM2/ CONIN(40,5),      WTMOLE(40),     RATKON(55),       KM  2992
     1              RATEFF(55),       RATEV(2,5),     QRATE,            KM  2993
     2              NVRATE,           LOCVRT(2)                         KM  2994
      RT = 1.986*T(IT)                                                  KM  2995
C     HNO4 DISSOCIATION RATE                                            KM  2996
      RATKON(21) = 7.8E15 * EXP(-20700./RT)                             KM  2997
      RATEFF(21) = RATKON(21)                                           KM  2998
C     PAN DISSOCIATION RATE                                             KM  2999
      RATKON(37) = 1.17E17 * EXP(-26910./RT)                            KM  3000
      RATEFF(37) = RATKON(37)                                           KM  3001
      WRITE(10,1)  TIME, T(IT), RATKON(21), RATKON(37)                  KM  3002
      RETURN                                                            KM  3003
1     FORMAT(1H010X,40HTEMPERATURE DEPENDENT RATES' TIME =      ,       KM  3004
     1  ,F8.2,5X,6HTEMP =,F8.2,5X,7HRATES = ,2E13.3)                    KM  3005
      END                                                               KM  3006



      SUBROUTINE TIMEX                                                  KM  3007
      RETURN                                                            KM  3008
      END                                                               KM  3009



      SUBROUTINE TSTEP (Y, N0)                                          KM  3010
C                                                                       KM  3011
C-----------------------------------------------------------------------KM  3012
C     THIS VERSION OF TSTEP SOLVES A BLOCK TRI-DIAGONAL SYSTEM OF ODE'S KM  3013
C     AND    IS DESIGNED FOR MF = 21 ONLY                               KM  3014
C     MODIFICATIONS BY F.W. LURMANN   (7.8.77).                         KM  3015
C-----------------------------------------------------------------------KM  3016
C TSTEP PERFORMS ONE STEP OF THE INTEGRATION OF AN INITIAL VALUE        KM  3017
C PROBLEM FOR A SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS.              KM  3018
C-----------------------------------------------------------------------KM  3019
      INTEGER N0                                                        KM  3020
      INTEGER IPIV, JSTART, KFLAG, L, LMAX, METH, MF, N, NFE, NJE,      KM  3021
     1        NQ, NQINDX, NQUSED, NSTEP                                 KM  3022
      INTEGER I, IBACK, IER, IREDO, J, J1, J2, M, MFOLD, MIO,           KM  3023
     1        MITER, MITER1, NEWJ, NSTEPJ                               KM  3024
      INTEGER ISTEPJ, KFC, KFH, MAXCOR                                  KM  3025
C/24                                                                    KM  3026
      DIMENSION Y(N0, 6)                                                KM  3027
C                                                                       KM  3028
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART     KM  3029
      COMMON /EPCOM2/ YMAX(1)                                           KM  3030
      COMMON /EPCOM3/ ERROR(1)                                          KM  3031
      COMMON /EPCOM4/ SAVE1(1)                                          KM  3032
      COMMON /EPCOM5/ SAVE2(1)                                          KM  3033
      COMMON /EPCOM6/ PW(1)                                             KM  3034
      COMMON /EPCOM7/ IPIV(1)                                           KM  3035
```

```
      COMMON /EPCOM9/ HUSED,NQUSED,NSTEP,NFE,NJE                        KM  3036
      COMMON /EPCM10/ TAU(13),EL(13),TQ(5),LMAX,METH,NQ,L,NQINDX        KM  3037
      COMMON /EPCM12/ PRL(3600),  WORKR(30), DUPPER(4), DLOWER(4)       KM  3038
      INTEGER BCFLAG                                                    KM  3039
      COMMON/CHEM1/ NOSTAT,              NOSTM1,             NOREAC,    KM  3040
     1              NOSPEC,              NSTDY,              NK         KM  3041
      COMMON/CHEM2/ CONIN(40,5),         WTMOLE(40),         RATKON(55), KM  3042
     1              RATEFF(55),          RATEV(2,5),         QRATE,     KM  3043
     2              NVRATE,              LOCVRT(2)                      KM  3044
      COMMON/CHEM3/ ZEE(5),              DELZ(4),            HTCELL(6),  KM  3045
     1              TDELZ(2) ,           DFINIT(6),          SCALOW(4),  KM  3046
     2              DCOF(5),             FLXWAL(40),         FLXDGE(40), KM  3047
     3              BCFLAG(40),          DPRATE(10),         DEPOWR(10), KM  3048
     4              LOCDPF(10),          NDPFLX ,            SCALUP(4)   KM  3049
C                                                                       KM  3050
      DATA ISTEPJ /20/, KFC /-3/, KFH /-7/, MAXCOR /3/                  KM  3051
C(6                                                                     KM  3052
      DATA ADDON  /1.0E-6/,    BIAS1  /2.5E1/, BIAS2  /2.5E1/,          KM  3053
     1     BIAS3  /1.0E2/,     CRDOWN /0.1E0/, DELRC  /0.3E0/,          KM  3054
     2     ETACF  /0.25E0/,    ETAMIN /0.1E0/, ETAMXF /0.2E0/,          KM  3055
     3     ETAMX1 /1.0E4/,     ETAMX2 /1.0E1/, ETAMX3 /1.5E0/,          KM  3056
     4     ONEPSM /1.00001E0/, SHORT  /0.1E0/, THRESH /1.3E0/           KM  3057
      DATA ONE /1.0E0/, PT5 /0.5E0/, ZERO /0.0E0/                       KM  3058
      KFLAG = 0                                                         KM  3059
      TOLD = T                                                          KM  3060
      FLOTN =  FLOAT(N)                                                 KM  3061
      IF (JSTART .GT. 0) GO TO 200                                      KM  3062
      IF (JSTART .NE. 0) GO TO 150                                      KM  3063
C-----------------------------------------------------------------------KM  3064
C ON THE FIRST CALL, THE ORDER IS SET TO 1 AND THE INITIAL              KM  3065
C DERIVATIVES ARE CALCULATED.  ETAMAX IS THE MAXIMUM RATIO BY           KM  3066
C WHICH H CAN BE INCREASED IN A SINGLE STEP.  IT IS 1.E04 FOR THE       KM  3067
C FIRST STEP TO COMPENSATE FOR THE SMALL INITIAL H, THEN 10 FOR         KM  3068
C THE NEXT 10 STEPS, AND THEN 1.5 THEREAFTER.  IF A FAILURE             KM  3069
C OCCURS (IN CORRECTOR CONVERGENCE OR ERROR TEST), ETAMAX IS SET AT 1   KM  3070
C FOR THE NEXT INCREASE.  ETAMIN = .1 IS THE MINIMUM RATIO BY WHICH     KM  3071
C H CAN BE REDUCED ON ANY RETRY OF A STEP.                              KM  3072
C-----------------------------------------------------------------------KM  3073
      CALL DIFFUN (N, T, Y, SAVE1)                                      KM  3074
      DO 110 I = 1,N                                                    KM  3075
 110    Y(I,2) = H*SAVE1(I)                                            KM  3076
      METH = MF/10                                                      KM  3077
      MITER = MF - 10*METH                                              KM  3078
      MITER1 = MITER + 1                                                KM  3079
      MFOLD = MF                                                        KM  3080
      NQ = 1                                                            KM  3081
      L = 2                                                             KM  3082
      TAU(1) = H                                                        KM  3083
      PRL1 = ONE                                                        KM  3084
      RC = ZERO                                                         KM  3085
      ETAMAX = ETAMX1                                                   KM  3086
      NQINDX = 2                                                        KM  3087
      NSTEP = 0                                                         KM  3088
      NSTEPJ = 0                                                        KM  3089
      NFE = 1                                                           KM  3090
```

```
      NJE = 0                                                          KM  3091
      GO TO 200                                                        KM  3092
C-----------------------------------------------------------------------KM  3093
C IF THE USER HAS CHANGED H, THEN Y MUST BE RESCALED.  IF THE          KM  3094
C USER HAS CHANGED MITER, THEN NEWJ IS SET TO MITER TO FORCE           KM  3095
C THE PARTIAL DERIVATIVEES TO BE UPDATED, IF THEY ARE BEING USED.      KM  3096
C-----------------------------------------------------------------------KM  3097
 150  IF (MF .EQ. MFOLD) GO TO 170                                     KM  3098
      MIO = MITER                                                      KM  3099
      METH = MF/10                                                     KM  3100
      MITER = MF - 10*METH                                             KM  3101
      MFOLD = MF                                                       KM  3102
      IF (MITER .EQ. MIO) GO TO 170                                    KM  3103
      NEWJ = MITER                                                     KM  3104
      MITER1 = MITER + 1                                               KM  3105
 170  IF (H .EQ. HOLD) GO TO 200                                       KM  3106
      ETA = H/HOLD                                                     KM  3107
      H = HOLD                                                         KM  3108
      IREDO = 3                                                        KM  3109
      GO TO 185                                                        KM  3110
C(2                                                                    KM  3111
  180 ETA = AMAX1(ETA,HMIN/ ABS(H),ETAMIN)                             KM  3112
  185 ETA = AMIN1(ETA,HMAX/ ABS(H),ETAMAX)                             KM  3113
      R1 = ONE                                                         KM  3114
      DO 190 J = 2,L                                                   KM  3115
        R1 = R1*ETA                                                    KM  3116
        DO 190 I = 1,N                                                 KM  3117
 190      Y(I,J) = Y(I,J)*R1                                           KM  3118
      H = H*ETA                                                        KM  3119
      RC = RC*ETA                                                      KM  3120
      IF (IREDO .EQ. 0) GO TO 690                                      KM  3121
C-----------------------------------------------------------------------KM  3122
C THIS SECTION COMPUTES THE PREDICTED VALUES BY EFFECTIVELY            KM  3123
C MULTIPLYING THE Y ARRAY BY THE PASCAL TRIANGLE MATRIX.  THEN         KM  3124
C COSET IS CALLED TO OBTAIN EL, THE VECTOR OF COEFFICIENTS OF          KM  3125
C LENGTH NQ + 1.  RC IS THE RATIO OF NEW TO OLD VALUES OF THE          KM  3126
C COEFFICIENT H/EL(2).  WHEN RC DIFFERS FROM 1 BY MORE THAN            KM  3127
C DELRC, NEWJ IS SET TO MITER TO FORCE THE PARTIAL DERIVATIVES         KM  3128
C TO BE UPDATED, IF USED.  DELRC IS 0.3.  IN ANY CASE, THE PARTIAL     KM  3129
C DERIVATIVES ARE UPDATED AT LEAST EVERY 20-TH STEP.                   KM  3130
C-----------------------------------------------------------------------KM  3131
 200  T = T + H                                                        KM  3132
      DO 210 J1 = 1,NQ                                                 KM  3133
        DO 210 J2 = J1,NQ                                              KM  3134
          J = (NQ + J1) - J2                                           KM  3135
          DO 210 I = 1,N                                               KM  3136
 210        Y(I,J) = Y(I,J) + Y(I,J+1)                                 KM  3137
      CALL COSET                                                       KM  3138
      BND = FLOTN*(TQ(4)*EPS)**2                                       KM  3139
      RL1 = ONE/EL(2)                                                  KM  3140
      RC = RC*(RL1/PRL1)                                               KM  3141
      PRL1 = RL1                                                       KM  3142
      IF (NSTEP .GE. NSTEPJ+ISTEPJ) NEWJ = MITER                       KM  3143
C(1                                                                    KM  3144
      DRC =  ABS(RC-ONE)                                               KM  3145
```

```
      IF (DRC .LE. DELRC) GO TO 215                                     KM  3146
      NEWJ = MITER                                                      KM  3147
      CRATE = ONE                                                       KM  3148
      RC = ONE                                                          KM  3149
      GO TO 220                                                         KM  3150
  215 IF ((MITER .NE. 0) .AND. (DRC .NE. ZERO)) CRATE = ONE             KM  3151
C-----------------------------------------------------------------------KM  3152
C UP TO 3 CORRECTOR ITERATIONS ARE TAKEN.  A CONVERGENCE TEST IS MADE   KM  3153
C ON THE ROOT MEAN SQUARE NORM OF EACH CORRECTION, USING BND, WHICH     KM  3154
C IS DEPENDENT ON EPS.  THE SUM OF THE CORRECTIONS IS ACCUMULATED IN    KM  3155
C THE VECTOR ERROR.  THE Y ARRAY IS NOT ALTERED IN THE CORRECTOR        KM  3156
C LOOP.  THE UPDATED Y VECTOR IS STORED TEMPORARILY IN SAVE1.           KM  3157
C-----------------------------------------------------------------------KM  3158
  220 DO 230 I = 1,N                                                    KM  3159
  230   ERROR(I) = ZERO                                                 KM  3160
      M = 0                                                             KM  3161
      CALL DIFFUN (N, T, Y, SAVE2)                                      KM  3162
      NFE = NFE + 1                                                     KM  3163
      IF (NEWJ .LE. 0) GO TO 290                                        KM  3164
C-----------------------------------------------------------------------KM  3165
C IF INDICATED, THE MATRIX P = I - H*RL1*J IS REEVALUATED BEFORE        KM  3166
C STARTING THE CORRECTOR ITERATION.  NEWJ IS SET TO 0 AS AN             KM  3167
C INDICATOR THAT THIS HAS BEEN DONE.  IF MITER = 1 OR 2, P IS           KM  3168
C COMPUTED AND PROCESSED IN PSET.  IF MITER = 3, THE MATRIX  IS         KM  3169
C P = I - H*RL1*D, WHERE D IS A DIAGONAL MATRIX.  RL1 IS 1/EL(2).       KM  3170
C-----------------------------------------------------------------------KM  3171
      NEWJ = 0                                                          KM  3172
      RC = ONE                                                          KM  3173
      NJE = NJE + 1                                                     KM  3174
      NSTEPJ = NSTEP                                                    KM  3175
      GO TO (250, 240, 260), MITER                                      KM  3176
  240 NFE = NFE + N                                                     KM  3177
  250 CON = -H*RL1                                                      KM  3178
      CALL PSET(Y, N0, CON, MITER, IER)                                 KM  3179
      IF (IER .NE. 0) GO TO 420                                         KM  3180
      GO TO 350                                                         KM  3181
  260 R = RL1*SHORT                                                     KM  3182
      DO 270 I = 1,N                                                    KM  3183
  270   PW(I) = Y(I,1) + R*(H*SAVE2(I) - Y(I,2))                        KM  3184
      CALL DIFFUN(N, T, PW, SAVE1)                                      KM  3185
      NFE = NFE + 1                                                     KM  3186
      HRL1 = H*RL1                                                      KM  3187
      DO 280 I = 1,N                                                    KM  3188
        R0 = H*SAVE2(I) - Y(I,2)                                        KM  3189
        PW(I) = ONE                                                     KM  3190
        D = SHORT*R0 - H*(SAVE1(I) - SAVE2(I))                          KM  3191
        SAVE1(I) = ZERO                                                 KM  3192
C(2                                                                     KM  3193
        IF ( ABS(R0) .LT. UROUND*YMAX(I)) GO TO 280                     KM  3194
        IF ( ABS(D) .EQ. ZERO) GO TO 420                                KM  3195
        PW(I) = SHORT*R0/D                                              KM  3196
        SAVE1(I) = PW(I)*RL1*R0                                         KM  3197
  280   CONTINUE                                                        KM  3198
      GO TO 370                                                         KM  3199
  290 GO TO (295, 350, 350, 310), MITER1                                KM  3200
```

```
C-----------------------------------------------------------------------KM  3201
C IN THE CASE OF FUNCTIONAL ITERATION, Y IS UPDATED DIRECTLY FROM       KM  3202
C THE RESULT OF THE LAST DIFFUN CALL.                                   KM  3203
C-----------------------------------------------------------------------KM  3204
 295  D = ZERO                                                          KM  3205
      DO 300 I = 1,N                                                    KM  3206
        R = RL1*(H*SAVE2(I) - Y(I,2))                                   KM  3207
        D = D + ((R - ERROR(I))/YMAX(I))**2                             KM  3208
        SAVE1(I) = Y(I,1) + R                                           KM  3209
 300    ERROR(I) = R                                                    KM  3210
      GO TO 400                                                         KM  3211
C-----------------------------------------------------------------------KM  3212
C IN THE CASE OF A CHORD METHOD, THE RESIDUAL -G(Y SUB N(M))            KM  3213
C IS COMPUTED AND THE LINEAR SYSTEM WITH THAT AS RIGHT-HAND SIDE        KM  3214
C AND P AS COEFFICIENT MATRIX IS SOLVED, USING THE LU DECOMPOSITION     KM  3215
C OF P IF MITER = 1 OR 2.  IF MITER = 3 THE SCALAR H*RL1 IS UPDATED.    KM  3216
C-----------------------------------------------------------------------KM  3217
 310  PHRL1 = HRL1                                                      KM  3218
      HRL1 = H*RL1                                                      KM  3219
      IF (HRL1 .EQ. PHRL1) GO TO 330                                    KM  3220
      R = HRL1/PHRL1                                                    KM  3221
      DO 320 I = 1,N                                                    KM  3222
        D = ONE - R*(ONE - ONE/PW(I))                                   KM  3223
C(1                                                                     KM  3224
        IF ( ABS(D) .EQ. ZERO) GO TO 440                                KM  3225
C)1                                                                     KM  3226
CS      IF (ABS(D) .EQ. ZERO) GO TO 440                                 KM  3227
C/4                                                                     KM  3228
 320    PW(I) = ONE/D                                                   KM  3229
 330  DO 340 I = 1,N                                                   KM  3230
 340    SAVE1(I) = PW(I)*(RL1*H*SAVE2(I) - (RL1*Y(I,2) + ERROR(I)))     KM  3231
      GO TO 370                                                         KM  3232
 350  DO 360 I = 1,N                                                   KM  3233
 360    SAVE1(I) = RL1*H*SAVE2(I) - (RL1*Y(I,2) + ERROR(I))             KM  3234
      CALL BLKSOL(PW,NK,NOSTAT,SAVE1,IPIV,DLOWER,DUPPER,PWL,WORKR,BCFLAGKM  3235
     1)                                                                 KM  3236
CERT    CALL SOL (N,  NO,  PW,  SAVE1,  IPIV)                          KM  3237
 370  D = ZERO                                                          KM  3238
      DO 380 I = 1,N                                                    KM  3239
        ERROR(I) = ERROR(I) + SAVE1(I)                                  KM  3240
        D = D + (SAVE1(I)/YMAX(I))**2                                   KM  3241
 380    SAVE1(I) = Y(I,1) + ERROR(I)                                    KM  3242
C-----------------------------------------------------------------------KM  3243
C TEST FOR CONVERGENCE.  IF M .GT. 0, AN ESTIMATE OF THE SQUARE OF      KM  3244
C THE CONVERGENCE RATE CONSTANT IS STORED IN CRATE, AND THIS IS USED    KM  3245
C IN THE TEST.                                                          KM  3246
C-----------------------------------------------------------------------KM  3247
C(2                                                                     KM  3248
  400 IF (M .NE. 0) CRATE = AMAX1(CRDOWN*CRATE,D/D1)                    KM  3249
      IF (D*AMIN1(ONE,CRATE) .LE. BND) GO TO 450                        KM  3250
      D1 = D                                                            KM  3251
      M = M + 1                                                         KM  3252
      IF (M .EQ. MAXCOR) GO TO 410                                      KM  3253
      CALL DIFFUN (N, T, SAVE1, SAVE2)                                  KM  3254
      GO TO (295, 350, 350, 310), MITER1                                KM  3255
```

```
C-----------------------------------------------------------------------KM  3256
C THE CORRECTOR ITERATION FAILED TO CONVERGE IN 3 TRIES. IF PARTIAL      KM  3257
C DERIVATIVES ARE INVOLVED BUT ARE NOT UP TO DATE, THEY ARE              KM  3258
C REEVALUATED FOR THE NEXT TRY.  OTHERWISE THE Y ARRAY IS RESTORED       KM  3259
C TO ITS VALUES BEFORE PREDICTION, AND H IS REDUCED,                     KM  3260
C IF POSSIBLE.  IF NOT, A NO-CONVERGENCE EXIT IS TAKEN.                  KM  3261
C-----------------------------------------------------------------------KM  3262
 410  NFE = NFE + MAXCOR - 1                                            KM  3263
      IF (NEWJ .EQ. -1) GO TO 440                                       KM  3264
 420  T = TOLD                                                          KM  3265
      ETAMAX = ONE                                                      KM  3266
      DO 430 J1 = 1,NQ                                                  KM  3267
        DO 430 J2 = J1,NQ                                               KM  3268
          J = (NQ + J1) - J2                                            KM  3269
          DO 430 I = 1,N                                                KM  3270
 430        Y(I,J) = Y(I,J) - Y(I,J+1)                                  KM  3271
C(1                                                                     KM  3272
      IF ( ABS(H) .LE. HMIN*ONEPSM) GO TO 680                           KM  3273
      ETA = ETACF                                                       KM  3274
      IREDO = 1                                                         KM  3275
      GO TO 180                                                         KM  3276
 440  NEWJ = MITER                                                      KM  3277
      GO TO 220                                                         KM  3278
C-----------------------------------------------------------------------KM  3279
C THE CORRECTOR HAS CONVERGED.  NEWJ IS SET TO -1 IF PARTIAL             KM  3280
C DERIVATIVES WERE USED, TO SIGNAL THAT THEY MAY NEED UPDATING ON        KM  3281
C SUBSEQUENT STEPS.  THE ERROR TEST IS MADE AND CONTROL PASSES TO        KM  3282
C STATEMENT 500 IF IT FAILS.                                             KM  3283
C-----------------------------------------------------------------------KM  3284
 450  IF (MITER .NE. 0) NEWJ = -1                                       KM  3285
      NFE = NFE + M                                                     KM  3286
      D = ZERO                                                          KM  3287
      DO 460 I = 1,N                                                    KM  3288
 460    D = D + (ERROR(I)/YMAX(I))**2                                   KM  3289
      E = FLOTN*(TQ(2)*EPS)**2                                          KM  3290
      IF (D .GT. E) GO TO 500                                           KM  3291
C-----------------------------------------------------------------------KM  3292
C AFTER A SUCCESSFUL STEP, THE Y ARRAY, TAU, NSTEP, AND NQINDX ARE       KM  3293
C UPDATED, AND A NEW VALUE OF H AT ORDER NQ IS COMPUTED.                 KM  3294
C THE VECTOR TAU CONTAINS THE NQ + 1 MOST RECENT VALUES OF H.            KM  3295
C A CHANGE IN NQ UP OR DOWN BY 1 IS CONSIDERED IF NQINDX = 0.            KM  3296
C IF NQINDX = 1 AND NQ .LT. MAXDER, THEN ERROR IS SAVED                  KM  3297
C FOR USE IN A POSSIBLE ORDER INCREASE ON THE NEXT STEP.                 KM  3298
C A CHANGE IN H OR NQ IS MADE ONLY OF THE INCREASE IN H                  KM  3299
C IS BY A FACTOR OF AT LEAST 1.3.                                        KM  3300
C IF NOT, NQINDX IS SET TO 2 TO PREVENT TESTING FOR THAT MANY            KM  3301
C STEPS.  IF NQ IS CHANGED, NQINDX IS SET TO NQ + 1 (NEW VALUE).         KM  3302
C-----------------------------------------------------------------------KM  3303
      KFLAG = 0                                                         KM  3304
      IREDO = 0                                                         KM  3305
      NSTEP = NSTEP + 1                                                 KM  3306
      HUSED = H                                                         KM  3307
      NQUSED = NQ                                                       KM  3308
      DO 470 IBACK = 1,NQ                                               KM  3309
        I = L - IBACK                                                   KM  3310
```

```
 470    TAU(I+1) = TAU(I)                                                KM  3311
      TAU(1) = H                                                         KM  3312
      DO 480 J = 1,L                                                     KM  3313
        DO 480 I = 1,N                                                   KM  3314
 480      Y(I,J) = Y(I,J) + ERROR(I)*EL(J)                               KM  3315
      NQINDX = NQINDX + 1                                                KM  3316
      IF ((L .EQ. LMAX) .OR. (NQINDX .NE. 1)) GO TO 495                  KM  3317
      DO 490 I = 1,N                                                     KM  3318
 490    Y(I,LMAX) = ERROR(I)                                            KM  3319
      CONP = TQ(5)                                                       KM  3320
 495  IF (ETAMAX .NE. ONE) GO TO 520                                     KM  3321
      IF (NQINDX .LT. 2) NQINDX = 2                                      KM  3322
      GO TO 690                                                          KM  3323
C-----------------------------------------------------------------------KM  3324
C THE ERROR TEST FAILED.  KFLAG KEEPS TRACK OF MULTIPLE FAILURES.        KM  3325
C T AND THE Y ARRAY ARE RESTORED TO THEIR PREVIOUS VALUES.  A NEW        KM  3326
C H FOR A RETRY OF THE STEP IS COMPUTED.  THE ORDER IS KEPT FIXED.       KM  3327
C-----------------------------------------------------------------------KM  3328
 500  KFLAG = KFLAG - 1                                                  KM  3329
      T = TOLD                                                           KM  3330
      DO 510 J1 = 1,NQ                                                   KM  3331
        DO 510 J2 = J1,NQ                                                KM  3332
          J = (NQ + J1) - J2                                             KM  3333
          DO 510 I = 1,N                                                 KM  3334
 510        Y(I,J) = Y(I,J) - Y(I,J+1)                                   KM  3335
      NEWJ = MITER                                                       KM  3336
      ETAMAX = ONE                                                       KM  3337
C(1                                                                      KM  3338
      IF ( ABS(H) .LE. HMIN*ONEPSM) GO TO 660                            KM  3339
      IF (KFLAG .LE. KFC) GO TO 630                                      KM  3340
      IREDO = 2                                                          KM  3341
C COMPUTE RATIO OF NEW H TO CURRENT H AT THE CURRENT ORDER. ------------KM  3342
C(1                                                                      KM  3343
  520 FLOTL =  FLOAT(L)                                                  KM  3344
      ETAQ = ONE/((BIAS2*D/E)**(PT5/FLOTL) + ADDON)                      KM  3345
      IF ((NQINDX .NE. 0) .OR. (KFLAG .NE. 0)) GO TO 580                 KM  3346
      ETAQM1 = ZERO                                                      KM  3347
      IF (NQ .EQ. 1) GO TO 540                                           KM  3348
C COMPUTE RATIO OF NEW H TO CURRENT H AT THE CURRENT ORDER LESS ONE. ---KM  3349
      D = ZERO                                                           KM  3350
      DO 530 I = 1,N                                                     KM  3351
 530    D = D + (Y(I,L)/YMAX(I))**2                                     KM  3352
      EDN = FLOTN*(TQ(1)*EPS)**2                                         KM  3353
      ETAQM1 = ONE/((BIAS1*D/EDN)**(PT5/(FLOTL - ONE)) + ADDON)          KM  3354
 540  ETAQP1 = ZERO                                                      KM  3355
      IF (L .EQ. LMAX) GO TO 560                                         KM  3356
C COMPUTE RATIO OF NEW H TO CURRENT H AT CURRENT ORDER PLUS ONE. -------KM  3357
      CNQUOT = (TQ(5)/CONP)*(H/TAU(2))**L                                KM  3358
      D = ZERO                                                           KM  3359
      DO 550 I = 1,N                                                     KM  3360
 550    D = D + ((ERROR(I) - CNQUOT*Y(I,LMAX))/YMAX(I))**2              KM  3361
      EUP = FLOTN*(TQ(3)*EPS)**2                                         KM  3362
      ETAQP1 = ONE/((BIAS3*D/EUP)**(PT5/(FLOTL + ONE)) + ADDON)          KM  3363
 560  NQINDX = 2                                                         KM  3364
      IF (ETAQ .GE. ETAQP1) GO TO 570                                    KM  3365
```

```
      IF (ETAQP1 .GT. ETAQM1) GO TO 600                                 KM  3366
      GO TO 590                                                         KM  3367
  570 IF (ETAQ .LT. ETAQM1) GO TO 590                                   KM  3368
  580 IF ((ETAQ .LT. THRESH) .AND. (KFLAG .EQ. 0)) GO TO 690            KM  3369
      ETA = ETAQ                                                        KM  3370
      IF ((KFLAG .LE. -2) .AND. (ETA .GT. ETAMXF)) ETA = ETAMXF         KM  3371
      GO TO 180                                                         KM  3372
  590 IF (ETAQM1 .LT. THRESH) GO TO 690                                 KM  3373
      CALL ADJUST (Y, NQ)                                               KM  3374
      L = NQ                                                            KM  3375
      NQ = NQ - 1                                                       KM  3376
      ETA = ETAQM1                                                      KM  3377
      NQINDX = L                                                        KM  3378
      GO TO 180                                                         KM  3379
  600 IF (ETAQP1 .LT. THRESH) GO TO 690                                 KM  3380
      NQ = L                                                            KM  3381
      ETA = ETAQP1                                                      KM  3382
      L = L + 1                                                         KM  3383
      DO 610 I = 1,N                                                    KM  3384
  610   Y(I,L) = ZERO                                                   KM  3385
      NQINDX = L                                                        KM  3386
      GO TO 180                                                         KM  3387
C-----------------------------------------------------------------------KM  3388
C CONTROL REACHES THIS SECTION IF 3 OR MORE CONSECUTIVE FAILURES        KM  3389
C HAVE OCCURRED.  IT IS ASSUMED THAT THE ELEMENTS OF THE Y ARRAY        KM  3390
C HAVE ACCUMULATED ERRORS OF THE WRONG ORDER.  THE ORDER IS REDUCED     KM  3391
C BY ONE, IF POSSIBLE.  THEN H IS REDUCED BY A FACTOR OF 0.1 AND       KM  3392
C THE STEP IS RETRIED.  AFTER A TOTAL OF 7 CONSECUTIVE FAILURES,       KM  3393
C AN EXIT IS TAKEN WITH KFLAG = -2.                                    KM  3394
C-----------------------------------------------------------------------KM  3395
  630 IF (KFLAG .EQ. KFH) GO TO 670                                     KM  3396
      IF (NQ .EQ. 1) GO TO 640                                          KM  3397
      ETA = ETAMIN                                                      KM  3398
      CALL ADJUST (Y, NQ)                                               KM  3399
      L = NQ                                                            KM  3400
      NQ = NQ - 1                                                       KM  3401
      NQINDX = L                                                        KM  3402
      GO TO 180                                                         KM  3403
C(1                                                                     KM  3404
  640 ETA = AMAX1(ETAMIN,HMIN/ ABS(H))                                  KM  3405
      H = H*ETA                                                         KM  3406
      CALL DIFFUN (N, T, Y, SAVE1)                                      KM  3407
      NFE = NFE + 1                                                     KM  3408
      DO 650 I = 1,N                                                    KM  3409
  650   Y(I,2) = H*SAVE1(I)                                             KM  3410
      NQINDX = 10                                                       KM  3411
      GO TO 200                                                         KM  3412
C-----------------------------------------------------------------------KM  3413
C ALL RETURNS ARE MADE THROUGH THIS SECTION.  H IS SAVED IN HOLD        KM  3414
C TO ALLOW THE CALLER TO CHANGE H ON THE NEXT STEP.                     KM  3415
C-----------------------------------------------------------------------KM  3416
  660 KFLAG = -1                                                        KM  3417
      GO TO 700                                                         KM  3418
  670 KFLAG = -2                                                        KM  3419
      GO TO 700                                                         KM  3420
```

```
 680  KFLAG = -3                                                        KM  3421
      GO TO 700                                                         KM  3422
 690  ETAMAX = ETAMX3                                                   KM  3423
      IF (NSTEP .LE. 10) ETAMAX = ETAMX2                                KM  3424
 700  HOLD = H                                                          KM  3425
      JSTART = NQ                                                       KM  3426
      RETURN                                                            KM  3427
C-----------------------  END OF SUBROUTINE TSTEP  ----------------------KM  3428
      END                                                               KM  3429



      SUBROUTINE UNMIXR                                                 KM  3430
      RETURN                                                            KM  3431
      END                                                               KM  3432



      SUBROUTINE UPFLX1(TIME,J)                                         KM  3433
C                                                                       KM  3434
C        *UPFLX1* UPDATES AREA SOURCE EMISSION FLUXES                   KM  3435
C                                                                       KM  3436
C         THIS ROUTINE HAS BEEN MODIFIED TO PARTITION A SINGLE          KM  3437
C         ALDEHYDE FLUX INTO 60% HCHO AND 40% RCHO.                     KM  3438
C                                                                       KM  3439
      COMMON/FLUXES/FLXIN( 7,200),  FLXTIM(200),   NFLUX                KM  3440
      COMMON/CHEM3/ ZEE(5),              DELZ(4),            HTCELL(6),  KM  3441
     1              TDELZ(2) ,           DFINIT(6),          SCALOW(4),  KM  3442
     2              DCOF(5),             FLXWAL(40),         FLXDGE(40), KM  3443
     3              BCFLAG(40),          DPRATE(10),         DEPOWR(10), KM  3444
     4              LOCDPF(10),          NDPFLX ,            SCALUP(4)   KM  3445
      COMMON /CHEM5/  REACT(20,55),SPEC(40),LOCFLX(10),                 KM  3446
     1                NASFLX,FLXW1(30),FLXW2(30)                        KM  3447
      INTEGER BCFLAG                                                    KM  3448
      IF(NASFLX.LE.0)  RETURN                                           KM  3449
      WRITE(10,1)TIME                                                   KM  3450
      JP1 =MIN0(J+1,NFLUX)                                              KM  3451
      DO 100 I = 1,NASFLX                                               KM  3452
      K = LOCFLX(I)                                                     KM  3453
      FLXW1(K) = FLXIN(I,J)                                             KM  3454
      FLXW2(K) = FLXIN(I,JP1)                                           KM  3455
100   CONTINUE                                                          KM  3456
C                                                                       KM  3457
C          SPLIT ALDEHYDES                                            KM  3458
C                                                                       KM  3459
      FLXW1(7) = 0.40*FLXW1(6)                                          KM  3460
      FLXW2(7) = 0.40*FLXW2(6)                                          KM  3461
      FLXW1(6) = 0.60*FLXW1(6)                                          KM  3462
      FLXW2(6) = 0.60*FLXW2(6)                                          KM  3463
C                                                                       KM  3464
C       SPLIT NOX                                                       KM  3465
C                                                                       KM  3466
      FLXW1(2) = 0.10*FLXW1(1)                                          KM  3467
      FLXW2(2) = 0.10*FLXW2(1)                                          KM  3468
      FLXW1(1) = 0.90*FLXW1(1)                                          KM  3469
```

```
      FLXW2(1) = 0.90*FLXW2(1)                                          KM  3470
      DO 200 I = 1,NASFLX                                               KM  3471
      K = LOCFLX(I)                                                     KM  3472
      WRITE(10,2) SPEC(K),FLXW1(K)                                      KM  3473
200   CONTINUE                                                          KM  3474
      K = 7                                                             KM  3475
      WRITE(10,2) SPEC(K),FLXW1(K)                                      KM  3476
      K = 2                                                             KM  3477
      WRITE(10,2) SPEC(K),FLXW1(K)                                      KM  3478
1     FORMAT(1H0,10X,38HAREA SOURCE FLUXES UPDATED AT TIME   ,F10.2)    KM  3479
2     FORMAT(1H ,10X,A4,3H = ,2E15.5)                                   KM  3480
      RETURN                                                            KM  3481
      END                                                               KM  3482



      SUBROUTINE UPRAT2 (T,IK1,RATEV,NOSTAT,NVRATE,CLOUDY)              KM  3483
C                                                                       KM  3484
C     -------- UPDATES THE ARRAY OF VARIABLE RATE CONSTANTS (RATEV)     KM  3485
C              FROM RATK1 + RATK2                                       KM  3486
C     -------- IT INITIALIZES UPPER ELEVATION RATE CONSTANTS  TO        KM  3487
C              THE RATE CONSTANT AT THE SURFACE IF HIRATE = NO          KM  3488
C                                                                       KM  3489
      DIMENSION   RATEV(2,5)                                            KM  3490
      COMMON/CHEM4/RATK1(100,5),       RATK2(100,5),      RLAT,         KM  3491
     1             RLONG,              TMZONE,            SUNTIM,       KM  3492
     2             HIRATE,             JDATE,             NRATE         KM  3493
      DATA YES,NEG,MORE,END/3HYES,2HNO,4HMORE,3HEND/                    KM  3494
      IF(HIRATE.EQ.YES)  GO TO 5                                        KM  3495
      DO 3 J=2,NOSTAT                                                   KM  3496
      RATK1(IK1,J) = RATK1(IK1,1)                                       KM  3497
      IF(NVRATE.LT.2)  GO TO 3                                          KM  3498
      RATK2(IK1,J) = RATK2(IK1,1)                                       KM  3499
  3   CONTINUE                                                          KM  3500
C                                                                       KM  3501
  5   DO 10 J=1,NOSTAT                                                  KM  3502
      RATEV(1,J) = RATK1(IK1,J)*CLOUDY                                  KM  3503
      IF(NVRATE.LT.2)  GO TO 10                                         KM  3504
      RATEV(2,J) = RATK2(IK1,J)*CLOUDY                                  KM  3505
  10  CONTINUE                                                          KM  3506
      WRITE(10,20) T  ,   ((RATEV(I,J),J=1,NOSTAT),I =1,NVRATE)         KM  3507
  20  FORMAT (1H0,10X,41HVARIABLE RATE CONSTANTS UPDATED AT TIME ,F6.1, KM  3508
     . /,11X,3HK1=,7X,5G12.4,/,11X,8HK(HCHO)=,2X,5G12.4)                KM  3509
      RETURN                                                            KM  3510
      END                                                               KM  3511
```

```
      SUBROUTINE UPSORC(TIME,IPS,NOSTAT,SPEC)                                KM  3512
C                                                                            KM  3513
C         UPDATES POINT SOURCE EMISSION RATES                                KM  3514
C                                                                            KM  3515
      DIMENSION SPEC(1)                                                      KM  3516
      COMMON /PS1/   TPASS(200),     PS(7,5,75),    FRACT(3),                KM  3517
     1               NPTSR,          NPSFLX,        LOCPSF(7)               KM  3518
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT        KM  3519
C                                                                            KM  3520
      IF(NPSFLX.LE.0)  GO TO 5                                               KM  3521
      IF(IPS.LE.NPTSR)  GO TO 10                                             KM  3522
5     CALL XMIT(-150,0.0,PSRATE)                                             KM  3523
      RETURN                                                                 KM  3524
10    CONTINUE                                                               KM  3525
      IPSP1 = MIN0(IPS+1,NPTSR)                                              KM  3526
      DO 20 J = 1,NOSTAT                                                     KM  3527
      DO 20 I = 1,NPSFLX                                                     KM  3528
      K = LOCPSF(I)                                                          KM  3529
      PSR1(K,J) = PS(I,J,IPS)                                                KM  3530
      PSR2(K,J) = PS(I,J,IPSP1)                                              KM  3531
20    CONTINUE                                                               KM  3532
C                                                                            KM  3533
C        SPLIT ALDEHYDES AND NOX                                             KM  3534
C                                                                            KM  3535
      DO 30 J = 1,NOSTAT                                                     KM  3536
      PSR1(7,J) = 0.40*PSR1(6,J)                                             KM  3537
      PSR2(7,J) = 0.40*PSR2(6,J)                                             KM  3538
      PSR1(6,J) = 0.60*PSR1(6,J)                                             KM  3539
      PSR2(6,J) = 0.60*PSR2(6,J)                                             KM  3540
      PSR1(2,J) = 0.10*PSR1(1,J)                                             KM  3541
      PSR2(2,J) = 0.10*PSR2(1,J)                                             KM  3542
      PSR1(1,J) = 0.90*PSR1(1,J)                                             KM  3543
      PSR2(1,J) = 0.90*PSR2(1,J)                                             KM  3544
30    CONTINUE                                                               KM  3545
C                                                                            KM  3546
C       WRITE UPDATES ON TAPE10                                              KM  3547
C                                                                            KM  3548
      WRITE(10,1)  TIME                                                      KM  3549
      DO 40 K = 1,NPSFLX                                                     KM  3550
      ID = LOCPSF(K)                                                         KM  3551
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)                           KM  3552
40    CONTINUE                                                               KM  3553
      ID = 2                                                                 KM  3554
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)                           KM  3555
      ID = 7                                                                 KM  3556
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)                           KM  3557
      RETURN                                                                 KM  3558
1     FORMAT(1H0,10X,37HPOINT SOURCE RATES UPDATED AT TIME =  ,F8.2)         KM  3559
2     FORMAT(1H ,10X,A4,4H =    ,5E12.3)                                     KM  3560
      END                                                                    KM  3561
```

```
      SUBROUTINE FMINF(F,NF,FMIN,NMIN)                                  UL     1
C                                                                       UL     2
C        FMINF LOCATES THE MINIMUM VALUE AMONG NF CONSECUTIVE           UL     3
C        MEMBERS OF THE F ARRAY AND RETURNS BOTH THE VALUE FMIN         UL     4
C        AND ITS ARRAY INDEX NMIN.                                      UL     5
C                                                                       UL     6
      DIMENSION F(NF)                                                   UL     7
C                                                                       UL     8
      A = F(1)                                                          UL     9
      DO 100 N=2,NF                                                     UL    10
      A = AMIN1(F(N),A)                                                 UL    11
  100 CONTINUE                                                          UL    12
      DO 110 N=1,NF                                                     UL    13
      IF(A-F(N)) 110,120,120                                            UL    14
  110 CONTINUE                                                          UL    15
      N = NF                                                            UL    16
  120 NMIN = N                                                          UL    17
      FMIN = A                                                          UL    18
      RETURN                                                            UL    19
C                                                                       UL    20
      END                                                               UL    21
```

```
      SUBROUTINE MCHAR (IFC,FROM,ITC,TO,NCHR)                           UL    22
C                                                                       UL    23
C     MCHAR MOVES A STRING OF CHARACTERS FROM ONE WORD TO ANOTHER.       UL    24
C                                                                       UL    25
C      IFC  = POSITION OF CHARACTER TO MOVE                             UL    26
C      FROM = SOURCE WORD                                               UL    27
C      ITC  =  POSITION OF CHARACTER TO MOVE TO                         UL    28
C      TO   = DESTINATION WORD                                          UL    29
C      NCHR = NUMBER OF CHARACTERS TO MOVE                              UL    30
C                                                                       UL    31
C     NOTE THAT FROM AND TO ARE EITHER 1 OR 4-CHARACTER WORDS           UL    32
C                                                                       UL    33
      DIMENSION  CHARF(4),  CHART(4)                                    UL    34
      DATA CHARF, CHART /8*1H   /                                       UL    35
C     TEMPORARY LOGICAL UNIT DEFINED                                    UL    36
      LUN = 4                                                           UL    37
C     READ CONTENTS OF SOURCE WORD FROM TO TEMPORARY ARRAY              UL    38
      WRITE(LUN,1)  FROM                                                UL    39
    1 FORMAT(A4)                                                        UL    40
      BACKSPACE LUN                                                     UL    41
      READ(LUN,2) CHARF                                                 UL    42
    2 FORMAT(4A1)                                                       UL    43
      BACKSPACE LUN                                                     UL    44
      WRITE(LUN,1)  TO                                                  UL    45
      BACKSPACE LUN                                                     UL    46
      READ(LUN,2) (CHART(I),I=1,4)                                      UL    47
C     TRANSFER CONTENTS TO TEMPORARY ARRAY                              UL    48
      DO 10  I = 1,NCHR                                                 UL    49
      CHART(ITC+I-1) = CHARF(IFC+I-1)                                   UL    50
   10 CONTINUE                                                          UL    51
      BACKSPACE LUN                                                     UL    52
```

```
      WRITE(LUN,2) (CHART(I),I=1,4)                                    UL    53
      BACKSPACE LUN                                                    UL    54
      READ(LUN,1)  TO                                                  UL    55
      RETURN                                                           UL    56
      END                                                              UL    57



      SUBROUTINE MDATE(I)                                              UL    58
      DIMENSION I(1)                                                   UL    59
      DATA IBLK /4H    /                                               UL    60
      I(1) = IBLK                                                      UL    61
      I(2) = IBLK                                                      UL    62
      RETURN                                                           UL    63
      END                                                              UL    64



      SUBROUTINE NEWPAG (TITLE,LSKIP,IDATE)                            UL    65
      DIMENSION     TITLE(20),          IDATE(1)                       UL    66
      DATA NPAGE,  LOUT/0,  6/                                         UL    67
      WRITE(LOUT,1)                                                    UL    68
      IF (LSKIP.EQ.0) GO TO 110                                        UL    69
      DO 100 I=1,LSKIP                                                 UL    70
      WRITE(LOUT,2)                                                    UL    71
  100 CONTINUE                                                         UL    72
  110 NPAGE=NPAGE+1                                                    UL    73
      CALL SECOND (A)                                                  UL    74
      WRITE(LOUT,3)  TITLE,  A,  IDATE(1),  NPAGE                      UL    75
      RETURN                                                           UL    76
C                                                                      UL    77
    1 FORMAT (1H1)                                                     UL    78
    2 FORMAT (1H )                                                     UL    79
    3 FORMAT (1H ,1X,20A4,1X,9HCP TIME =,F8.2,2X,5HDATE ,2X,A10,2X,5HPAGUL    80
     1E   ,I4)                                                        UL    81
      END                                                              UL    82



      SUBROUTINE PREDAT                                                UL    83
C                                                                      UL    84
C        *PREDAT*  TRANSFERS DATA-CARD-IMAGES FROM INPUT(TAPE5)        UL    85
C        TO TAPE3 AND OUTPUT(TAPE6).                                   UL    86
C        PREDAT RETURNS WHEN IT READS THE WORD *MORE* IN COL. 1-4      UL    87
C        OR WHEN IT FINDS AN EOF.                                      UL    88
C                                                                      UL    89
      DIMENSION IMAGE(20)                                              UL    90
      DATA LDT,LOUT,ARO/3,6,1H+/                                       UL    91
      DATA LIN /5/                                                     UL    92
      DATA  MORE/4HMORE/                                               UL    93
      REWIND LDT                                                       UL    94
      LYN = 0                                                          UL    95
100   CONTINUE                                                         UL    96
C     ******* REMOVE THE NEXT TWO CARDS FOR UNIVAC/IBM *******         UL    97
      READ (LIN,1) IMAGE                                               UL    98
```

```
      IF (EOF(LIN)) 120,110                                                  UL    99
C     ****** INSERT THE NEXT CARD INSTEAD *****                              UL   100
C                                                                            UL   101
C     READ(LIN,1,END=120) IMAGE                                              UL   102
110   CONTINUE                                                               UL   103
      IF (MOD(LYN,40).EQ.0) WRITE (LOUT,2)                                   UL   104
      IF (MOD(LYN,10).EQ.0) WRITE (LOUT,3) (K,K=1,80,10),(ARO,K=1,8)         UL   105
      LYN = LYN+1                                                            UL   106
      WRITE (LOUT,4) LYN,IMAGE                                               UL   107
      WRITE (LDT,1) IMAGE                                                    UL   108
      IF (IMAGE(1).NE.MORE) GO TO 100                                        UL   109
120   CONTINUE                                                               UL   110
      REWIND LDT                                                             UL   111
      WRITE (LOUT,5)                                                         UL   112
      RETURN                                                                 UL   113
1     FORMAT (20A4)                                                          UL   114
2     FORMAT (1H1,8X,20HIMAGES OF DATA-CARDS)                                UL   115
3     FORMAT (1H010X8I10/14X6HCARD  A1,8(9H.........A1))                     UL   116
4     FORMAT (I18,2X,20A4)                                                   UL   117
5     FORMAT (1H1)                                                           UL   118
      END                                                                    UL   119



      SUBROUTINE SECOND(A)                                                   UL   120
      A = 0.0                                                                UL   121
      RETURN                                                                 UL   122
      END                                                                    UL   123



      SUBROUTINE SETPLT(A,B,C,D)                                             UL   124
C                                                                            UL   125
C     GENERAL-PURPOSE PRINTER-PLOT ROUTINE                                   UL   126
C                                                                            UL   127
      COMMON /IMAG/ IMAGE                                                    UL   128
      DIMENSION  IMAGE(26,51),                LABX(5,6),LABEL(5)             UL   129
      EQUIVALENCE  (IC,CI)                                                   UL   130
      LOGICAL CONFRM                                                         UL   131
      DATA          MAXX,MAXY,ARO/104,51,1H+/                                UL   132
      DATA          IBLNK,IDASH     /4H     ,4H----/                         UL   133
C                                                                            UL   134
C     NCHAR IS THE NUMBER OF CHARACTERS PER WORD                             UL   135
      DATA          NCHAR/4/                                                 UL   136
C                                                                            UL   137
C                                                                            UL   138
C                                                                            UL   139
C-----FOR THIS ENTRY, CALL SETPLOT (X-LOW,                                   UL   140
C                                   X-HIGH,                                  UL   141
C                                   Y-LOW,                                   UL   142
C                                   Y-HIGH)                                  UL   143
C         THIS IS THE INITIALIZING CALL - THE ARGUMENTS DEFINE THE           UL   144
C              ENDS OF THE AXES TO BE PLOTTED                                UL   145
C                                                                            UL   146
C     ENTRY SETPLOT                                                          UL   147
```

```
      CONFRM = .FALSE.                                                  UL   148
100   DO 110 J = 1,51                                                   UL   149
      DO 110 I = 1,26                                                   UL   150
      IMAGE(I,J) = IBLNK                                                UL   151
110   CONTINUE                                                          UL   152
      DO 120 I = 1,26                                                   UL   153
120   IMAGE(I,1) = IDASH                                                UL   154
      DO 130 J = 1,MAXY                                                 UL   155
      CALL MCHAR(1,1HI,1,IMAGE(1,J),1)                                  UL   156
130   CONTINUE                                                          UL   157
      DX = (B-A)/100.                                                   UL   158
      DY = (D-C)/(MAXY-1)                                               UL   159
      XORG = A                                                          UL   160
      YORG = C                                                          UL   161
      IF (.NOT.CONFRM) GO TO 140                                        UL   162
      IF (DX.LT..6*DY) DX = .6*DY                                       UL   163
      IF (DY.LT.5./3.*DX) DY = 5./3.*DX                                 UL   164
140   KVAL = IBLNK                                                      UL   165
      KGO = 19                                                          UL   166
      DO 160 J = 1,6                                                    UL   167
      VAL = XORG+DX*20*(J-1)                                            UL   168
      GO TO 230                                                         UL   169
150   LABX(1,J) = LABEL(1)                                              UL   170
      LABX(2,J) = LABEL(2)                                              UL   171
      LABX(3,J) = LABEL(3)                                              UL   172
      LABX(4,J) = LABEL(4)                                              UL   173
160   LABX(5,J) = LABEL(5)                                              UL   174
      RETURN                                                            UL   175
C                                                                       UL   176
C                                                                       UL   177
C-----THIS ENTRY IS IDENTICAL TO SETPLOT, EXCEPT IT IS USED FOR         UL   178
C                   PLOTTING X AND Y TO SAME SCALE                      UL   179
C                   (A COMPUTED CIRCLE IS PLOTTED AS A CIRCLE)          UL   180
C                                                                       UL   181
C     ENTRY ISOPLT(A,B,C,D)   FOR UNIVAC OR IBM .                       UL   182
      ENTRY ISOPLT                                                      UL   183
      CONFRM = .TRUE.                                                   UL   184
      GO TO 100                                                         UL   185
C                                                                       UL   186
C                                                                       UL   187
C-----FOR THIS ENTRY, CALL GETDXDY (DX,                                 UL   188
C                                   DY)                                 UL   189
C         USE THIS ENTRY IF YOU NEED TO KNOW THE X- AND Y-INCREMENTS    UL   190
C              WHICH HAVE BEEN COMPUTED BY ISOPLOT                      UL   191
C                                                                       UL   192
C     ENTRY GTDXDY (A,B)   FOR THE UNIVAC OR IBM                        UL   193
      ENTRY GTDXDY                                                      UL   194
      A = DX                                                            UL   195
      B = DY                                                            UL   196
      RETURN                                                            UL   197
C                                                                       UL   198
C                                                                       UL   199
C-----FOR THIS ENTRY, CALL PLOTPNT (X-COORD,                            UL   200
C                                   Y-COORD,                            UL   201
C                                   CHARACTER)                          UL   202
```

```
C          CHARACTER MUST BE THE FIRST ONE IN THE WORD, E.G., 1HA.       UL   203
C         THIS IS THE ENTRY WHICH ENTERS A POINT IN THE PLOT             UL   204
C                                                                        UL   205
C     ENTRY PLTPNT(A,B,C)   FOR THE UNIVAC OR IBM                        UL   206
      ENTRY PLTPNT                                                       UL   207
      MGO = 1                                                            UL   208
      CI = C                                                             UL   209
170   JX = (A-XORG)/DX+1.5                                               UL   210
      JY = (B-YORG)/DY+1.5                                               UL   211
      IF (JX.GT.MAXX.OR.JX.LE.0.OR.JY.GT.MAXY.OR.JY.LE.0) RETURN         UL   212
      IR = JX/NCHAR+1                                                    UL   213
      JZ = MOD(JX,NCHAR)                                                 UL   214
      IF (JZ.GT.0) GO TO 180                                             UL   215
      IR = IR-1                                                          UL   216
      JZ = NCHAR                                                         UL   217
180   CONTINUE                                                           UL   218
      IF (MGO.EQ.2) GO TO 190                                            UL   219
      CALL MCHAR(1,CI,JZ,IMAGE(IR,JY),1)                                 UL   220
      RETURN                                                             UL   221
C                                                                        UL   222
C                                                                        UL   223
C-----FOR THIS ENTRY, CALL GETPNT (X=COORD,                              UL   224
C                                  Y=COORD,                              UL   225
C                                   CHARACTER)                           UL   226
C          CHARACTER IS RETURNED IN THE FIRST POSITION OF VARIABLE =C=.  UL   227
C                  USE THIS ENTRY TO FIND OUT THE CHARACTER              UL   228
C                  NOW AT (X,Y)                                          UL   229
C                                                                        UL   230
C     ENTRY GETPNT(A,B,C)   FOR THE UNIVAC OR IBM                        UL   231
      ENTRY GETPNT                                                       UL   232
      MGO = 2                                                            UL   233
      C = 0.                                                             UL   234
      GO TO 170                                                          UL   235
190   CALL MCHAR(JZ,IMAGE(IR,JY),1,C,1)                                  UL   236
      RETURN                                                             UL   237
C                                                                        UL   238
C                                                                        UL   239
C-----FOR THIS ENTRY, CALL PLOTXP1 (CHARACTER)                           UL   240
C          CHARACTER MUST BE THE FIRST ONE IN THE WORD, E.G., 1HA.       UL   241
C         USE THIS ENTRY TO ADD A CHARACTER TO THE RIGHT OF              UL   242
C              THE LAST MARK ENTERED                                     UL   243
C                                                                        UL   244
C     ENTRY PLTPX1(A)   FOR THE UNIVAC OR IBM                            UL   245
      ENTRY PLTPX1                                                       UL   246
      JX = JX+1                                                          UL   247
      IF (JX.GT.MAXX) RETURN                                             UL   248
      CI = A                                                             UL   249
      IR = JX/NCHAR+1                                                    UL   250
      JZ = MOD(JX,NCHAR)                                                 UL   251
      IF (JZ.GT.0) GO TO 200                                             UL   252
      IR = IR-1                                                          UL   253
      JZ = NCHAR                                                         UL   254
200   CONTINUE                                                           UL   255
      CALL MCHAR(1,CI,JZ,IMAGE(IR,JY),1)                                 UL   256
      RETURN                                                             UL   257
```

```
C                                                                       UL   258
C                                                                       UL   259
C-----FOR THIS ENTRY, CALL PLOTYM1 (CHARACTER)                          UL   260
C          CHARACTER MUST BE THE FIRST ONE IN THE WORD, E.G., 1HA.       UL   261
C         USE THIS ENTRY TO ADD A CHARACTER BELOW                       UL   262
C              THE LAST MARK ENTERED                                    UL   263
C                                                                       UL   264
C     ENTRY PLTYM1(A)   FOR THE UNIVAC OR IBM                           UL   265
      ENTRY PLTYM1                                                      UL   266
      JY = JY-1                                                         UL   267
      IF (JY.LE.0) RETURN                                               UL   268
      CI = A                                                            UL   269
      CALL MCHAR(1,CI,JZ,IMAGE(IR,JY),1)                                UL   270
      RETURN                                                            UL   271
C                                                                       UL   272
C                                                                       UL   273
C         YOU MUST CALL PLOTOUT TO GET PLOT PRINTED                     UL   274
C-----FOR THIS ENTRY, CALL PLOTOUT (PRINT-TAPE)                         UL   275
C                                                                       UL   276
C     ENTRY PLTOUT(A)      FOR THE UNIVAC OR IBM                        UL   277
      ENTRY PLTOUT                                                      UL   278
      CI = A                                                            UL   279
      IF (IC.LE.0.OR.IC.GT.49) IC = 6                                   UL   280
      WRITE (IC,1)                                                      UL   281
      KVAL = IARO                                                       UL   282
      KGO = 14                                                          UL   283
      DO 220 JYY = 1,MAXY                                               UL   284
      JY = MAXY+1-JYY                                                   UL   285
      VAL = YORG+DY*(JY-1)                                              UL   286
      LABEL(1) = IBLNK                                                  UL   287
      LABEL(2) = IBLNK                                                  UL   288
      LABEL(3) = IBLNK                                                  UL   289
      LABEL(4) = IBLNK                                                  UL   290
      LABEL(5) = IBLNK                                                  UL   291
      IF (MOD(JYY,10).EQ.1) GO TO 230                                   UL   292
210   WRITE (IC,2) LABEL,(IMAGE(I,JY),I=1,26)                           UL   293
220   CONTINUE                                                          UL   294
      WRITE (IC,3) (ARO,I=1,6),LABX                                     UL   295
      RETURN                                                            UL   296
C                                                                       UL   297
230   V = ABS(VAL)                                                      UL   298
      IF (V.LT.1.E-76) GO TO 240                                        UL   299
      IF (V.GT.9.E4.OR.V.LT.1.E-2) GO TO 250                            UL   300
240   WRITE (4,4) VAL,KVAL                                              UL   301
      GO TO 260                                                         UL   302
250   WRITE (4,5) VAL,KVAL                                              UL   303
260   CONTINUE                                                          UL   304
      BACKSPACE 4                                                       UL   305
      READ (4,6) LABEL                                                  UL   306
      IF (KGO.EQ.19) GO TO 150                                          UL   307
      IF (KGO.EQ.14) GO TO 210                                          UL   308
      STOP                                                              UL   309
C                                                                       UL   310
1     FORMAT (1H1)                                                      UL   311
2     FORMAT (31A4)                                                     UL   312
```

```
3     FORMAT (20X,A1, 5(19X,A1)/8X,30A4/)                                UL   313
4     FORMAT (F17.4,2X,A1)                                               UL   314
5     FORMAT (E17.4,2X,A1)                                               UL   315
6     FORMAT (5A4)                                                       UL   316
      END                                                                UL   317



      SUBROUTINE SOLAR (SLA,SLO,TZ,IY,IM,ID,TIME,D,NV)                   UL   318
C***                                                                     UL   319
C***    SLA...  LATITUDE (DEG)  SOUTH = MINUS                            UL   320
C***    SLO...  LONGITUDE (DEG)  EAST = MINUS                            UL   321
C***    TZ...  TIME ZONE                                                 UL   322
C***           ALSO INCLUDES FRACTION IF LOCAL TIME IS NOT               UL   323
C***            STANDARD MERIDIAN TIME.  E.G. POONA, INDIA 5.5           UL   324
C***    IY..  YEAR                                                       UL   325
C***    IM..  MONTH                                                      UL   326
C***    ID..  DAY                                                        UL   327
C***    TIME.. LOCAL STANDARD TIME IN HOURS AND MINUTES.                 UL   328
C***           1 30 PM = 1330  ** STANDARD TIME **                       UL   329
C***    D..  RETURNED VALUE                                              UL   330
C***    NV..  VALUE TO BE RETURNED, SELECTED AS FOLLOWS....              UL   331
C***          1...  DECLINATION (DEG.)                                   UL   332
C***          2...  EQUATION OF TIME ADJUSTMENT (HRS.)                   UL   333
C***          3...  TRUE SOLAR TIME (HRS.)                               UL   334
C***          4...  HOUR ANGLE (DEG.)                                    UL   335
C***          5...  SOLAR ELEVATION (DEG.)                              UL   336
C***          6...  OPTICAL AIRMASS                                      UL   337
C***    0 ) NV ) 7.  OTHERWISE, D = 9999.                                UL   338
C***                                                                     UL   339
      DIMENSION MD(11)                                                   UL   340
      DATA MD/31,29,31,30,31,30,2*31,30,31,30/                           UL   341
      DATA A,B,C,SIGA/0.15,3.885,1.253,279.9348/                         UL   342
      RAD=572957.75913E-4                                                UL   343
      SDEC=39784.988432E-5                                               UL   344
      RE=1.                                                              UL   345
      IF(SLO.LT.0.) RE=-1.                                               UL   346
      KZ=TZ                                                              UL   347
      TC=(TZ-KZ)*RE                                                      UL   348
      TZZ=KZ*RE                                                          UL   349
      SLB=SLA/RAD                                                        UL   350
      K=ID                                                               UL   351
      TIMH=TIME/100.                                                     UL   352
      I=TIMH                                                             UL   353
      TIMLOC=(TIMH-I)/0.6+I+TC                                           UL   354
      IMC=IM-1                                                           UL   355
      IF(IMC.LT.1) GOTO2                                                 UL   356
      DO1I=1,IMC                                                         UL   357
    1 K=K+MD(I)                                                          UL   358
    2 LEAP=1                                                             UL   359
      NL=MOD(IY,4)                                                       UL   360
      IF(NL.LT.1) LEAP=2                                                 UL   361
      SMER=TZZ*15.                                                       UL   362
      TK=((SMER-SLO)*4.)/60.                                             UL   363
      KR=1                                                               UL   364
```

```
      IF(K.GE.61.AND.LEAP.LT.2) KR=2                                    UL   365
      DAD=(TIMLOC+TZZ)/24.                                              UL   366
      DAD=DAD+K-KR                                                      UL   367
      DF=DAD*360./365.242                                               UL   368
      DE=DF/RAD                                                         UL   369
      DESIN=SIN(DE)                                                     UL   370
      DECOS=COS(DE)                                                     UL   371
      DESIN2=SIN(DE*2.)                                                 UL   372
      DECOS2=COS(DE*2.)                                                 UL   373
      SIG=SIGA+DF+1.914827*DESIN-0.079525*DECOS+0.019938*DESIN2-0.00162*UL   374
     $DECOS2                                                            UL   375
      SIG=SIG/RAD                                                       UL   376
      DECSIN=SDEC*SIN(SIG)                                              UL   377
      EFFDEC= ASIN(DECSIN)                                              UL   378
      IF(NV.NE.1) GOTO10                                                UL   379
      D=EFFDEC*RAD                                                      UL   380
      RETURN                                                            UL   381
   10 EQT=0.12357*DESIN-0.004289*DECOS+0.153809*DESIN2+0.060783*DECOS2  UL   382
      IF(NV.NE.2) GOTO11                                                UL   383
      D=EQT                                                             UL   384
      RETURN                                                            UL   385
   11 TST=TK+TIMLOC-EQT                                                 UL   386
      IF(NV.NE.3) GOTO12                                                UL   387
      D=TST                                                             UL   388
      IF(D.LT.0.) D=D+24.                                               UL   389
      IF(D.GE.24.) D=D-24.                                              UL   390
      RETURN                                                            UL   391
   12 HRANGL=ABS(TST-12.)*15.                                           UL   392
      IF(NV.NE.4) GOTO13                                                UL   393
      D=HRANGL                                                          UL   394
      RETURN                                                            UL   395
   13 HRANGL=HRANGL/RAD                                                 UL   396
      SOLSIN=DECSIN*SIN(SLB)+COS(EFFDEC)*COS(SLB)*COS(HRANGL)           UL   397
      SOLEL= ASIN(SOLSIN)*RAD                                           UL   398
      IF(NV.NE.5) GOTO14                                                UL   399
      D=SOLEL                                                           UL   400
      RETURN                                                            UL   401
   14 IF(NV.NE.6) GOTO8                                                 UL   402
      IF(SOLEL.LE.0.) GOTO8                                             UL   403
      TK=SOLEL+8                                                        UL   404
      E=1./TK**C                                                        UL   405
      D=1./(A*E+SOLSIN)                                                 UL   406
      RETURN                                                            UL   407
    8 D=9999.                                                           UL   408
      RETURN                                                            UL   409
      END                                                               UL   410


      SUBROUTINE XMIT(N,A,B)                                            UL   411
C                                                                       UL   412
C         IF  N  POSITIVE, TRANSMITS  N  WORDS FROM  A  TO  B           UL   413
C         IF  N  NEGATIVE, TRANSMITS  A  TO  N  WORDS OF  B             UL   414
C                                                                       UL   415
      DIMENSION A(2),B(2)                                               UL   416
C                                                                       UL   417
      IF (N) 100,120,120                                                UL   418
100   K = IABS(N)                                                       UL   419
      DO 110 I = 1,K                                                    UL   420
      B(I) = A(1)                                                       UL   421
110   CONTINUE                                                          UL   422
      RETURN                                                            UL   423
C                                                                       UL   424
120   DO 130 I = 1,N                                                    UL   425
      B(I) = A(I)                                                       UL   426
130   CONTINUE                                                          UL   427
      RETURN                                                            UL   428
C                                                                       UL   429
      END                                                               UL   430
```

APPENDIX D

AN IMPROVED PHOTOCHEMICAL MECHANISM AND RELATED MODIFICATIONS FOR THE CHEMICAL-DIFFUSION MODULE

1. Description of Chemical Mechanism and Related Modifications

Concurrent with this study, an improved chemical mechanism was developed by ERT for use in the trajectory model. This appendix describes the new mechanism and related changes in the chemical-diffusion module for its use. Changes in the FORTRAN source code and data sets for the KEMOD program are included.

The chemical reactions and chemical species included in the improved mechanism are listed in Tables D-1 and D-2. The major differences between it and the mechanism described in previous sections are as follows:

1) A distinction is made between ethylene and higher olefinic hydrocarbon compounds to account for their differences in reactivity.

2) The ozone-olefin mechanism reflects the formation of more stable products and fewer radical species.

3) The aromatic hydrocarbon photooxidation mechanism is more reliable.

The development of this mechanism and the results for its evaluation relative to smog chamber data are described in Lloyd et al. 1979.

In adapting this mechanism to the atmosphere, new kinetic data and new assumptions have been employed in determining certain chemical reaction rate constants. The rate constants for photolysis of nitrogen dioxide, nitrous acid, formaldehyde, and higher aldehydes by solar ultraviolet radiation have been updated using new solar actinic flux, quantum yield, and absorption cross-section data (Dermerjian et al. 1979, Moortgat et al. 1978). The new clear-sky photolysis rates are shown in Tables D-3 through D-6 as a function of solar zenith angle and elevation.

TABLE D-1

THE PHOTOCHEMICAL REACTION MECHANISM

	Reactions	Rate Constant
1	NO2 + HV = 0 + NO	Radiation Dependent
2	0 + 02 + M = 03 + M	4.12E+06
3	03 + NO = NO2 + 02	2.50E+01
4	NO + NO2 + N20 = 2HONO	2.20E-09
5	2HONO = NO + NO2 + H20	1.40E-03
6	HONO + HV = OH + NO	Radiation Dependent
7	OH + NO + M = HONO + M	1.44E+04
8	OH + NO2 + M = HN03 + M	1.44E+04
9	OH + CO = HO2 + CO2	4.40E+02
10	HO2 + NO = NO2 + OH	1.20E+04
11	HO2 + NO2 = HN04	1.71E+03
12	2H02 = H202 + 02	3.61E+03
13	NO2 + 03 = NO3 + 02	5.00E-02
14	NO3 + NO = 2NO2	2.70E+04
15	NO3 + NO2 = N205	9.30E+02
16	N205 + H20 = 2HN03	1.00E-06
17	N205 = NO3 + NO2	Temperature Dependent
18	OH + ALKE = AO2	Elevation Dependent
19	OH + C2H4 = AO2	Elevation Dependent
20	AO2 + NO = NO2 + AO	2.90E+04
21	AO + 02 = .5RCHO + 1.5HCHO + HO2	4.10E+05
22	HN04 = HO2 + NO2	Temperature Dependent
23	0 + ALKE = .3EPOX + .3RCHO + .4H02 + .4RO2	Elevation Dependent
24	0 + C2H4 = .3EPOX + .3RCHO + .4H02 + .4RO2 + .4CO	Elevation Dependent
25	03 + ALKE = .5HCHO + .4RCHO + .4HD + .4RD + .1R02	Elevation Dependent
26	03 + C2H4 = HCHO + .8HD + .4H02 + .2CO2	Elevation Dependent
27	HD + NO = HCHO + NO2	2.90E+04

TABLE D-1 (Continued)

	Reactions	Rate Constant
28	RD + NO = RCHO + NO2	2.90E+04
29	HD + NO2 = HCHO + NO3	1.90E+04
30	RD + NO2 = RCHO + NO3	1.90E+04
31	HD + HCHO = OZID	1.00E+01
32	HD + RCHO = OZID	1.00E+01
33	RD + HCHO = OZID	1.00E+01
34	RD + RCHO = OZID	1.00E+01
35	OH + PA = H2O + PAO2	3.80E+03
36	PAO2 + NO = NO2 + .85PAO + .15RO2	2.90E+04
37	PAO2 + NO = NTRA	2.60E+03
38	PAO = RO2 + .5HCHO + .5RCHO	1.40E+05
39	RO2 + NO = NO2 + PAO	2.90E+04
40	PAO + O2 = RCHO + HO2	6.70E+04
41	PAO + NO2 = .85NTRA + .15RCHO + .15HONO	2.30E+03
42	OH + RCHO = RCO3 + H2O	2.20E+04
43	RCHO + HV = RO2 + HO2 + CO	Radiation Dependent
44	RCO3 + NO = CO2 + NO2 + RO2	2.90E+04
45	RCO3 + NO2 = PAN	1.70E+04
46	PAN = RCO3 + NO2	Temperature Dependent
47	HCHO + HV = HO2 + CO	Radiation Dependent
48	HCHO + OH = H2O + HO2 + CO	1.60E+04
49	RO2 + HO2 = RO2H + O2	4.20E+03
50	OH + AR = HO2 + AC	1.67E+04
51	OH + AR = ARO	3.33E+03
52	OH + AC = ARP + HO2	4.90E+04
53	ARO + NO = NO2 + ARIN	2.90E+04
54	ARO + NO2 = ANO2	1.90E+04
55	ARO + HO2 = AC + H2O2	1.80E+03
56	ARIN = RO2 + OH	1.00E-01
57	AR + OH = H2O + ABO	5.00E+03
58	ABO + NO = NO2 + HO2 + ACHO	2.90E+04

TABLE D-1 (Continued)

	Reactions	Rate Constant
59	$SO_2 + OH \rightarrow SO_4$	1.76E+03
60	$SO_2 + HO_2 \rightarrow SO_4\ OH$	3.00E-02
61	$SO_2 + HD \rightarrow SO_4 + HCHO$	2.90E+03
62	$SO_2 + RD \rightarrow SO_4 + RCHO$	2.90E+03
63	$SO_2 + RO_2 \rightarrow SO_4 + PAO$	8.00E+00
64	$SO_2 + AO_2 \rightarrow SO_4 + AO$	8.00E+00

TABLE D-2

CHEMICAL SPECIES SYMBOL DEFINITIONS

Species	Symbol Designation	Species No.
AO	Alkoxy radical equivalent of AO_2	17
AO_2	Product of OH addition to olefin in the presence of O_2	18
AR	Aromatic hydrocarbons	12
AC	Product of OH addition to aromatic hydrocarbon followed by H atom abstraction by O_2.	14
ARO	Product of addition of OH to a cresol in the presence of O_2	13
ARP	Product of OH addition to AC followed by H atom abstraction by O_2.	-
ABO	Product of H-abstraction from side chain alkyl group of benzene ring followed by addition of O_2 to radical formed	16
ARIN	Intermediate formed from reaction of ARO with NO, forming NO_2	15
ACHO	Aromatic aldehyde	39
ANO_2	Aromatic nitro compound	38
AROH	Cresol	-
CO	Carbon monoxide	5
CO_2	Carbon dioxide	-
HD	Criegee intermediate ($HCHO_2$)	25
RD	Criegee intermediate ($RCHO_2$)	26
EPOX	Epoxide formed from O atom addition to olefin	-
HCHO	Formaldehyde	6
HONO	Nitrous acid	4
HNO_3	Nitric acid	34

TABLE D-2 (Continued)

Species	Symbol Designation	Species No.
HNO_4	Pernitric acid, HO_2HO_2	20
HO_2	Hydroperoxyl radical	22
H_2O	Water	33
H_2O_2	Hydrogen peroxide	36
HV	Photon	-
M	Any third body, such as N_2 or O_2	-
NO	Nitric oxide	1
NO_2	Nitrogen dioxide	2
NO_3	Nitrate radical	19
N_2O_5	Dinitrogen pentoxide	21
NTRA	Organic nitrate	37
O	Oxygen atom (ground state)	32
O_2	Oxygen	-
O_3	Ozone	3
OH	Hydroxyl radical	31
OZID	Ozonides	35
C_2H_4	Ethene	9
ALKE	Alkenes other than ethene	8
PA	Alkanes (paraffinic hydrocarbons)	10
PAN	Peroxyacetyl Nitrate	11
PAO	Alkoxy radical formed by PA	27
PAO_2	Alkyl peroxy radical from the O_2 addition to the radical formed by H-abstraction from a paraffinic hydrocarbon	28
R	Generalized alkyl group (e.g., C_2H_5, C_3H_7, etc.)	-
RCHO	Aldehydes other than formaldehyde	7
RCO_3	Acyl peroxy radical	24

TABLE D-2 (Continued)

Species	Symbol Designation	Species No.
RO	Alkoxyl radical	-
RO_2	Alkyl peroxy radical	23
RO_2H	Product of disproportionation between HO_2 and RO_2	-
SO_2	Sulfur dioxide	29
SO_4	Sulfate	30

TABLE D-3

ATMOSPHERIC PHOTODISSOCIATION RATES FOR RCHO (PER MINUTE)

ELEVATIONS IN METERS

Zenith Angle	0.0	150.0	360.0	640.0	980.0	1380.0	1840.0	4210.0
0.0	1.380E-03	1.494E-03	1.602E-03	1.716E-03	1.836E-03	1.944E-03	2.046E-03	2.382E-03
10.0	1.362E-03	1.476E-03	1.584E-03	1.698E-03	1.806E-03	1.914E-03	2.202E-03	2.358E-03
20.0	1.296E-03	1.404E-03	1.512E-03	1.620E-03	1.734E-03	1.842E-03	1.938E-03	2.274E-03
30.0	1.176E-03	1.284E-03	1.386E-03	1.494E-03	1.602E-03	1.704E-03	1.800E-03	2.130E-03
40.0	1.014E-03	1.116E-03	1.212E-03	1.308E-03	1.410E-03	1.506E-03	1.596E-03	1.920E-03
50.0	8.040E-04	8.820E-04	9.720E-04	1.062E-03	1.146E-03	1.236E-03	1.314E-03	1.614E-03
60.0	5.520E-04	6.180E-04	6.840E-04	7.500E-04	8.160E-04	8.820E-04	9.540E-04	1.212E-04
70.0	2.820E-04	3.240E-04	3.600E-04	4.020E-04	4.440E-04	4.860E-04	5.280E-04	7.080E-04
78.0	1.140E-04	1.260E-04	1.380E-04	1.560E-04	1.740E-04	1.920E-04	2.100E-04	3.000E-04
86.0	1.800E-05	2.400E-05	2.400E-05	2.400E-05	3.000E-05	3.000E-05	3.000E-05	3.600E-05

TABLE D-4

ATMOSPHERIC PHOTODISSOCIATION RATES FOR NO2 (PER MINUTE)

ELEVATIONS IN METERS

Zenith Angle	0.0	150.0	360.0	640.0	980.0	1380.0	1840.0	4210.0
0.0	5.340E-01	5.650E-01	5.940E-01	6.210E-01	6.480E-01	6.710E-01	6.910E-01	7.580E-01
10.0	5.290E-01	5.610E-01	5.890E-01	6.170E-01	6.440E-01	6.680E-01	6.880E-01	7.550E-01
20.0	5.160E-01	5.490E-01	5.790E-01	6.070E-01	6.330E-01	6.570E-01	6.780E-01	7.460E-01
30.0	4.930E-01	5.260E-01	5.570E-01	5.860E-01	6.140E-01	6.380E-01	6.500E-01	7.200E-01
40.0	4.570E-01	4.920E-01	5.230E-01	5.530E-01	5.820E-01	6.070E-01	6.290E-01	7.030E-01
50.0	4.040E-01	4.390E-01	4.720E-01	5.030E-01	5.310E-01	5.560E-01	5.790E-01	6.570E-01
60.0	3.250E-01	3.600E-01	3.920E-01	4.230E-01	4.510E-01	4.760E-01	4.990E-01	5.810E-01
70.0	2.140E-01	2.430E-01	2.720E-01	3.000E-01	3.260E-01	3.480E-01	3.690E-01	4.510E-01
78.0	1.050E-01	1.230E-01	1.410E-01	1.600E-01	1.790E-01	1.960E-01	2.110E-01	2.800E-01
86.0	2.300E-02	2.500E-02	2.800E-02	3.100E-02	3.400E-02	3.700E-02	4.000E-02	5.000E-02

TABLE D-5

ATMOSPHERICPHOTODISSOCIATION RATES FOR HCHO* (PER MINUTE)

ELEVATIONS IN METERS

Zenith Angle	0.0	150.0	360.0	640.0	980.0	1380.0	1840.0	4210.0
0.0	6.140E-03	6.620E-03	7.090E-03	7.560E-03	8.020E-03	8.460E-03	8.860E-03	1.020E-02
10.0	6.060E-03	6.540E-03	7.000E-03	7.470E-03	7.940E-03	8.380E-03	8.780E-03	1.010E-02
20.0	5.810E-03	6.290E-03	6.750E-03	7.210E-03	7.670E-03	8.100E-03	8.500E-03	9.850E-03
30.0	5.380E-03	5.840E-03	6.290E-03	6.750E-03	7.190E-03	7.620E-03	8.010E-03	9.370E-03
40.0	4.740E-03	5.190E-03	5.620E-03	6.050E-03	6.480E-03	6.890E-03	7.260E-03	8.610E-03
50.0	3.880E-03	4.280E-03	4.690E-03	5.080E-03	5.470E-03	5.850E-03	6.200E-03	7.500E-03
60.0	2.800E-03	3.130E-03	3.460E-03	3.790E-03	4.120E-03	4.440E-03	4.740E-03	5.940E-03
70.0	1.560E-03	1.780E-03	1.990E-03	2.220E-03	2.440E-03	2.660E-03	2.880E-03	3.810E-03
78.0	6.700E-04	7.600E-04	8.600E-04	9.600E-04	1.070E-03	1.170E-03	1.280E-03	1.810E-03
86.0	1.300E-04	1.500E-04	1.600E-04	1.800E-04	1.900E-04	2.100E-04	2.200E-04	2.900E-04

* Radical and molecular product formation paths combined into on reaction:

$$HCHO + h\nu \overset{k_{47}}{=} HO2 + CO$$

TABLE D-6

ATMOSPHERIC PHOTODISSOCIATION RATES FOR HONO (PER MINUTE)

ELEVATIONS IN METERS

Zenith Angle	0.0	150.0	360.0	640.0	980.0	1380.0	1840.0	4210.0
0.0	9.580E-02	1.020E-01	1.080E-01	1.130E-01	1.180E-01	1.230E-01	1.270E-01	1.400E-01
10.0	9.500E-02	1.010E-01	1.070E-01	1.120E-01	1.170E-01	1.220E-01	1.260E-01	1.400E-01
20.0	9.250E-02	9.870E-02	1.050E-01	1.100E-01	1.150E-01	1.200E-01	1.240E-01	1.380E-01
30.0	8.810E-02	9.450E-02	1.000E-01	1.060E-01	1.120E-01	1.160E-01	1.210E-01	1.350E-01
40.0	8.120E-02	8.770E-02	9.380E-02	9.960E-02	1.050E-01	1.100E-01	1.450E-01	1.290E-01
50.0	7.100E-02	7.770E-02	8.370E-02	8.950E-02	9.510E-02	1.000E-01	1.040E-01	1.200E-01
60.0	5.630E-02	6.260E-02	6.850E-02	7.410E-02	7.950E-02	8.430E-02	8.860E-02	1.050E-02
70.0	3.610E-02	4.110E-02	4.600E-02	5.080E-02	5.540E-02	5.960E-02	6.340E-02	7.940E-02
78.0	1.740E-02	2.000E-02	2.290E-02	2.600E-02	2.890E-02	3.170E-02	3.440E-02	4.680E-02
86.0	3.900E-03	4.300E-03	4.800E-03	5.200E-03	5.700E-03	6.100E-03	6.560E-02	8.780E-02

New assumptions concerning the typical reactivity of individual hydrocarbon groups in the atmosphere are employed. The rate constants for reactions involving alkanes (PA) assume that the atmospheric alkane mixture is as reactive as pure n-butane. The atmospheric mixture of aromatic hydrocarbons (AR) is assumed to be as reactive as a 50 percent o-xylene/50 percent m-xylene mixture. For the reactions involving higher olefins (alkenes excluding ethylene) the reactivity of the mixture is assumed to decrease with elevation. It is believed that the most reactive olefins (double bonded C_4's, C_5's, and C_6's), which are emitted near the surface, react so quickly that few are in fact available for diffusion to the higher elevations. In other words, their reaction time scale is significantly shorter than their diffusion time scale. The elevation dependent rate constants for reactions involving higher olefins (ALKE) are shown in Table D-7. In the lowest two cells of the air parcel (typically 0 to 150 meters), the higher olefin mixture is assumed to be as reactive as a 50 percent propene/50 percent cis-2-butene mixture. In the third and fourth vertical cells, the mixture is assumed to react as if it were composed of 75 percent propene/25 percent cis-2-butene. Above the fourth cell, the rate constants for pure propene are assumed.

In this mechanism, the temperature dependence of the dinitrogen pentoxide (N2O5) dissociation rate constant is included. The rate constant for this reaction step is determined from the expression

$$R_{18} = 3.42 \times 10^{16} \exp(-10{,}600/T) \ (\text{min}^{-1})$$

where T is temperature in degrees Kelvin (Baulch et al. 1973).

In order to utilize the two olefin class mechanism with the RAPS emission inventory, the RAPS single olefin emission rates are uniformly partitioned into ethylene and higher olefin emission rates. The single olefin emission rates are partitioned assuming a 50/50 split on a molar basis. This assumes that olefins emissions are approximately 35 percent ethylene and 65 percent higher olefins by weight which is typical of vehicle olefin emissions.

Two additional features have been implemented in the chemical diffusion module for computing certain species concentrations. The concentrations of oxygen atoms (O) and hydroxyl radicals (OH) are computed using algebraic expressions based on the steady-state or chemical equilibrium assumption.

TABLE D-7

HIGHER OLEFIN REACTION RATE CONSTANTS

Reaction	Rate Constants Cells 1&2	Cells 3&4	Cell 5
$O3 + ALKE \xrightarrow{k_{25}} .5HCHO + .5RCHO + .4HD + 0.4RD + .1RO2$	4.58E-2	2.37E-2	1.60E-3
$OH + ALKE \xrightarrow{k_{18}} AO2$	5.80E+4	4.75E+4	3.70E+4
$O + ALKE \xrightarrow{k_{23}} .3EPOX + .3RCHO + .4HO2 + .4RO2$	1.46E+4	9.90E+3	5.20E+3

Both species react so quickly that the assumption is appropriate. By computing their concentrations via explicit algebraic equations instead of numerical integration, the numerical stiffness of the system of differential equations is reduced. This results in computer time savings. Second, the concentrations of six "product only" species (not reactants) are computed in an approximate manner. These species include nitric acid, organic nitrates, aromatic nitrates, aromatic aldehydes, hydrogen peroxide, and ozonides. These predicted concentrations are probably accurate within an order of magnitude and may be of interest to researchers.

Lastly, the manner in which the phenomenon of surface deposition is simulated in the model has been reformulated. The deposition term in the model's equation has been modified so that it is calculated assuming only the concentrations in a ten meter thick surface layer are effected by the phenomenon. Formerly, the effects of surface deposition may have been exaggerated by assuming the concentrations in the lowest 30 to 100 meters of the air parcel were subject to deposition.

2. KEMOD Input Data Modifications

Numerous changes in the KEMOD input deck are required for simulations with the improved chemical mechanism. A listing of the revised portion of the input deck for the sample problem is shown in Figure D-1. This figure includes all the data cards positioned after the emission rate cards, even though only some of the cards differ from that described in Section 5. The general sequence in which the cards are read is, of course, identical to that described in Section 5.

The changes in the data deck are summarized as follows:

1) The number of reaction equals 64.

2) The number of species equals 39.

3) The number of species not formally integrated is 9.

4) There is a 39 card species list.

5) There are some different emission species indices.

6) NO2 is specified as a species with surface deposition.

7) There are different indices for SO2 and SO4 on the deposition cards.

8) There are 39 cards for initial concentration (see Table D-8 for delineation of user specified and internally computed species initial concentrations).

9) There are 64 chemical rate constant cards.

10) There are 64 chemical reaction cards.

11) There are 4 time-varying photolyic reactions and 4 cards indicating their reaction numbers.

TABLE D-8

USER SPECIFIED AND INTERNALLY COMPUTED INITIAL CONCENTRATIONS

User Specified Species	Internally Computed Species
NO	NO2
03	HONO
CO	PAN
HCHO	ARO
RCHO	AC
ALKE	ARIN
C2H4	ABO
PA	AO
AR	AO2
HO2	NO3
SO2	HNO4
SO4	N2O5
OH	RO2
H2O	RCO3
HNO3*	HD
OZID*	RD
H2O2*	PAO
NTRA*	PAO2
ANO2*	0
ACHO*	

*"Products Only" Species

3. KEMOD FORTRAN Source Code Modifications

In order to accommodate the changes related to use of the improved mechanism, many KEMOD subroutines have been modified. In four subroutines (DRIVE, PEDERV, PSET, and TSTEP) the user must change the array sizes in common block CHEM2 so that it appears as follows:

```
COMMON/CHEM2/  CONIN (40, 5), WTMOLE (40), RATKON (65), RATEFF (65),
1              RATEV (4, 5), QRATE, NVRATE, LOCVRT (4)
```

This is the only change in these four routines. A new subroutine named PHOTOR has been added to replace the tasks formerly performed by subroutines PHOTOD and UPRAT2. The names of subroutines with significant changes are shown below. Complete listings of these subroutines appear on the following pages.

DIFFUN	RATES
ISTATE	STEADY
JACOB	TEMPR
KEMOD2	UNMIXR
PHOTOR	UPFLX1
PRODUK	UPSORC
RATEHI	

IMAGES OF DATA-CARDS

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 321  POINT SOURCES 46         735.35    .0000       .3910-008   .0000       .9444-009
 322                                     .0000       .0000       .0000
 323                46      .5037+000   .9812-001   .1570+000   .2412+000   .0000
 324  POINT SOURCES 47         735.84    .9919-008   .7141-011   .7141-011   .2380-011
 325                                     .7141-011   .1204-008   .1860-010
 326                47      .3779-001   .5420+000   .1515+000   .2687+000   .0000
 327  POINT SOURCES 48         735.84    .7541-008   .0000       .0000       .0000
 328                                     .0000       .0000       .0000
 329                48      .5250+000   .1023+000   .1635+000   .2092+000   .0000
 330  POINT SOURCES 49         736.08    .8620-008   .0000       .0000       .0000

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 331                                     .0000       .0000       .0000
 332                49      .5249+000   .1023+000   .1635+000   .2092+000   .0000
 333  POINT SOURCES 50         736.22    .7189-008   .0000       .0000       .0000
 334                                     .0000       .0000       .3288-011
 335                50      .5265+000   .1026+000   .1641+000   .2069+000   .0000
 336  POINT SOURCES 51         742.74    .1731-008   .1503-011   .1503-011   .5010-012
 337                                     .1503-011   .2086-009   .8867-010
 338                51      .4026-001   .5771+000   .1610+000   .2216+000   .0000
 339  POINT SOURCES 52         742.74    .1731-008   .1503-011   .1503-011   .5010-012
 340                                     .1503-011   .2086-009   .8867-010

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 341                52      .4026-001   .5771+000   .1610+000   .2216+000   .0000
 342  POINT SOURCES 53         742.87    .6204-008   .3006-011   .3006-011   .1002-011
 343                                     .3006-011   .3216-009   .9149-009
 344                53      .4120-001   .5898+000   .1646+000   .2043+000   .0000
 345  POINT SOURCES 54         752.32    .6287-009   .4527-012   .4527-012   .1509-012
 346                                     .4527-012   .7635-010   .1179-011
 347                54      .5265+000   .1026+000   .1641+000   .2069+000   .0000
 348  POINT SOURCES 55         -10.00    .0000       .0000       .0000       .0000
 349                                     .0000       .0000       .0000
 350  INITIAL INTEGRATION TIME STEP SIZE                1.0E-04     (MINUTES)

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 351  MAXIMUM INTEGRATION TIME STEP SIZE      6.0                 (MINUTES)
 352  INTEGRATION ERROR CONTROL CRITERIA          1.0E-03
 353  UPDATE INTERVAL                         15.0                (MINUTES)
 354  PRINT INTERVAL                          30.0                (MINUTES)
 355  DO WE OUTPUT INTEGRATION PARAMETERS<    YES
 356  DO WE GENERATE PHOTODISSOCIATION RATES< YES
 357  SHOULD THEY VARY WITH ELEVATION<        YES
 358  DO WE HAVE SKY CLEARNESS RATIOS INPUT<  YES
 359  DO WE HAVE TEMPERATURES INPUT<          YES
 360  SHOULD WE PUNCH GROUND CONCENTRATIONS<  NO
```

Figure D-1. KEMOD Input Deck for Improved Chemical Mechanism

IMAGES OF DATA-CARDS

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 361  MAXIMUM NUMBER OF INTEGRATION STEPS            1000
 362  NUMBER OF CHEMICAL REACTIONS                     64
 363  NUMBER OF SPECIES                                39
 364  NUMBER OF SPECIES NOT INTEGRATED                  9
 365  NUMBER OF VERTICAL MESH POINTS                    5
 366  ELEVATION OF FIRST  VERTICAL MESH POINT 0.0           (METERS ABOVE SURFACE)
 367  ELEVATION OF SECOND VERTICAL MESH POINT 120.0
 368  ELEVATION OF THIRD  VERTICAL MESH POINT 300.0
 369  ELEVATION OF FOURTH VERTICAL MESH POINT 600.0
 370  ELEVATION OF FIFTH  VERTICAL MESH POINT 1200.0

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 371  SPECIES NAME, MOLE WT, AND BOUND COND    NO        30.
 372                                 2         NO2       46.                 1
 373                                 3         O3        48.                 1
 374                                 4         HONO      47.
 375                                 5         CO        28.
 376                                 6         HCHO      30.
 377                                 7         RCHO      58.
 378                                 8         ALKE      49.
 379                                 9         C2H4      28.
 380                                 10        PA        58.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 381                                 11        PAN       121.
 382                                 12        AR        92.
 383                                 13        ARO       123.
 384                                 14        AC        100.
 385                                 15        ARIN      100.
 386                                 16        A3O       100.
 387                                 17        AO        58.
 388                                 18        AO2       74.
 389                                 19        NO3       62.
 390                                 20        HNO4      79.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 391                                 21        N2O5      108.
 392                                 22        HO2       33.
 393                                 23        RO2       89.
 394                                 24        RCO3      103.
 395                                 25        HO        74.
 396                                 26        RO        73.
 397                                 27        PAO       73.
 398                                 28        PAO2      89.
 399                                 29        SO2       64.
 400                                 30        SO4       96.
```

Figure D-1 (Continued)

IMAGES OF DATA-CARDS

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 401                                 31        OH        17.
 402                                 32        O         16.
 403                                 33        H2O       18.
 404                                 34        HNO3      63.
 405                                 35        OZID      50.
 406                                 36        H2O2      34.
 407                                 37        NTRA      80.
 408                                 38        ANO2      100.
 409                                 39        ACHO      70.
 410  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 1          1

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 411  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 2         10
 412  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 3          9
 413  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 4         12
 414  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 5          5
 415  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 6          5
 416  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 7         29
 417  SPECIE INDEX FOR POINT SOURCE FLUX NO. 1          1
 418  SPECIE INDEX FOR POINT SOURCE FLUX NO. 2         10
 419  SPECIE INDEX FOR POINT SOURCE FLUX NO. 3          9
 420  SPECIE INDEX FOR POINT SOURCE FLUX NO. 4         12

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 421  SPECIE INDEX FOR POINT SOURCE FLUX NO. 5          5
 422  SPECIE INDEX FOR POINT SOURCE FLUX NO. 6          5
 423  SPECIE INDEX FOR POINT SOURCE FLUX NO. 7         29
 424  DEPOSITION SPECIE INDEX,VELOCITY,EXP.             2       0.60       1.0
 425  DEPOSITION SPECIE INDEX,VELOCITY,EXP.             3       0.30       1.0
 426  DEPOSITION SPECIE INDEX,VELOCITY,EXP.            29       0.60       1.0
 427  DEPOSITION SPECIE INDEX,VELOCITY,EXP.            30       0.18       1.0
 428  NO      IC 1            3.0E-03   2.5E-03   1.5E-03   1.5E-03   1.5E-03
 429  NO2      2 COMPUTED
 430  O3      IC 3          .029      .043      .058      .058      .058

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 431  HONO     4 COMPUTED
 432  CO      IC 5          .133      .110      .066      .065      .066
 433  HCHO    IC 6             4.0E-04   3.4E-04   2.0E-04   2.0E-04   2.0E-04
 434  RCHO    IC 7             2.0E-04   1.7E-04   1.0E-04   1.0E-04   1.0E-04
 435  ALKE    IC 8             2.0E-04   1.7E-04   1.0E-04   1.0E-04   1.0E-04
 436  C2H4    IC 9             4.0E-04   3.4E-04   2.0E-04   2.0E-04   2.0E-04
 437  PA      IC 10            3.0E-03  2.54E-03   1.5E-03   1.5E-03   1.5E-03
 438  PAN      11 COMPUTED
 439  AR      IC 12            6.0E-04  5.10E-04   3.0E-04   3.0E-04   3.0E-04
 440  ARO     13 COMPUTED
```

Figure D-1 (Continued)

IMAGES OF DATA-CARDS

```
        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 441    AC       14COMPUTED
 442    ARIN     15COMPUTED
 443    ABO      16COMPUTED
 444    AO       17COMPUTED
 445    AO2      18COMPUTED
 446    NO3      19COMPUTED
 447    HNO4     20COMPUTED
 448    N2O5     21COMPUTED
 449    HO2     IC 22            1.0E-06   1.0E-06   1.0E-06   1.0E-06   1.0E-06
 450    RO2      23COMPUTED

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 451    RCO3     24COMPUTED
 452    HO       25COMPUTED
 453    RO       26COMPUTED
 454    PAO      27COMPUTED
 455    PAO2     28COMPUTED
 456    SO2    IC  29            1.0E-02   .85E-03   .50E-03   .50E-03   .50E-03
 457    SO4     IC 30            1.0E-03   .85E-03   .50E-03   .50E-03   .50E-03
 458    OH      IC 31            1.00E-8   1.00E-8   1.00E-8   1.00E-8   1.00E-8
 459    O        32COMPUTED
 460    H2O     IC 33            1.7E 04   1.7E 04   1.7E 04   1.7E 04   1.7E 04

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 461    HNO3   IC  34            1.E-09    1.E-09    1.E-09    1.E-09    1.E-09
 462    OZID   IC  35            1.E-09    1.E-09    1.E-09    1.E-09    1.E-09
 463    H2O2   IC  36            1.E-09    1.E-09    1.E-09    1.E-09    1.E-09
 464    NTRA   IC  37            1.E-09    1.E-09    1.E-09    1.E-09    1.E-09
 465    ANO2   IC  38            1.E-09    1.E-09    1.E-09    1.E-09    1.E-09
 466    ACHO   IC  39            1.E-09    1.E-09    1.E-09    1.E-09    1.E-09
 467              REACTION NO.        1                     COMPUTED FROM SOLAR
 468              REACTION NO.        2            4.12E+05
 469              REACTION NO.        3            2.50E+01
 470              REACTION NO.        4            2.20E-09

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 471              REACTION NO.        5            1.40E-03
 472              REACTION NO.        6                     COMPUTED FROM SOLAR
 473              REACTION NO.        7            1.44E+04
 474              REACTION NO.        8            1.44E+04
 475              REACTION NO.        9            4.40E+02
 476              REACTION NO.        10           1.20E+04
 477              REACTION NO.        11           1.71E+03
 478              REACTION NO.        12           3.61E+03
 479              REACTION NO.        13           5.00E-02
 480              REACTION NO.        14           2.70E+04
```

Figure D-1 (Continued)

IMAGES OF DATA-CARDS

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 491              REACTION NO.        15          9.30E 02
 492              REACTION NO.        16          1.00E-03
 493              REACTION NO.        17                    COMPUTED FROM TEMPERATURE
 494              REACTION NO.        18                   COMPUTED FROM MESH PT HEIGHT
 495              REACTION NO.        19          1.20E+04
 496              REACTION NO.        20          2.90E+04
 497              REACTION NO.        21          4.10E+05
 498              REACTION NO.        22                    COMPUTED FROM TEMPERATURE
 499              REACTION NO.        23                   COMPUTED FROM MESH PT HEIGHT
 500              REACTION NO.        24          1.10E+03

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 491              REACTION NO.        25                   COMPUTED FROM MESH PT HEIGHT
 492              REACTION NO.        26          2.50E-03
 493              REACTION NO.        27          2.90E+04
 494              REACTION NO.        28          2.90E+04
 495              REACTION NO.        29          1.90E+04
 496              REACTION NO.        30          1.90E+04
 497              REACTION NO.        31          1.00E+01
 498              REACTION NO.        32          1.00E+01
 499              REACTION NO.        33          1.00E+01
 500              REACTION NO.        34          1.00E+01

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 501              REACTION NO.        35           4.40E+3
 502              REACTION NO.        36          2.90E+04
 503              REACTION NO.        37          2.60E+03
 504              REACTION NO.        38          1.40E+05
 505              REACTION NO.        39          2.90E+04
 506              REACTION NO.        40          6.70E+04
 507              REACTION NO.        41          2.30E+03
 508              REACTION NO.        42          2.20E+04
 509              REACTION NO.        43                    COMPUTED FROM SOLAR
 510              REACTION NO.        44          2.90E+04

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 511              REACTION NO.        45          1.70E+04
 512              REACTION NO.        46                    COMPUTED FROM TEMPERATURE
 513              REACTION NO.        47                    COMPUTED FROM SOLAR
 514              REACTION NO.        48          1.60E+04
 515              REACTION NO.        49          4.20E+03
 516              REACTION NO.        50           2.24E+4
 517              REACTION NO.        51           5.60E+3
 518              REACTION NO.        52          4.90E+04
 519              REACTION NO.        53          2.90E+04
 520              REACTION NO.        54          1.90E+04
```

Figure D-1 (Continued)

IMAGES OF DATA-CARDS

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 521              REACTION NO.         55           1.80E+03
 522              REACTION NO.         56           1.00E-01
 523              REACTION NO.         57            6.70E+3
 524              REACTION NO.         58           2.90E+04
 525              REACTION NO.         59           1.76E 03
 526              REACTION NO.         60           3.00E-02
 527              REACTION NO.         61           2.90E 03
 528              REACTION NO.         62           2.90E 03
 529              REACTION NO.         63           8.00E 00
 530              REACTION NO.         64           8.00E 00

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 531    1.NO2   +    HV               =    1.O     +  1.NO
 532    1.O     +    O2    +    M     =    1.O3    +    M
 533    1.O3    +  1.NO               =    1.NO2   +    O2
 534    1.NO    +  1.NO2   +  1.H2O   =    2.HONO
 535    2.HONO                        =    1.NO    +  1.NO2   +  1.H2O
 536    1.HONO  +    HV               =    1.OH    +  1.NO
 537    1.OH    +  1.NO    +    M     =    1.HONO  +    M
 538    1.OH    +  1.NO2   +    M     =    1.HNO3  +    M
 539    1.OH    +  1.CO               =    1.HO2   +  1.CO2
 540    1.HO2   +  1.NO               =    1.NO2   +  1.OH

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 541    1.HO2   +  1.NO2              =    1.HNO4
 542    2.HO2                         =      H2O2  +    O2
 543    1.NO2   +  1.O3               =    1.NO3   +    O2
 544    1.NO3   +  1.NO               =    2.NO2
 545    1.NO3   +  1.NO2              =    1.N2O5
 546    1.N2O5  +  1.H2O              =    2.HNO3
 547    1.N2O5                        =    1.NO3   +  1.NO2
 548    1.OH    +  1.ALKE             =    1.AO2
 549    1.OH    +  1.C2H4             =    1.AO2
 550    1.AO2   +  1.NO               =    1.NO2   +  1.AO

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 551    1.AO    +    O2               =    .5RCHO  + 1.5HCHO  +  1.HO2
 552    1.HNO4                        =    1.HO2   +  1.NO2
 553    1.O     +  1.ALKE             =    .3EPOX  +  .3RCHO  +  .4HO2   +  .4RO2
 554    1.O     +  1.C2H4             =    .3EPOX  +  .3RCHO  +  .4HO2   +  .4RO2   +  .4CO
 555    1.O3    +  1.ALKE             =    .5HCHO  +  .5RCHO  +  .4HO    +  .4RO    +  .1RO2
 556    1.O3    +  1.C2H4             =    1.HCHO  +  .8HO    +  .4HO2   +  .2CO2
 557    1.HO    +  1.NO               =    1.HCHO  +  1.NO2
 558    1.RO    +  1.NO               =    1.RCHO  +  1.NO2
 559    1.HO    +  1.NO2              =    1.HCHO  +  1.NO3
 560    1.RO    +  1.NO2              =    1.RCHO  +  1.NO3
```

Figure D-1 (Continued)

IMAGES OF DATA-CARDS

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 561    1.HO    +  1.HCHO             =     OZID
 562    1.HO    +  1.RCHO             =     OZID
 563    1.RO    +  1.HCHO             =     OZID
 564    1.RO    +  1.RCHO             =     OZID
 565    1.OH    +  1.PA               =   1.H2O   +  1.PAO2
 566    1.PAO2  +  1.NO               =   1.NO2   + .85PAO  + .15RO2
 567    1.PAO2  +  1.NO               =     NTRA
 568    1.PAO                          =  1.RO2   +  .5HCHO +  .5RCHO
 569    1.RO2   +  1.NO               =   1.NO2   +  1.PAO
 570    1.PAO   +    O2               =   1.RCHO  +  1.HO2

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 571    1.PAO   +  1.NO2              =  .85NTRA + .15RCHO + .15HONO
 572    1.OH    +  1.RCHO             =   1.RCO3  +  1.H2O
 573    1.RCHO  +    HV               =   1.RO2   +  1.HO2   +  1.CO
 574    1.RCO3  +  1.NO               =     CO2   +  1.NO2   +  1.RO2
 575    1.RCO3  +  1.NO2  +           =   1.PAN
 576    1.PAN                         =   1.RCO3  +  1.NO2
 577    1.HCHO  +    HV               =   1.HO2   +  1.CO
 578    1.HCHO  +  1.OH               =   1.H2O   +  1.HO2   +  1.CO
 579    1.RO2   +  1.HO2              =     RO2H  +    O2
 580    1.OH    +  1.AR               =   1.AC    +  1.HO2

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 581    1.OH    +  1.AR               =   1.ARO
 582    1.OH    +  1.AC               =     ARP   +  1.HO2
 583    1.ARO   +  1.NO               =   1.NO2   +  1.ARIN
 584    1.ARO   +  1.NO2              =     ANO2
 585    1.ARO   +  1.HO2              =   1.AO    +  1.H2O2
 586    1.ARIN                        =   1.RO2   +  1.OH
 587    1.AR    +  1.OH               =   1.H2O   +  1.ABO
 588    1.ABO   +  1.NO               =   1.NO2   +  1.HO2   +    ACHO
 589    1.SO2   +  1.OH               =   1.SO4
 590    1.SO2   +  1.HO2              =   1.SO4   +  1.OH

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.....
 591    1.SO2   +  1.HO               =   1.SO4   +  1.HCHO
 592    1.SO2   +  1.RO               =   1.SO4   +  1.RCHO
 593    1.SO2   +  1.RO2              =   1.SO4   +  1.PAO
 594    1.SO2   +  1.AO2              =   1.SO4   +  1.AO
 595  LATITUDE                                      38.60
 596  LONGITUDE ,                                   90.20
 597  TIME ZONE                                      6.00
 598  DATE                                         760629
 599  NUMBER OF PHOTOLYTIC REACTIONS                    4
 600  REACTION NUMBER OF 1ST PHOTOLYTIC RATE            1          NO2 + HV
```

IMAGES OF DATA-CARDS

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 601  REACTION NUMBER OF 2ND PHOTOLYTIC RATE          5          HONO + HV
 602  REACTION NUMBER OF 3RD PHOTOLYTIC RATE         43          RCHO + HV
 603  REACTION NUMBER OF 4TH PHOTOLYTIC RATE         47          HCHO + HV
 604   CLEARNESS RATIOS                          360.000      .547
 605   CLEARNESS RATIOS                          420.000      .702
 606   CLEARNESS RATIOS                          480.000      .757
 607   CLEARNESS RATIOS                          540.000      .724
 608   CLEARNESS RATIOS                          600.000      .695
 609   CLEARNESS RATIOS                          660.000      .545
 610   CLEARNESS RATIOS                          720.000      .557

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 611   CLEARNESS RATIOS                          780.000     1.000
 612   CLEARNESS RATIOS                          840.000      .727
 613   CLEARNESS RATIOS                          900.000      .975
 614   CLEARNESS RATIOS                          960.000     1.000
 615   CLEARNESS RATIOS                         1020.000      .772
 616   CLEARNESS RATIOS                          -10.000
 617   TEMPERATURE (DEG C)                       360.000    22.051
 618   TEMPERATURE (DEG C)                       420.000    23.462
 619   TEMPERATURE (DEG C)                       480.000    25.332
 620   TEMPERATURE (DEG C)                       540.000    26.715

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 621   TEMPERATURE (DEG C)                       600.000    28.200
 622   TEMPERATURE (DEG C)                       660.000    28.693
 623   TEMPERATURE (DEG C)                       720.000    28.319
 624   TEMPERATURE (DEG C)                       780.000    28.849
 625   TEMPERATURE (DEG C)                       840.000    28.763
 626   TEMPERATURE (DEG C)                       900.000    28.947
 627   TEMPERATURE (DEG C)                       960.000    29.459
 628   TEMPERATURE (DEG C)                      1020.000    29.238
 629   TEMPERATURE (DEG C)                       -10.000
 630   DIFFUSIVITIES 1          360.0              52.3      52.3      31.4

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 631                             20.1       8.8       8.8
 632   DIFFUSIVITIES 2          420.0              74.5      74.5      25.1
 633                             15.7       6.1       6.1
 634   DIFFUSIVITIES 3          480.0             240.2     240.2      45.1
 635                             27.5      33.0      33.0
 636   DIFFUSIVITIES 4          540.0            1331.0    1331.0     973.2
 637                             27.8      36.4      36.4
 638   DIFFUSIVITIES 5          600.0            1621.5    1621.5    3884.1
 639                            1823.0       9.0       9.0
 640   DIFFUSIVITIES 6          660.0             146.5     146.6     500.4
```

IMAGES OF DATA-CARDS

```
      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 641                              290.9        8.2        8.2
 642   DIFFUSIVITIES 7            720.0                  2827.8     2827.8     7543.1
 643                             4910.7      106.3        106.3
 644   DIFFUSIVITIES 8            790.0                  2019.0     2019.0     6310.0
 645                             5702.1      438.4        438.4
 646   DIFFUSIVITIES 9            840.0                  1729.5     1729.5     5118.5
 647                             4384.2      290.0        290.0
 648   DIFFUSIVITIES10            900.0                  1424.5     1424.5     4121.3
 649                             3369.0      191.9        191.9
 650   DIFFUSIVITIES11            960.0                  1439.5     1439.5     2183.3

      1         11        21         31         41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 651                             1091.7       39.1        39.1
 652   DIFFUSIVITIES11           1020.0                  1439.5     1439.5     2183.3
 653                             1091.7       39.1        39.1
 654   DIFFUSIVITIES11            -10.0                     .0         .0         .0
 655                                .0         .0         .0
 656   END
```

```
      SUBROUTINE DIFFUN (N,T,Y,YDOT)                                        NKM    1
C                                                                           NKM    2
C     ------       DIFFUN DEFINES THE DIFFERENTIAL EQUATIONS  ----------NKM    3
C                                                                           NKM    4
C                         YDOT = F(Y,T)                                     NKM    5
C                                                                           NKM    6
      INTEGER BCFLAG                                                        NKM    7
      COMMON/CHEM1/ NOSTAT,            NOSTM1,          NOREAC,           NKM    8
     1              NOSPEC,            NSTDY,           NK                NKM    9
      COMMON/CHEM2/ CONIN(40,5),       WTMOLE(40),      RATKON(65),       NKM   10
     1              RATEFF(65),        RATEV(4,5),      QRATE,            NKM   11
     2              NVRATE,            LOCVRT(4)                          NKM   12
      COMMON/CHEM3/ ZEE(5),            DELZ(4),         HTCELL(6),        NKM   13
     1              TOELZ(2) ,         DFINIT(6),       SCALOW(4),        NKM   14
     2              DCOF(5),           FLXWAL(40),      FLXDGE(40),       NKM   15
     3              BCFLAG(40),        DPRATE(10),      DEPOWR(10),       NKM   16
     4              LOCDPF(10),        NDPFLX ,         SCALUP(4)         NKM   17
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT     NKM   18
      COMMON /CHEM5/  REACT(28,65),SPEC(40),LOCFLX(10),                   NKM   19
     1                NASFLX,FLXW1(30),FLXW2(30)                          NKM   20
      COMMON /PS1/    TPASS(200), PS(7,5,75), FRACT(3),                   NKM   21
     1               NPTSR,NPSFLX,LOCPSF(7)                               NKM   22
      DIMENSION  Y(N),      YDOT(N),      SINK(30)                        NKM   23
      DATA SINK /30*0.0000/                                               NKM   24
C                                                                         NKM   25
      CALL RATES(Y,YDOT)                                                  NKM   26
C                                                                         NKM   27
      IF(NDPFLX.LT.1)  GO TO 50                                           NKM   28
C     ----    UPDATE FLUXES OF DEPOSITING SPECIES                         NKM   29
      DO 40 K=1,NDPFLX                                                    NKM   30
      ID = LOCDPF(K)                                                      NKM   31
      SINK(ID) = 0.00                                                     NKM   32
      IF(Y(ID).LT.1.E-16)  GO TO 40                                       NKM   33
      SINK(ID) = - DPRATE(K)*(Y(ID)**DEPOWR(K))                           NKM   34
   40 CONTINUE                                                            NKM   35
   50 CONTINUE                                                            NKM   36
C                                                                         NKM   37
C       INTERPOLATE EMISSION FLUXES TO THE CURRENT TIME                   NKM   38
C                                                                         NKM   39
      FDELT = (T - TLAST)/UPDINT                                          NKM   40
      DO 60 I = 1,NASFLX                                                  NKM   41
      K = LOCFLX(I)                                                       NKM   42
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                    NKM   43
60    CONTINUE                                                            NKM   44
      K = 7                                                               NKM   45
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                    NKM   46
      K = 2                                                               NKM   47
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                    NKM   48
      K = 9                                                               NKM   49
      FLXWAL(K)=FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                      NKM   50
      DO 75 J =1,NOSTAT                                                   NKM   51
      DO 70 I =1,NPSFLX                                                   NKM   52
      K =LOCPSF(I)                                                        NKM   53
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT               NKM   54
70    CONTINUE                                                            NKM   55
      K = 2                                                               NKM   56
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT               NKM   57
      K = 7                                                               NKM   58
```

```
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT                NKM   59
      K = 9                                                                NKM   60
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT                NKM   61
75    CONTINUE                                                             NKM   62
C                                                                          NKM   63
      NM1 = NK*NOSTM1                                                      NKM   64
      NM2 = NK*(NOSTAT-2)                                                  NKM   65
C     ------- EQUATIONS FOR SURFACE AND TOP EDGE MESH POINTS               NKM   66
      DO 100  I=1,NK                                                       NKM   67
      IL=NM1+I                                                             NKM   68
      YDOT(I) = DCOF(1)*(Y(I+NK)-Y(I)) + YDOT(I)                           NKM   69
     1                 +  TDELZ(1)*(FLXMAL(I)+SINK(I)) + PSRATE(I,1)      NKM   70
      IF(BCFLAG(I) .EQ. 2)  YDOT(I) = 0.0                                  NKM   71
      YDOT(IL) = DCOF(NOSTAT)*(Y(I+NM2)-Y(IL)) + YDOT(IL)                  NKM   72
     1                       +  PSRATE(I,NOSTAT)                           NKM   73
  100 CONTINUE                                                             NKM   74
C     ------- EQUATIONS FOR INTERMEDIATE MESH POINTS                       NKM   75
      DO 200 J=2,NOSTM1                                                    NKM   76
      NM1= (J-2)*NK                                                        NKM   77
      NN = (J-1)*NK                                                        NKM   78
      NP1 = J*NK                                                           NKM   79
      JM1 = J- 1                                                           NKM   80
      DO 150 I = 1,NK                                                      NKM   81
      LN = I + NN                                                          NKM   82
      YDOT(LN) = SCALOW(JM1)*Y(I+NM1) - DCOF(J)*Y(LN) +                    NKM   83
     1           SCALUP(J)*Y(I+NP1) + YDOT(LN) + PSRATE(I,J)               NKM   84
  150 CONTINUE                                                             NKM   85
  200 CONTINUE                                                             NKM   86
      RETURN                                                               NKM   87
      END                                                                  NKM   88
```

```
      SUBROUTINE ISTATE                                                    NKM   89
C                                                                          NKM   90
C       COMPUTES THE INITIAL CONCENTRATIONS OF THE SPECIES LISTED BELOW    NKM   91
C       USING THE STEADY STATE ASSUMPTION.                                 NKM   92
C       THIS ROUTINE IS SPECIFIC TO THE ERT PHOTOCHEMICAL MECHANISM        NKM   93
C         39 SPECIES   X  64 REACTIONS    1.31.79                        NKM   94
C                                                                          NKM   95
C            NO2      HONO     RO       HO       NO3      N2O5           NKM   96
C            ARO      AC       ARIN     ABO      AO2      AO             NKM   97
C            PAO2     HNO4     PAN      RCO3     RO2      PAO            NKM   98
C            O                                                           NKM   99
C                                                                          NKM  100
      COMMON/CHEM1/ NOSTAT,            NOSTM1,            NOREAC,        NKM  101
     1              NOSPEC,            NSTOY,             NK             NKM  102
      COMMON/CHEM2/ CONIN(40,5),       WTMOLE(40),        RATKON(65),    NKM  103
     1              RATEFF(65),        RATEV(4,5),        QRATE,         NKM  104
     2              NVRATE,            LOCVRT(4)                         NKM  105
      DIMENSION     C(40),    R(65),   A(2,2),   B(2)                     NKM  106
      EQUIVALENCE (R,RATEFF)                                               NKM  107
      DATA YES /3HYES /                                                    NKM  108
C                                                                          NKM  109
      DO 200 K = 1,NOSTAT                                                  NKM  110
      IF(QRATE.EQ.YES) CALL RATEHI(K)                                      NKM  111
      CALL UNMIXR(K)                                                       NKM  112
      CALL XMIT(NOSPEC,CONIN(1,K),C)                                       NKM  113
```

```
C                                                                       NKM  114
C                    - - - NO2 - - -                                    NKM  115
      C( 2) = C( 1)*C( 3)*R( 3)/R( 1)                                   NKM  116
C                    - - -  O  - - -                                    NKM  117
      C(32) = R( 1)*C( 2)/(R( 2) + R(23)*C( 8) + R(24)*C( 9))           NKM  118
C                    - - - HONO - - -                                   NKM  119
      AA = 2.*R( 5)                                                     NKM  120
      BB = R( 6)                                                        NKM  121
      CC = -2.*R( 4)*C( 1)*C( 2)*C(33) - R( 7)*C(31)*C( 1)              NKM  122
      C( 4) = (-BB + (BB*BB - 4.*AA*CC)**.5)/2./AA                      NKM  123
C                    - - -  RO  - - -                                   NKM  124
      C(26) = (.40*R(25)*C( 3)*C( 8))/(R(28)*C( 1) + R(30)*C( 2)        NKM  125
     1          + R(33)*C( 6) + R(34)*C( 7))                           NKM  126
C                    - - -  HO  - - -                                   NKM  127
      C(25) = (.40*R(25)*C( 3)*C( 8) + .80*R(26)*C( 3)*C( 9))/          NKM  128
     1        (R(27)*C( 1) + R(29)*C( 2) + R(31)*C( 6) + R(32)*C( 7))   NKM  129
C                    - - - NO3 AND N2O5 - - -                           NKM  130
      A(1,1) = R(14)*C( 1) + R(15)*C( 2)                                NKM  131
      A(1,2) = -R(17)                                                   NKM  132
      A(2,1) = R(15)*C( 2)                                              NKM  133
      A(2,2) = -(R(16)*C(33) + R(17))                                   NKM  134
      B( 1) = R(13)*C( 2)*C( 3) + R(29)*C( 2)*C(25) + R(30)*C( 2)*C(26) NKM  135
      B( 2) = 0.0                                                       NKM  136
      CALL SOL2B2(A,B)                                                  NKM  137
      C(19) = B( 1)                                                     NKM  138
      C(21) = B( 2)                                                     NKM  139
C                    - - - ARO - - -                                    NKM  140
      C(13) = (R(51)*C(12)*C(31))/(R(53)*C( 1) + R(54)*C( 2)            NKM  141
     1          + R(55)*C(22))                                          NKM  142
C                    - - - AC  - - -                                    NKM  143
      C(14) = (R(50)*C(12)*C(31) + R(55)*C(13)*C(22))/(R(52)*C(31))     NKM  144
C                    - - - ARIN - - -                                   NKM  145
      C(15) = R(53)*C( 1)*C(13)/R(56)                                   NKM  146
C                    - - - ABO  - - -                                   NKM  147
      C(16) = R(57)*C(12)*C(31)/(R(58)*C( 1))                           NKM  148
C                    - - - AO2  - - -                                   NKM  149
      C(18) = (R(18)*C( 8)*C(31) + R(19)*C( 9)*C(31))/(R(20)*C( 1))     NKM  150
C                    - - -  AO  - - -                                   NKM  151
      C(17) = R(20)*C( 1)*C(18)/R(21)                                   NKM  152
C                    - - - PAO2 - - -                                   NKM  153
      C(28) = R(35)*C(10)*C(31)/(R(36)*C( 1) + R(37)*C( 1))             NKM  154
C                    - - - HNO4 - - -                                   NKM  155
      C(20) = R(11)*C( 2)*C(22)/R(22)                                   NKM  156
C                    - - - PAN AND RCO3 - - -                           NKM  157
      A(1,1) = -R(46)                                                   NKM  158
      A(1,2) = R(45)*C( 2)                                              NKM  159
      A(2,1) = +R(46)                                                   NKM  160
      A(2,2) = -R(44)*C( 1) - R(45)*C( 2)                               NKM  161
      B( 1) = 0.0                                                       NKM  162
      B( 2) = -R(42)*C( 7)*C(31)                                        NKM  163
      CALL SOL2B2(A,B)                                                  NKM  164
      C(11) = B(1)                                                      NKM  165
      C(24) = B(2)                                                      NKM  166
C                    - - - RO2 AND PAO - - -                            NKM  167
      A(1,1) = R(39)*C( 1) + R(49)*C(22)                                NKM  168
      A(1,2) = -R(38)                                                   NKM  169
      A(2,1) = -R(39)*C( 1)                                             NKM  170
      A(2,2) = R(38) + R(40) + R(41)*C( 2)                              NKM  171
```

```
      B(1) = .40*R(23)*C( 8)*C(32) + .10*R(25)*C( 3)*C( 8) + .15*R( 36)*NKM  172
     1       C( 1)*C(28) + R(43)*C( 7) + R(44)*C( 1)*C(24) + R(56)*C(15)NKM  173
     2  + 0.4*R(24)*C(9)*C(32)                                            NKM  174
      B(2) = .85*R(36)*C( 1)*C(28)                                       NKM  175
      CALL SOL292(A,B)                                                   NKM  176
      C(23) = B(1)                                                       NKM  177
      C(27) = B(2)                                                       NKM  178
      CALL XMIT(NOSPEC,C,CONIN(1,K))                                     NKM  179
200   CONTINUE                                                           NKM  180
      RETURN                                                             NKM  181
      END                                                                NKM  182



      SUBROUTINE JACOB(A,C,R,N)                                          NKM  183
C                                                                        NKM  184
C     JACOB COMPUTES THE JACOBIAN OF THE CHEMICAL RATE EQUATIONS.        NKM  185
C                                                                        NKM  186
      DIMENSION    A(N,N),       C(1),       R(1)                        NKM  187
C                                                                        NKM  188
C     ZERO THE A ARRAY                                                   NKM  189
      NN2 = N*N                                                          NKM  190
      CALL XMIT(-NN2,0.0,A)                                              NKM  191
C                                                                        NKM  192
C   JACOBIAN OF THE ERT PHOTOCHEMICAL MECHANISM  (39X64)  1.31.79        NKM  193
C                                                                        NKM  194
      A(  1,  1) = - R(  3)*C(  3) - R(  4)*C(  2)*C( 33) - R(  7)       NKM  195
     *    *C( 31) - R( 10)*C( 22) - R( 14)*C( 19) - R( 20)*C( 18) -      NKM  196
     *    R( 27)*C( 25) - R( 28)*C( 26) - R( 36)*C( 28) - R( 37)*C( 28) NKM  197
     *     - R( 39)*C( 23) - R( 44)*C( 24) - R( 53)*C( 13) - R( 58)      NKM  198
     *    *C( 16)                                                        NKM  199
      A(  1,  2) = + R(  1) - R(  4)*C(  1)*C( 33)                       NKM  200
      A(  1,  3) = - R(  3)*C(  1)                                       NKM  201
      A(  1,  4) = +    2.00*R(  5)*C(  4) + R(  6)                      NKM  202
      A(  1, 13) = - R( 53)*C(  1)                                       NKM  203
      A(  1, 16) = - R( 58)*C(  1)                                       NKM  204
      A(  1, 18) = - R( 20)*C(  1)                                       NKM  205
      A(  1, 19) = - R( 14)*C(  1)                                       NKM  206
      A(  1, 22) = - R( 10)*C(  1)                                       NKM  207
      A(  1, 23) = - R( 39)*C(  1)                                       NKM  208
      A(  1, 24) = - R( 44)*C(  1)                                       NKM  209
      A(  1, 25) = - R( 27)*C(  1)                                       NKM  210
      A(  1, 26) = - R( 28)*C(  1)                                       NKM  211
      A(  1, 28) = - R( 36)*C(  1) - R( 37)*C(  1)                       NKM  212
      A(  2,  1) = + R(  3)*C(  3) - R(  4)*C(  2)*C( 33) + R( 10)       NKM  213
     *    *C( 22) +    2.00*R( 14)*C( 19) + R( 20)*C( 18) + R( 27)       NKM  214
     *    *C( 25) + R( 28)*C( 26) + R( 36)*C( 28) + R( 39)*C( 23) +      NKM  215
     *    R( 44)*C( 24) + R( 53)*C( 13) + R( 58)*C( 16)                  NKM  216
      A(  2,  2) = - R(  1) - R(  4)*C(  1)*C( 33) - R(  8)*C( 31) -     NKM  217
     *    R( 11)*C( 22) - R( 13)*C(  3) - R( 15)*C( 19) - R( 29)*C( 25) NKM  218
     *     - R( 30)*C( 26) - R( 41)*C( 27) - R( 45)*C( 24) - R( 54)      NKM  219
     *    *C( 13)                                                        NKM  220
      A(  2,  3) = + R(  3)*C(  1) - R( 13)*C(  2)                       NKM  221
      A(  2,  4) = +    2.00*R(  5)*C(  4)                               NKM  222
      A(  2, 11) = + R( 46)                                              NKM  223
      A(  2, 13) = + R( 53)*C(  1) - R( 54)*C(  2)                       NKM  224
      A(  2, 16) = + R( 58)*C(  1)                                       NKM  225
      A(  2, 18) = + R( 20)*C(  1)                                       NKM  226
```

```
      A(  2, 19) = +    2.00*R( 14)*C(  1) - R( 15)*C(  2)                    NKM  227
      A(  2, 20) = + R( 22)                                                   NKM  228
      A(  2, 21) = + R( 17)                                                   NKM  229
      A(  2, 22) = + R( 10)*C(  1) - R( 11)*C(  2)                            NKM  230
      A(  2, 23) = + R( 39)*C(  1)                                            NKM  231
      A(  2, 24) = + R( 44)*C(  1) - R( 45)*C(  2)                            NKM  232
      A(  2, 25) = + R( 27)*C(  1) - R( 29)*C(  2)                            NKM  233
      A(  2, 26) = + R( 28)*C(  1) - R( 30)*C(  2)                            NKM  234
      A(  2, 27) = - R( 41)*C(  2)                                            NKM  235
      A(  2, 28) = + R( 36)*C(  1)                                            NKM  236
      A(  3,  1) = - R(  3)*C(  3)                                            NKM  237
      A(  3,  2) = - R( 13)*C(  3)                                            NKM  238
      A(  3,  3) = - R(  3)*C(  1) - R( 13)*C(  2) - R( 25)*C(  8) -          NKM  239
     *    R( 26)*C(  9)                                                      NKM  240
      A(  3,  8) = - R( 25)*C(  3)                                            NKM  241
      A(  3,  9) = - R( 26)*C(  3)                                            NKM  242
      A(  4,  1) = +    2.00*R(  4)*C(  2)*C( 33) + R(  7)*C( 31)             NKM  243
      A(  4,  2) = +    2.00*R(  4)*C(  1)*C( 33) +    0.15*R( 41)            NKM  244
     *    *C( 27)                                                            NKM  245
      A(  4,  4) = -    4.00*R(  5)*C(  4) - R(  6)                           NKM  246
      A(  4, 27) = +    0.15*R( 41)*C(  2)                                    NKM  247
      A(  5,  5) = - R(  9)*C( 31)                                            NKM  248
      A(  5,  6) = + R( 47) + R( 48)*C( 31)                                   NKM  249
      A(  5,  7) = + R( 43)                                                   NKM  250
      A(  5,  9) = +    0.40*R( 24)*C( 32)                                    NKM  251
      A(  6,  1) = + R( 27)*C( 25)                                            NKM  252
      A(  6,  2) = + R( 29)*C( 25)                                            NKM  253
      A(  6,  3) = +    0.50*R( 25)*C(  8) + R( 26)*C(  9)                    NKM  254
      A(  6,  6) = - R( 31)*C( 25) - R( 33)*C( 26) - R( 47) - R( 48)          NKM  255
     *    *C( 31)                                                            NKM  256
      A(  6,  8) = +    0.50*R( 25)*C(  3)                                    NKM  257
      A(  6,  9) = + R( 26)*C(  3)                                            NKM  258
      A(  6, 17) = +    1.50*R( 21)                                           NKM  259
      A(  6, 25) = + R( 27)*C(  1) + R( 29)*C(  2) - R( 31)*C(  6) +          NKM  260
     *    R( 61)*C( 29)                                                      NKM  261
      A(  6, 26) = - R( 33)*C(  6)                                            NKM  262
      A(  6, 27) = +    0.50*R( 38)                                           NKM  263
      A(  7,  1) = + R( 28)*C( 26)                                            NKM  264
      A(  7,  2) = + R( 30)*C( 26) +    0.15*R( 41)*C( 27)                    NKM  265
      A(  7,  3) = +    0.50*R( 25)*C(  8)                                    NKM  266
      A(  7,  7) = - R( 32)*C( 25) - R( 34)*C( 26) - R( 42)*C( 31) -          NKM  267
     *    R( 43)                                                             NKM  268
      A(  7,  8) = +    0.30*R( 23)*C( 32) +    0.50*R( 25)*C(  3)            NKM  269
      A(  7,  9) = +    0.30*R( 24)*C( 32)                                    NKM  270
      A(  7, 17) = +    0.50*R( 21)                                           NKM  271
      A(  7, 25) = - R( 32)*C(  7)                                            NKM  272
      A(  7, 26) = + R( 28)*C(  1) + R( 38)*C(  2) - R( 34)*C(  7) +          NKM  273
     *    R( 62)*C( 29)                                                      NKM  274
      A(  7, 27) = +    0.50*R( 38) + R( 40) +    0.15*R( 41)*C(  2)          NKM  275
      A(  8,  3) = - R( 25)*C(  8)                                            NKM  276
      A(  8,  8) = - R( 18)*C( 31) - R( 23)*C( 32) - R( 25)*C(  3)            NKM  277
      A(  9,  3) = - R( 26)*C(  9)                                            NKM  278
      A(  9,  9) = - R( 19)*C( 31) - R( 24)*C( 32) - R( 26)*C(  3)            NKM  279
      A( 10, 10) = - R( 35)*C( 31)                                            NKM  280
      A( 11,  2) = + R( 45)*C( 24)                                            NKM  281
      A( 11, 11) = - R( 46)                                                   NKM  282
      A( 11, 24) = + R( 45)*C(  2)                                            NKM  283
      A( 12, 12) = - R( 50)*C( 31) - R( 51)*C( 31) - R( 57)*C( 31)            NKM  284
```

```
      A( 13,  1) = - R( 53)*C( 13)                                          NKM  285
      A( 13,  2) = - R( 54)*C( 13)                                          NKM  286
      A( 13, 12) = + R( 51)*C( 31)                                          NKM  287
      A( 13, 13) = - R( 53)*C(  1) - R( 54)*C(  2) - R( 55)*C( 22)          NKM  288
      A( 13, 22) = - R( 55)*C( 13)                                          NKM  289
      A( 14, 12) = + R( 50)*C( 31)                                          NKM  290
      A( 14, 13) = + R( 55)*C( 22)                                          NKM  291
      A( 14, 14) = - R( 52)*C( 31)                                          NKM  292
      A( 14, 22) = + R( 55)*C( 13)                                          NKM  293
      A( 15,  1) = + R( 53)*C( 13)                                          NKM  294
      A( 15, 13) = + R( 53)*C(  1)                                          NKM  295
      A( 15, 15) = - R( 56)                                                 NKM  296
      A( 16,  1) = - R( 58)*C( 16)                                          NKM  297
      A( 16, 12) = + R( 57)*C( 31)                                          NKM  298
      A( 16, 16) = - R( 58)*C(  1)                                          NKM  299
      A( 17,  1) = + R( 20)*C( 18)                                          NKM  300
      A( 17, 17) = - R( 21)                                                 NKM  301
      A( 17, 18) = + R( 20)*C(  1) + R( 64)*C( 29)                          NKM  302
      A( 18,  1) = - R( 20)*C( 18)                                          NKM  303
      A( 18,  8) = + R( 18)*C( 31)                                          NKM  304
      A( 18,  9) = + R( 19)*C( 31)                                          NKM  305
      A( 18, 18) = - R( 20)*C(  1) - R( 64)*C( 29)                          NKM  306
      A( 19,  1) = - R( 14)*C( 19)                                          NKM  307
      A( 19,  2) = + R( 13)*C(  3) - R( 15)*C( 19) + R( 29)*C( 25) +        NKM  308
     *    R( 30)*C( 26)                                                     NKM  309
      A( 19,  3) = + R( 13)*C(  2)                                          NKM  310
      A( 19, 19) = - R( 14)*C(  1) - R( 15)*C(  2)                          NKM  311
      A( 19, 21) = + R( 17)                                                 NKM  312
      A( 19, 25) = + R( 29)*C(  2)                                          NKM  313
      A( 19, 26) = + R( 30)*C(  2)                                          NKM  314
      A( 20,  2) = + R( 11)*C( 22)                                          NKM  315
      A( 20, 20) = - R( 22)                                                 NKM  316
      A( 20, 22) = + R( 11)*C(  2)                                          NKM  317
      A( 21,  2) = + R( 15)*C( 19)                                          NKM  318
      A( 21, 19) = + R( 15)*C(  2)                                          NKM  319
      A( 21, 21) = - R( 16)*C( 33) - R( 17)                                 NKM  320
      A( 22,  1) = - R( 10)*C( 22) + R( 58)*C( 16)                          NKM  321
      A( 22,  2) = - R( 11)*C( 22)                                          NKM  322
      A( 22,  3) = +    0.40*R( 26)*C(  9)                                  NKM  323
      A( 22,  5) = + R(  9)*C( 31)                                          NKM  324
      A( 22,  6) =          R( 47) + R( 48)*C( 31)                          NKM  325
      A( 22,  7) = + R( 43)                                                 NKM  326
      A( 22,  8) = +    0.40*R( 23)*C( 32)                                  NKM  327
      A( 22,  9) = +    0.40*R( 24)*C( 32) +    0.40*R( 26)*C(  3)          NKM  328
      A( 22, 12) = + R( 50)*C( 31)                                          NKM  329
      A( 22, 13) = - R( 55)*C( 22)                                          NKM  330
      A( 22, 14) = + R( 52)*C( 31)                                          NKM  331
      A( 22, 16) = + R( 58)*C(  1)                                          NKM  332
      A( 22, 17) = + R( 21)                                                 NKM  333
      A( 22, 20) = + R( 22)                                                 NKM  334
      A( 22, 22) = - R( 10)*C(  1) - R( 11)*C(  2) -    4.00*R( 12)         NKM  335
     *    *C( 22) - R( 49)*C( 23) - R( 55)*C( 13) - R( 60)*C( 29)           NKM  336
      A( 22, 23) = - R( 49)*C( 22)                                          NKM  337
      A( 22, 27) = + R( 40)                                                 NKM  338
      A( 23,  1) = +    0.15*R( 36)*C( 28) - R( 39)*C( 23) + R( 44)         NKM  339
     *    *C( 24)                                                           NKM  340
      A( 23,  3) = +    0.10*R( 25)*C(  8)                                  NKM  341
      A( 23,  7) = + R( 43)                                                 NKM  342
```

```
      A( 23,  8) = +    0.40*R( 23)*C( 32) +    0.10*R( 25)*C(  3)      NKM  343
      A( 23,  9) = +    0.40*R( 24)*C( 32)                              NKM  344
      A( 23, 15) = + R( 56)                                             NKM  345
      A( 23, 22) = - R( 49)*C( 23)                                      NKM  346
      A( 23, 23) = - R( 39)*C(  1) - R( 49)*C( 22) - R( 63)*C( 29)      NKM  347
      A( 23, 24) = + R( 44)*C(  1)                                      NKM  348
      A( 23, 27) = + R( 38)                                             NKM  349
      A( 23, 28) = +    0.15*R( 36)*C(  1)                              NKM  350
      A( 24,  1) = - R( 44)*C( 24)                                      NKM  351
      A( 24,  2) = - R( 45)*C( 24)                                      NKM  352
      A( 24,  7) = + R( 42)*C( 31)                                      NKM  353
      A( 24, 11) = + R( 46)                                             NKM  354
      A( 24, 24) = - R( 44)*C(  1) - R( 45)*C(  2)                      NKM  355
      A( 25,  1) = - R( 27)*C( 25)                                      NKM  356
      A( 25,  2) = - R( 29)*C( 25)                                      NKM  357
      A( 25,  3) = +    0.40*R( 25)*C(  8) +    0.80*R( 26)*C(  9)      NKM  358
      A( 25,  6) = - R( 31)*C( 25)                                      NKM  359
      A( 25,  7) = - R( 32)*C( 25)                                      NKM  360
      A( 25,  8) = +    0.40*R( 25)*C(  3)                              NKM  361
      A( 25,  9) = +    0.80*R( 26)*C(  3)                              NKM  362
      A( 25, 25) = - R( 27)*C(  1) - R( 29)*C(  2) - R( 31)*C(  6) -    NKM  363
     *    R( 32)*C(  7) - R( 61)*C( 29)                                NKM  364
      A( 26,  1) = - R( 28)*C( 26)                                      NKM  365
      A( 26,  2) = - R( 30)*C( 26)                                      NKM  366
      A( 26,  3) = +    0.40*R( 25)*C(  8)                              NKM  367
      A( 26,  6) = - R( 33)*C( 26)                                      NKM  368
      A( 26,  7) = - R( 34)*C( 26)                                      NKM  369
      A( 26,  8) = +    0.40*R( 25)*C(  3)                              NKM  370
      A( 26, 26) = - R( 28)*C(  1) - R( 30)*C(  2) - R( 33)*C(  6) -    NKM  371
     *    R( 34)*C(  7) - R( 62)*C( 29)                                NKM  372
      A( 27,  1) = +    0.85*R( 36)*C( 28) + R( 39)*C( 23)              NKM  373
      A( 27,  2) = - R( 41)*C( 27)                                      NKM  374
      A( 27, 23) = + R( 39)*C(  1) + R( 63)*C( 29)                      NKM  375
      A( 27, 27) = - R( 38) - R( 40) - R( 41)*C(  2)                    NKM  376
      A( 27, 28) = +    0.85*R( 36)*C(  1)                              NKM  377
      A( 28,  1) = - R( 36)*C( 28) - R( 37)*C( 28)                      NKM  378
      A( 28, 10) = + R( 35)*C( 31)                                      NKM  379
      A( 28, 28) = - R( 36)*C(  1) - R( 37)*C(  1)                      NKM  380
C                                                                       NKM  381
      IF(N.LE.28)    RETURN                                             NKM  382
C                                                                       NKM  383
C                                                                       NKM  384
C         ELEMENTS OF THE JACOBIAN FOR SO2 AND SO4 REACTIONS            NKM  385
C                                                                       NKM  386
      A(  6, 29) = + R( 61)*C( 25)                                      NKM  387
      A(  7, 29) = + R( 62)*C( 26)                                      NKM  388
      A( 17, 29) = + R( 64)*C( 18)                                      NKM  389
      A( 18, 29) = - R( 64)*C( 18)                                      NKM  390
      A( 22, 29) = - R( 60)*C( 22)                                      NKM  391
      A( 23, 29) = - R( 63)*C( 23)                                      NKM  392
      A( 25, 29) = - R( 61)*C( 25)                                      NKM  393
      A( 26, 29) = - R( 62)*C( 26)                                      NKM  394
      A( 27, 29) = + R( 63)*C( 23)                                      NKM  395
      A( 29, 18) = - R( 64)*C( 29)                                      NKM  396
      A( 29, 22) = - R( 60)*C( 29)                                      NKM  397
      A( 29, 23) = - R( 63)*C( 29)                                      NKM  398
      A( 29, 25) = - R( 61)*C( 29)                                      NKM  399
      A( 29, 26) = - R( 62)*C( 29)                                      NKM  400
```

```
      A( 29, 29) = - R( 59)*C( 31) - R( 60)*C( 22) - R( 61)*C( 25) -     NKM  401
     *    R( 62)*C( 26) - R( 63)*C( 23) - R( 64)*C( 18)                  NKM  402
      A( 30, 18) = + R( 64)*C( 29)                                       NKM  403
      A( 30, 22) = + R( 60)*C( 29)                                       NKM  404
      A( 30, 23) = + R( 63)*C( 29)                                       NKM  405
      A( 30, 25) = + R( 61)*C( 29)                                       NKM  406
      A( 30, 26) = + R( 62)*C( 29)                                       NKM  407
      A( 30, 29) = + R( 59)*C( 31) + R( 60)*C( 22) + R( 61)*C( 25) +     NKM  408
     *    R( 62)*C( 26) + R( 63)*C( 23) + R( 64)*C( 18)                  NKM  409
C                                                                        NKM  410
      RETURN                                                             NKM  411
      END                                                                NKM  412



      SUBROUTINE KEMOD2(INTIM,TSTOP,NUMFLX,FLXUNT)                       NKM  413
C                                                                        NKM  414
C     -------  KEMOD2 IS THE SECONDARY DRIVER FOR THE CHEMICAL/          NKM  415
C              DIFFUSION MODULE OF THE ERT TRAJECTORY MODEL.             NKM  416
C                                                                        NKM  417
C              KEMOD2 CONTROLS ALL I/O EXCEPT INITIAL EMISSION INPUTS.   NKM  418
C              KEMOD2 CONTROLS THE UPDATING PROCESSES.                   NKM  419
C                                                                        NKM  420
C     -------  FEATURES OF KEMOD2                                        NKM  421
C                  INTEGRATION BY EPISODE PACKAGE                        NKM  422
C                  VARIABLE SIZE VERTICAL MESH                           NKM  423
C                  GENERALIZED BOUNDARY CONDITIONS                       NKM  424
C                  TIME + HEIGHT DEPENDENT PHOTODISSOCIATION RATES,      NKM  425
C                   WITH SKY CLEARNESS RATIOS                            NKM  426
C                  TEMPERATURE DEPENDENT REACTION RATE CONSTANTS          NKM  427
C                                                                        NKM  428
      REAL INTIM                                                         NKM  429
      INTEGER BCFLAG                                                     NKM  430
      COMMON/CHEM1/ NOSTAT,              NOSTM1,             NOREAC,     NKM  431
     1              NOSPEC,              NSTOY,              NK          NKM  432
      COMMON/CHEM2/ CONIN(40,5),         WTMOLE(43),         RATKON(65), NKM  433
     1              RATEFF(65),          RATEV(4,5),         QRATE,      NKM  434
     2              NVRATE,              LOCVRT(4)                       NKM  435
      COMMON/CHEM3/ ZEE(5),              DELZ(4),            HTCELL(6),  NKM  436
     1              TDELZ(2) ,           DFINIT(6),          SCALOW(4),  NKM  437
     2              DCOF(5),             FLXWAL(40),         FLXDGE(40), NKM  438
     3              BCFLAG(40),          DPRATE(10),         DEPOWR(10), NKM  439
     4              LOCDPF(10),          NDPFLX ,            SCALUP(4)   NKM  440
      COMMON /CHEM4/  RLAT,  RLONG,  TMZONE,  HIRATE,  JDATE             NKM  441
      COMMON /CHEM5/  REACT(20,65),SPEC(40),LOCFLX(10),                  NKM  442
     1                NASFLX,FLXW1(30),FLXW2(30)                         NKM  443
      COMMON/INPUTS/TITLE(20),           IDATE(10),          NCURV       NKM  444
      COMMON/FLUXES/FLXIN( 7,200),       FLXTIM(200),        NFLUX       NKM  445
      COMMON /PS1/  TPASS(200),          PS(7,5,75),         FRACT(3),   NKM  446
     1              NPTSR,               NPSFLX,             LOCPSF(7)   NKM  447
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPOINT    NKM  448
      COMMON/EPCOM6/PW(4500)                                             NKM  449
      COMMON/PRPLOT/KPS(5) ,             KSYM(5),            GRCON(5,100)NKM  450
     1    ,      TOUT(100),   VALMAX(5)                                  NKM  451
      COMMON/EPCOM9/HUSED,               NQUSED,             NSTEP,      NKM  452
     1              NFE,                 NJE                             NKM  453
      DIMENSION     YO(150),             PPM (40,5),         TEMP(25),   NKM  454
     1              TMTEMP(25)  ,        HTINV(100,8),       WORK(200),  NKM  455
```

```
     2              CLOUDT(25),         CLOUDF(25)          ,VFR(5,200), NKM  456
     3              PTSR(7,200)                                          NKM  457
C                                                                         NKM  458
      EQUIVALENCE (YES,IES),  (RNEG,NEG)                                  NKM  459
      EQUIVALENCE (PPM ,CONIN)                                            NKM  460
      EQUIVALENCE (VFR,PH),  (PTSR,PW(1001))                              NKM  461
C                                                                         NKM  462
      DATA LUP, LIN, LOUT, YES, RNEG  /1, 3, 6, 3HYES, 2HNO/              NKM  463
      DATA KSYM /1HN, 1H2, 1HO, 1HS, 1HC  /                               NKM  464
      DATA RMOLE /4HMOLE/                                                 NKM  465
      DATA NROW,MAXR,MAXMSH,MAXNK,NRHT,KTMAX /40,55,5,30,100,100/         NKM  466
      DATA MF,IERROR /21,3/                                               NKM  467
C                                                                         NKM  468
C------------ READ INPUTS FOR COMPUTING POLLUTANT CONCENTRATIONS           NKM  469
C                                                                         NKM  470
      READ(LIN,30) DELT                                                   NKM  471
      READ(LIN,30) BIGSTP                                                 NKM  472
      READ(LIN,30) EPS                                                    NKM  473
      READ(LIN,30) UPDINT                                                 NKM  474
      READ(LIN,30) PRNTIN                                                 NKM  475
      READ(LIN,29) TIMOUT                                                 NKM  476
      READ(LIN,29) SUNGEN                                                 NKM  477
      READ(LIN,29) HIRATE                                                 NKM  478
      READ(LIN,29) QCLOUD                                                 NKM  479
      READ(LIN,29) VTEMP                                                  NKM  480
      READ(LIN,29) SPUNCH                                                 NKM  481
      READ(LIN,31) NUMSTP                                                 NKM  482
      READ(LIN,31) NOREAC                                                 NKM  483
      READ(LIN,31) NOSPEC                                                 NKM  484
      READ(LIN,31) NSTDY                                                  NKM  485
      READ(LIN,31) NOSTAT                                                 NKM  486
      READ(LIN,30) (ZEE(I), I=1,NOSTAT)                                   NKM  487
C                                                                         NKM  488
      NK = NOSPEC -NSTDY                                                  NKM  489
      NPUNCH = NK                                                         NKM  490
      SUNTIM = UPDINT                                                     NKM  491
      RDCHEM = YES                                                        NKM  492
      QRATE = YES                                                         NKM  493
      F = YES                                                             NKM  494
      TSTOP2 = TSTOP                                                      NKM  495
      KOK = 100                                                           NKM  496
      NASFLX = NUMFLX                                                     NKM  497
C                                                                         NKM  498
C     ---- NOSPEC IS THE TOTAL NUMBER OF SPECIE    (MAX = 40)            NKM  499
C     ---- NSTDY IS THE NUMBER OF SPECIE HELD CONSTANT OR IN STEADY STATNKM  500
C     ---- NK IS THE NUMBER OF SPECIE INTEGRATED    (MAXNK = 30)         NKM  501
C     ---- NOSTAT IS THE NUMBER OF VERTICAL STATIONS (MIN=4,MAX=5)       NKM  502
C     -----LOCFLX IS LOCATION  INDEX FOR AREA SOURCE EMISSION FLUXES     NKM  503
C     -----NUMFLX IS NUMBER OF AREA SOURCE EMISSION FLUXES               NKM  504
C     -----LOCPSF IS LOCATION  INDEX OF POINT SOURCE EMISSION FLUXES     NKM  505
C     -----NPSFLX IS NUMBER OF POINT SOURCE EMMISION FLUXES              NKM  506
C     -----LOCDPF IS LOCATION  INDEX FOR DEPOSITING FLUXES               NKM  507
C     -----NDPFLX IS NUMBER OF DEPOSITING FLUXES                         NKM  508
C                                                                         NKM  509
C     --------------BOUNDARY CONDITION CODE FOR WALL--------------------NKM  510
C                                                                         NKM  511
C     BCFLAG = 0    DC/DZ = PHI/KZ    WHERE    PHI = 0.  (DEFAULT CASE)  NKM  512
C                                     OR       PHI = AREA SOURCE EMISSION FNKM  513
```

```
C                                       AND     KZ  = DIFFUSION COEFFICIENT NKM  514
C                                                                          NKM  515
C     BCFLAG = 1   DC/DZ = -DPRATE*(PPM**DEPOWR)/KZ  DEPOSITING SPECIE     NKM  516
C     -----WHERE                                                           NKM  517
C     -----DPRATE(I) IS THE DEPOSITION VELOCITY OF THE I*TH  SPECIE        NKM  518
C     -----IN METERS/MINUTE/(PPM**(DEPOWR-1))                              NKM  519
C     -----INPUT AS POSITIVE QUANTITY FOR UPTAKE AT WALL                   NKM  520
C                                                                          NKM  521
C     BCFLAG = 2    C = INITIAL CONCENTRATION FOR ALL TIME                 NKM  522
C                                                                          NKM  523
C     ---------------------------------------------------------------------NKM  524
C                                                                          NKM  525
C     ------  ZERO PPM ARRAY  AND CHECK NK AND NOSTAT INPUTS               NKM  526
      NRC = MAXMSH*NROW                                                    NKM  527
      CALL XMIT (-NRC,0.,CONIN)                                            NKM  528
      IF(NOSTAT.GE.4)  GO TO 200                                           NKM  529
      WRITE(LOUT,77) NOSTAT                                                NKM  530
      GO TO 800                                                            NKM  531
  200 CONTINUE                                                             NKM  532
      IF(NK.LE.MAXNK)  GO TO 210                                           NKM  533
      WRITE(LOUT,76) NOSPEC ,NSTDY, NK                                     NKM  534
      GO TO 800                                                            NKM  535
  210 CONTINUE                                                             NKM  536
      IF(ROCHEM.NE.YES)  GO TO 219                                         NKM  537
      NDPFLX = 0                                                           NKM  538
      DO 215  I=1,NOSPEC                                                   NKM  539
      READ(LIN,51)  SPEC(I),WTMOLE(I),BCFLAG(I)                            NKM  540
      IF(BCFLAG(I) .EQ. 1)  NDPFLX = NDPFLX + 1                            NKM  541
  215 CONTINUE                                                             NKM  542
      IF(NUMFLX.GT.0) READ(LIN,31) (LOCFLX(I),I=1,NUMFLX)                  NKM  543
      IF(NPSFLX.GT.0) READ(LIN,31) (LOCPSF(I),I=1,NPSFLX)                  NKM  544
      IF(NDPFLX.GT.0) READ(LIN,53) (LOCDPF(I),DPRATE(I),DEPOWR(I),         NKM  545
     1  I=1,NDPFLX)                                                        NKM  546
      IF(NDPFLX.LE.0)  GO TO 219                                           NKM  547
C     ------ ADJUST DEPOSITION VELOCITIES SO AS TO EFFECT 10 METER LAYER   NKM  548
      A = .5*(ZEE(2)-ZEE(1))                                               NKM  549
      A = 10./A                                                            NKM  550
      DO 218 I = 1,NDPFLX                                                  NKM  551
  218 DPRATE(I) = A*DPRATE(I)                                              NKM  552
  219 CONTINUE                                                             NKM  553
      DO 220 I = 1,NOSPEC                                                  NKM  554
      READ(LIN,41) (PPM(I,J),J=1,NOSTAT)                                   NKM  555
  220 CONTINUE                                                             NKM  556
      IF(ROCHEM.NE.YES)  GO TO 230                                         NKM  557
      DO 222 I=1,NOREAC                                                    NKM  558
      READ(LIN,30) RATKON(I)                                               NKM  559
  222 CONTINUE                                                             NKM  560
      DO 224 J=1,NOREAC                                                    NKM  561
      READ(LIN,40) (REACT(I,J),I=1,20)                                     NKM  562
  224 CONTINUE                                                             NKM  563
C                                                                          NKM  564
  230 CONTINUE                                                             NKM  565
C                                                                          NKM  566
      TIME = INTIM                                                         NKM  567
C                                                                          NKM  568
C     ------- THE PHOTODISSOCIATION RATES VARY WITH HEIGHT IF HIRATE=YES   NKM  569
C     ------- LOCVRT IS THE SPECIES INDEX FOR THE RATES (INPUT)            NKM  570
C                                                                          NKM  571
```

```
      READ(LIN,30) RLAT                                                 NKM  572
      READ(LIN,30) RLONG                                                NKM  573
      READ(LIN,30) TMZONE                                               NKM  574
      READ(LIN,31) JDATE                                                NKM  575
      READ(LIN,31)  NVRATE                                              NKM  576
      READ(LIN,31) (LOCVRT(I),I=1,NVRATE)                               NKM  577
C                                                                       NKM  578
      IF(SUNGEN.EQ.YES)  GO TO 244                                      NKM  579
      WRITE(LOUT,22)                                                    NKM  580
      STOP                                                              NKM  581
  244 CONTINUE                                                          NKM  582
C                                                                       NKM  583
      IF(QCLOUD.EQ.RNEG)  GO TO 249                                     NKM  584
C     ------- INPUT SKY COVER FACTORS AND TIMES                         NKM  585
C     ------- LAST SKY COVER UPDATE TIME (CLOUDT) MUST BE A NEGATIVE    NKM  586
      DO 246  I=1,26                                                    NKM  587
      READ(LIN,59)  A, B                                                NKM  588
      IF(A.LT.0.0)  GO TO 248                                           NKM  589
      IF(I.GT.25)  GO TO 247                                            NKM  590
      CLOUDT(I) = A                                                     NKM  591
      CLOUDF(I) = B                                                     NKM  592
      NCLOUD = I                                                        NKM  593
  246 CONTINUE                                                          NKM  594
      GO TO 248                                                         NKM  595
  247 WRITE(LOUT,57)                                                    NKM  596
      GO TO 800                                                         NKM  597
  248 CONTINUE                                                          NKM  598
      ICLOUD = 1                                                        NKM  599
      CLOUDY = CLOUDF(1)                                                NKM  600
  249 IF(QCLOUD.EQ.RNEG)  CLOUDY= 1.0                                   NKM  601
C     ------  CALCULATE INITIAL PHOTODISSOCIATION RATES                 NKM  602
      CALL PHOTOR(TIME,NOSTAT,ZEE,CLOUDY,RATEV)                         NKM  603
      CALL RATEHI(1)                                                    NKM  604
260   CONTINUE                                                          NKM  605
6                                                                       NKM  606
C     ------------ READ TEMPERATURES IF VTEMP = YES                     NKM  607
      ITEMP = 1                                                         NKM  608
      NTEMP = 1                                                         NKM  609
      TEMP(1) = 298.2                                                   NKM  610
      IF(VTEMP.NE.YES)  GO TO 265                                       NKM  611
      DO 261 I = 1,26                                                   NKM  612
      READ(LIN,59)  A, B                                                NKM  613
      IF(I.GT.25)   GO TO 262                                           NKM  614
      IF(A.LT.0.)  GO TO 265                                            NKM  615
      TMTEMP(I) = A                                                     NKM  616
      TEMP(I) = B + 273.2                                               NKM  617
      NTEMP = I                                                         NKM  618
  261 CONTINUE                                                          NKM  619
  262 CONTINUE                                                          NKM  620
      WRITE(LOUT,79)                                                    NKM  621
      GO TO 800                                                         NKM  622
  265 CONTINUE                                                          NKM  623
      CALL TEMPR (ITEMP,INTIM,TEMP)                                     NKM  624
C                                                                       NKM  625
C     -------- READ TABLE OF DIFFUSIVITY UPDATE TIMES, MIXING HEIGHTS,  NKM  626
C              AND DIFFUSIVITY COEFFICIENTS.                            NKM  627
C              MIXING HEIGHT INFORMATION IS OPTIONAL.                   NKM  628
      K = NOSTAT+3                                                      NKM  629
```

```
      DO 270 I = 1,101                                                  NKM  630
      READ(LIN,41) A,B,(HTINV(I,J),J=3,K)                               NKM  631
      IF (A.LT.0.) GO TO 280                                            NKM  632
      HTINV(I,1) = A                                                    NKM  633
      HTINV(I,2) = B                                                    NKM  634
270   CONTINUE                                                          NKM  635
      PRINT 45                                                          NKM  636
      GO TO 800                                                         NKM  637
280   NHT = I-1                                                         NKM  638
      IHT = 1                                                           NKM  639
      KHT = 1                                                           NKM  640
290   CONTINUE                                                          NKM  641
C                                                                       NKM  642
C     INITIALIZE PLOT PARAMETERS                                        NKM  643
C      FOR                                                              NKM  644
      KPS(1) = 1                                                        NKM  645
      KPS(2) = 2                                                        NKM  646
      KPS(3) = 3                                                        NKM  647
      KPS(4) = 29                                                       NKM  648
      KPS(5) = 5                                                        NKM  649
      NCURV = 5                                                         NKM  650
      CALL XMIT(-NCURV,0.0,VALMAX)                                      NKM  651
C                                                                       NKM  652
C      BANNER PAGE                                                      NKM  653
C                                                                       NKM  654
      WRITE(LOUT,99)                                                    NKM  655
C                                                                       NKM  656
C     ----- REGULATE THE AREA SOURCE EMISSION FLUX SCHEDULES            NKM  657
      IF(NUMFLX.EQ.0)  GO TO 299                                        NKM  658
      DO 297 I=1,NUMFLX                                                 NKM  659
      DO 293 J=1,NFLUX                                                  NKM  660
      WORK(J) = FLXIN(I,J)                                              NKM  661
  293 CONTINUE                                                          NKM  662
      CALL SKEDUL(WORK,FLXTIM,NFLUX,NUPDAT,UPDINT,INTIM,TSTOP,-1.,F,-1) NKM  663
      IF(F.EQ. YES)  GO TO 800                                          NKM  664
C     -----   CONVERT FLUXES FROM MASS OR MOLE FRACTION TO PPM          NKM  665
      ID = LOCFLX(I)                                                    NKM  666
      WT = 1.E6                                                         NKM  667
      IF(FLXUNT.NE.RMOLE)    WT = 28.97E6 /WTMOLE(ID)                   NKM  668
      DO 295 J=1,NUPDAT                                                 NKM  669
      FLXIN(I,J) = WORK(J)*WT                                           NKM  670
  295 CONTINUE                                                          NKM  671
  297 CONTINUE                                                          NKM  672
C     ----------  THE NEW SCHEDULE OF FLUXES IS STORED IN FLXIN         NKM  673
      NFLUX = NUPDAT                                                    NKM  674
      IFLXTM = 1                                                        NKM  675
  299 CONTINUE                                                          NKM  676
C     ------- INITIALIZE FLUXES                                         NKM  677
      NPMX = 5*MAXNK                                                    NKM  678
      CALL  XMIT(-NPMX,0.0,PSR1)                                        NKM  679
      CALL  XMIT(-NPMX,0.0,PSR2)                                        NKM  680
      CALL  XMIT(-MAXNK,0.0,FLXW1)                                      NKM  681
      CALL  XMIT(-MAXNK,0.0,FLXW2)                                      NKM  682
      CALL XMIT(-NROW,0.,FLXWAL)                                        NKM  683
      CALL XMIT(-NROW,0.,FLXDGE)                                        NKM  684
      CALL UPFLX1(TIME,IFLXTM)                                          NKM  685
C                                                                       NKM  686
C     -----   GENERATE VERTICAL MESH PARAMETERS                         NKM  687
```

```
C                                                                       NKM  688
      NOSTM1 = NOSTAT - 1                                               NKM  689
      DELZ(1) = ZEE(2) - ZEE(1)                                         NKM  690
      HTCELL(1) = DELZ(1)                                               NKM  691
      DO 300 K = 2,NOSTM1                                               NKM  692
      DELZ(K) = ZEE(K+1) - ZEE(K)                                       NKM  693
      HTCELL(K) = .5*(DELZ(K) + DELZ(K-1))                              NKM  694
300   CONTINUE                                                          NKM  695
      HTCELL(NOSTAT) = DELZ(NOSTM1)                                     NKM  696
C                                                                       NKM  697
C      ----- CONVERT POINT SOURCE EMISSIONS FROM ABSOLUTE MASS          NKM  698
C            OR MOLES TO RATES ON A REGULAR UPDATE SCHEDULE.            NKM  699
C                                                                       NKM  700
C      ----- POINT SOURCE MASS IS BLED IN OVER (UP TO) THREE            NKM  701
C            UPDATE INTERVALS. THE FRACTION IN EACH INTERVAL            NKM  702
C            IS SET BY THE FRACT ARRAY (SEE KEMOD DATA STATEMENT).     KNKM  703
C                                                                       NKM  704
      IF(NPSFLX.LE.0)  GO TO 307                                        NKM  705
      CALL XMIT(-2625,0.0,PS)                                           NKM  706
      A = 1.E6/UPDINT                                                   NKM  707
      IF(FLXUNT.EQ.RMOLE)  GO TO 303                                    NKM  708
      DO 302 K = 1,NPSFLX                                               NKM  709
      ID =LOCPSF(K)                                                     NKM  710
      WORK(K) = 28.97/MTMOLE(ID)                                        NKM  711
302   CONTINUE                                                          NKM  712
303   CONTINUE                                                          NKM  713
      B = INTIM - 0.499*UPDINT                                          NKM  714
      K4 = 0                                                            NKM  715
      DO 305 K = 1,NPTSR                                                NKM  716
      K1 = IFIX( (TPASS(K)-B)/UPDINT) + 1                               NKM  717
      K2 = K1 + 1                                                       NKM  718
      K3 = K1 + 2                                                       NKM  719
      K4 = MAXO(K4,K3)                                                  NKM  720
      DO 305 J = 1,NOSTAT                                               NKM  721
      VF = A*VFR(J,K)/HTCELL(J)                                         NKM  722
      IF(J.EQ.1  .OR.  J.EQ.NOSTAT)  VF = 2.*VF                         NKM  723
      DO 305 I = 1,NPSFLX                                               NKM  724
      C = VF*PTSR(I,K)                                                  NKM  725
      IF(FLXUNT.NE.RMOLE) C = C*WORK(I)                                 NKM  726
      PS(I,J,K1) = PS(I,J,K1) + FRACT(1)*C                              NKM  727
      PS(I,J,K2) = PS(I,J,K2) + FRACT(2)*C                              NKM  728
      PS(I,J,K3) = PS(I,J,K3) + FRACT(3)*C                              NKM  729
305   CONTINUE                                                          NKM  730
      NPTSR = K4                                                        NKM  731
      TPASS(1) = INTIM                                                  NKM  732
      DO 306 K = 2,NPTSR                                                NKM  733
      TPASS(K) = TPASS(K-1) + UPDINT                                    NKM  734
306   CONTINUE                                                          NKM  735
      N = MAXNK*NOSTAT                                                  NKM  736
      CALL XMIT(-N,0.0,PSRATE)                                          NKM  737
      CALL XMIT(-2400,0.0,PW)                                           NKM  738
307   IPS = 1                                                           NKM  739
      CALL UPSORC(TIME,IPS,NOSTAT,SPEC)                                 NKM  740
C                                                                       NKM  741
C      ------- PRINT INPUTS                                             NKM  742
C                                                                       NKM  743
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  744
      WRITE(LOUT,16) INTIM,TSTOP,DELT,BIGSTP,NOREAC,NOSPEC,NSTOY,NOSTAT NKM  745
```

```
      IF(NUMFLX.EQ.0)  GO TO 311                                        NKM  746
      DO 308 I= 1,NUMFLX                                                NKM  747
      ID = LOCFLX(I)                                                    NKM  748
      WORK(I) = SPEC(ID)                                                NKM  749
308   CONTINUE                                                          NKM  750
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  751
      WRITE(LOUT,68)  (WORK(I) , I=1,NUMFLX)                            NKM  752
      DO 310  J = 1, NFLUX                                              NKM  753
      IF(MOD(J,47).NE.0)  GO TO 309                                     NKM  754
      CALL NEWPAG (TITLE,0,IDATE)                                       NKM  755
      WRITE(LOUT,68)  (WORK(I), I=1,NUMFLX)                             NKM  756
309   WRITE(LOUT,69) FLXTIM(J), (FLXIN(I,J), I=1,NUMFLX)                NKM  757
      FLXTIM(J+1)=FLXTIM(J) + UPDINT                                    NKM  758
310   CONTINUE                                                          NKM  759
311   CONTINUE                                                          NKM  760
      IF(NPSFLX.EQ.0)  GO TO 317                                        NKM  761
      DO 312  K=1,NPSFLX                                                NKM  762
      ID = LOCPSF(K)                                                    NKM  763
      WORK(K) = SPEC(ID)                                                NKM  764
312   CONTINUE                                                          NKM  765
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  766
      WRITE(LOUT,83)  (WORK(K),K=1,NPSFLX)                              NKM  767
      DO 314 K = 1,NPTSR                                                NKM  768
      IF(MOD(K,8).NE.0)  GO TO 313                                      NKM  769
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  770
      WRITE(LOUT,83)  (WORK(KK),KK=1,NPSFLX)                            NKM  771
313   WRITE(LOUT,84)  TPASS(K), (PS(I,1,K),I=1,NPSFLX)                  NKM  772
      DO 314 J =2,NOSTAT                                                NKM  773
      WRITE(LOUT,85)  (PS(I,J,K),I=1,NPSFLX)                            NKM  774
314   CONTINUE                                                          NKM  775
317   CONTINUE                                                          NKM  776
C                                                                       NKM  777
      CALL XMIT(NOREAC,RATKON,RATEFF)                                   NKM  778
      CALL UNMIXR(1)                                                    NKM  779
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  780
      WRITE(LOUT,37)                                                    NKM  781
      DO 320 I=1,NOREAC                                                 NKM  782
      IF(MOD(I,44).NE.0)  GO TO 319                                     NKM  783
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  784
      WRITE(LOUT,37)                                                    NKM  785
319   WRITE(LOUT,38) I, (REACT(J,I),J=1,20), RATEFF(I)                  NKM  786
320   CONTINUE                                                          NKM  787
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  788
      WRITE(LOUT,28)                                                    NKM  789
      DO 330 J = 1,NOSPEC                                               NKM  790
 325  WRITE(LOUT,27)  J,  SPEC(J), WTMOLE(J), BCFLAG(J)                 NKM  791
  330 CONTINUE                                                          NKM  792
C                                                                       NKM  793
C     ------  SCHEDULE AND INITIALIZE DIFFUSIVITIES                     NKM  794
C                                                                       NKM  795
      IF(NHT.EQ.1)   GO TO 360                                          NKM  796
      K = NOSTAT + 3                                                    NKM  797
      TSTOP = HTINV(NHT,1)                                              NKM  798
      CALL XMIT (NHT,HTINV(1,1),WORK(1))                                NKM  799
      DO 350 J =2,K                                                     NKM  800
      CALL SKEDUL(HTINV(1,J),WORK,NHT,NINT,UPDINT,INTIM,TSTOP,-1.,F,1)  NKM  801
      IF(F.EQ.YES)  GO TO 800                                           NKM  802
350   CONTINUE                                                          NKM  803
```

```
      NHT = NINT                                                        NKM  804
      HTINV(1,1) = INTIM                                                NKM  805
      DO 355 I =2,NHT                                                   NKM  805
      HTINV(I,1) = HTINV(I-1,1) + UPDIN1                                NKM  807
355   CONTINUE                                                          NKM  808
      TSTOP = TSTOP2                                                    NKM  809
360   CONTINUE                                                          NKM  810
      K = NOSTAT + 1                                                    NKM  811
      DO 370 I =1,K                                                     NKM  812
      J = I + 2                                                         NKM  813
      DFINIT(I) = HTINV(1,J)                                            NKM  814
370   CONTINUE                                                          NKM  815
      CALL DIFCOF(NOSTAT)                                               NKM  816
C      ----     INITIALIZE RATEFF                                       NKM  817
      CALL XMIT(NOREAC,RATKON,RATEFF)                                   NKM  818
C                                                                       NKM  819
410   IPROD = NK*NOSTAT                                                 NKM  820
      TDELZ(1) = 2./HTCELL(1)                                           NKM  821
      TDELZ(2) = 2./HTCELL(NOSTAT)                                      NKM  822
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  823
      WRITE(LOUT,14)                                                    NKM  824
      HTCELL(1) = HTCELL(1)/2.                                          NKM  825
      HTCELL(NOSTAT) = HTCELL(NOSTAT)/2.                                NKM  826
      I = 1                                                             NKM  827
      WRITE(LOUT,44) I,ZEE(I),DFINIT(I),HTCELL(I)                       NKM  828
      DO 420 I = 2,NOSTAT                                               NKM  829
      WRITE(LOUT,43) DFINIT(I), DELZ(I-1)                               NKM  830
      IF(I.EQ.NOSTAT)  GO TO 415                                        NKM  831
      WRITE(LOUT,42)  I, ZEE(I), HTCELL(I)                              NKM  832
      GO TO 420                                                         NKM  833
415   WRITE(LOUT,44) I,ZEE(I),DFINIT(I),HTCELL(I)                       NKM  834
420   CONTINUE                                                          NKM  835
      HTCELL(1) = HTCELL(1)*2.                                          NKM  836
      HTCELL(NOSTAT) = HTCELL(NOSTAT)*2.                                NKM  837
C                                                                       NKM  838
C     ----- INITIALIZE INTERNALLY COMPUTED CONCENTRATIONS               NKM  839
      CALL ISTATE                                                       NKM  840
C     ----------  INITIALIZE YO VECTOR OF CONCENTRATIONS FOR DRIVE       NKM  841
      DO 425 I=1,NOSTAT                                                 NKM  842
      K= NK*(I-1) + 1                                                   NKM  843
      CALL XMIT(NK,CONIN(1,I),YO(K))                                    NKM  844
  425 CONTINUE                                                          NKM  845
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  846
      ITIM = ITHOUR(TIME)                                               NKM  847
      WRITE(LOUT,97)   INTIM, ITIM                                      NKM  848
      WRITE(LOUT,18)  (SPEC(J),J=1,NOSPEC)                              NKM  849
      DO 450 K=1,NOSTAT                                                 NKM  850
      WRITE(LOUT,32)  ZEE(K), ( PPM(J,K),J=1,NOSPEC)                    NKM  851
450   CONTINUE                                                          NKM  852
C                                                                       NKM  853
      IF(SPUNCH.NE.YES)  GO TO 454                                      NKM  854
C     ------  WRITE I.C. ON TAPE LUP FOR PUNCH OR FILE STORAGE          NKM  855
      WRITE(LUP,80) TITLE                                               NKM  856
      WRITE(LUP,81) TIME,  (ZEE(K),K=1,NOSTAT)                          NKM  857
      NOELV = 1                                                         NKM  858
      DO 453 K=1,NOELV                                                  NKM  859
      WRITE(LUP,81) ( PPM(J,K),J=1,NPUNCH)                              NKM  860
453   CONTINUE                                                          NKM  861
```

```
454   CONTINUE                                                          NKM  862
      IF(NCURV.EQ.0) GO TO 459                                          NKM  863
      KT = 1                                                            NKM  864
      DO 455 I = 1,NCURV                                                NKM  865
      J = KPS(I)                                                        NKM  866
      IF(J.NE.5)  GO TO 456                                             NKM  867
C     SCALE CO FOR PLOTTING                                             NKM  868
      GRCON(I,KT) = .10*PPM(J,1)                                        NKM  869
      IF(GRCON(I,KT).GT.VALMAX(I)) VALMAX(I)=GRCON(I,KT)                NKM  870
      GO TO 455                                                         NKM  871
456   CONTINUE                                                          NKM  872
      GRCON(I,KT) = PPM(J,1)                                            NKM  873
      IF(PPM(J,1).GT.VALMAX(I)) VALMAX(I) = PPM(J,1)                    NKM  874
  455 CONTINUE                                                          NKM  875
      TOUT(KT) = INTIM                                                  NKM  876
  459 CONTINUE                                                          NKM  877
C     ----------   INITIALIZE ADDITIONAL PARAMETERS FOR INTEGRATION     NKM  878
      NSTEP = 0                                                         NKM  879
      N = IPROD                                                         NKM  880
      KNTER = 0                                                         NKM  881
      TPRINT = INTIM + PRNTIN -.01                                      NKM  882
      RECALL = YES                                                      NKM  883
      UPD4 = UPDINT/4.0                                                 NKM  884
C                                                                       NKM  885
C     ------------- THE INTEGRATION TIME CYCLE BEGINS HERE -------------NKM  886
C                                                                       NKM  887
      CALL TIMEX(0,A,B)                                                 NKM  888
  460 CONTINUE                                                          NKM  889
      OLDTIM = TIME                                                     NKM  890
      IF(RECALL .NE. YES)  GO TO 470                                    NKM  891
      INDEX = 1                                                         NKM  892
      TOUTEP = OLDTIM + UPDINT/10.                                      NKM  893
      IF(KNTER .GT. 1)   DELT = .01                                     NKM  894
  470 IF(NSTEP .GT. NUMSTP)  GO TO 800                                  NKM  895
      TLAST = TIME                                                      NKM  896
C                                                                       NKM  897
C                    INTEGRATE BY EPISODE                               NKM  898
C                                                                       NKM  899
      CALL DRIVE(N,TIME,DELT,YO,TOUTEP,EPS,IERROR,MF,INDEX,BIGSTP,KOK)  NKM  900
C                                                                       NKM  901
C                                                                       NKM  902
      IF(KOK.GE.0)  GO TO 475                                           NKM  903
      ISTOP = IES                                                       NKM  904
      WRITE(LOUT,60)                                                    NKM  905
      GO TO 480                                                         NKM  906
475   IF(INDEX .LT. 0)  GO TO 800                                       NKM  907
      INDEX = 2                                                         NKM  908
      KNTER = KNTER + 1                                                 NKM  909
      IF(RECALL.NE. YES)   GO TO 480                                    NKM  910
C     MAKE SHORT CALL TO EPISODE (DRIVE) IF RECALL EQUALS YES           NKM  911
C     THEN RESET TOUTEP TO ACTUAL TOUT ON NEXT CALL                     NKM  912
      TOUTEP = OLDTIM + UPDINT                                          NKM  913
      RECALL = RNEG                                                     NKM  914
      GO TO 470                                                         NKM  915
480   TOUTEP = OLDTIM + 2.*UPDINT                                       NKM  916
C     ------------ UNPACK YO FOR CHECKING AND OUTPUT                    NKM  917
      DO 500 I=1,NOSTAT                                                 NKM  918
      K = NK*(I-1) + 1                                                  NKM  919
```

```
      CALL XMIT(NK,YO(K),CONIN(1,I)  )                                  NKM  920
500   CONTINUE                                                          NKM  921
620   CONTINUE                                                          NKM  922
      CALL PRODUK(UPDINT)                                               NKM  923
C                                                                       NKM  924
      IF(TIME.LT.TPRINT)  GO TO 694                                     NKM  925
      TPRINT = TPRINT + PRNTIN                                          NKM  926
 650  CONTINUE                                                          NKM  927
C                                                                       NKM  928
C     ------- OUTPUT                                                   NKM  929
C                                                                       NKM  930
      CALL NEWPAG(TITLE,0,IDATE)                                        NKM  931
      IF(NOPFLX. EQ. 0)  GO TO 682                                      NKM  932
      DO 681 J=1,NOPFLX                                                 NKM  933
      I= LOCOPF(J)                                                      NKM  934
      SINK =- OPRATE(J)*(PPM(I,1)**OEPOWR(J))                           NKM  935
  681 WRITE(10,88)  SPEC(I),  SINK                                      NKM  936
  682 CONTINUE                                                          NKM  937
      ITIM = ITHOUR(TIME)                                               NKM  938
      WRITE(LOUT,91)  TIME, ITIM, RATEV(1,1), DFINIT(1)                 NKM  939
      IF(NUMFLX.LE.0)  GO TO 656                                        NKM  940
      DO 655 I = 1,NUMFLX                                               NKM  941
      K = LOCFLX(I)                                                     NKM  942
      WORK(I) = SPEC(K)                                                 NKM  943
      WORK(I+10) = FLXWAL(K)*TDELZ(1)                                   NKM  944
655   CONTINUE                                                          NKM  945
656   IF(NPSFLX.LE.0)  GO TO 661                                        NKM  946
      DO 660 I = 1,NPSFLX                                               NKM  947
      K = LOCPSF(I)                                                     NKM  948
      WORK(I+20) = SPEC(K)                                              NKM  949
      WORK(I+30) = PSRATE(K,1)                                          NKM  950
660   CONTINUE                                                          NKM  951
661   CONTINUE                                                          NKM  952
      NMX = MAX0(NUMFLX,NPSFLX)                                         NKM  953
      NMV = NMX + 1                                                     NKM  954
      NMX = NMX + 3                                                     NKM  955
      DO 662 I = NMV,NMX                                                NKM  956
      IF(I.EQ.8) K=2                                                    NKM  957
      IF(I.EQ.9) K=7                                                    NKM  958
      IF(I.EQ.10) K=9                                                   NKM  959
      WORK(I) = SPEC(K)                                                 NKM  960
      WORK(I+10) = FLXWAL(K)*TDELZ(1)                                   NKM  961
      WORK(I+20) = SPEC(K)                                              NKM  962
      WORK(I+30) = PSRATE(K,1)                                          NKM  963
662   CONTINUE                                                          NKM  964
      WRITE(LOUT,92)                                                    NKM  965
      WRITE(LOUT,93)  (WORK(I),WORK(I+10)                               NKM  966
     1 ,              WORK(I+20), WORK(I+30),I=1,NMX)                   NKM  967
      WRITE(LOUT,94)                                                    NKM  968
      WRITE(LOUT,9) (SPEC(J),J=1,NOSPEC)                                NKM  969
      DO 690 K = 1,NOSTAT                                               NKM  970
      WRITE(LOUT,32) ZEE(K) , (PPM(J,K),J=1,NOSPEC)                     NKM  971
  690 CONTINUE                                                          NKM  972
      IF (TIMOUT.NE.YES)  GO TO 691                                     NKM  973
      WRITE(LOUT,66)  INDEX,HUSED,NQUSED,NSTEP,NFE,NJE                  NKM  974
      NTSTP = NSTEP                                                     NKM  975
      CALL TIMEX(NTSTP,A,B)                                             NKM  976
  691 CONTINUE                                                          NKM  977
```

```
      IF(SPUNCH.NE.YES)  GO TO 694                                      NKM  978
C     ------  WRITE OUTPUT ON TAPE LUP FOR PUNCH OR FILE STORAGE        NKM  979
      WRITE(LUP,81) TIME                                                NKM  980
      DO 693 K=1,NOELV                                                  NKM  981
      WRITE(LUP,81) (PPM(J,K),J=1,NPUNCH)                               NKM  982
  693 CONTINUE                                                          NKM  983
  694 CONTINUE                                                          NKM  984
C                                                                       NKM  985
      IF(NCURV.EQ.0) GO TO 700                                          NKM  986
      KT = KT + 1                                                       NKM  987
      IF(KT.GT.KTMAX) GO TO 700                                         NKM  988
      DO 695 I = 1,NCURV                                                NKM  989
      J = KPS(I)                                                        NKM  990
      IF(J.NE.5)  GO TO 696                                             NKM  991
C     SCALE CO FOR PLOTTING                                             NKM  992
      GRCON(I,KT) = .10*PPM(J,1)                                        NKM  993
      IF(GRCON(I,KT).GT.VALMAX(I)) VALMAX(I)=GRCON(I,KT)                NKM  994
      GO TO 695                                                         NKM  995
696   CONTINUE                                                          NKM  996
      GRCON(I,KT) = PPM(J,1)                                            NKM  997
      IF(PPM(J,1).GT.VALMAX(I)) VALMAX(I) = PPM(J,1)                    NKM  998
  695 CONTINUE                                                          NKM  999
      TOUT(KT) = TIME                                                   NKM 1000
  700 CONTINUE                                                          NKM 1001
      IF(TIME.GE.(TSTOP-.01)) ISTOP = IES                               NKM 1002
      IF(ISTOP.EQ. IES)  GO TO 800                                      NKM 1003
C                                                                       NKM 1004
C     ------------ UPDATE EMISSION FLUXES                               NKM 1005
      IFLXTM = IFLXTM+1                                                 NKM 1006
      IF (IFLXTM.GT.NFLUX) IFLXTM = NFLUX                               NKM 1007
      CALL UPFLX1(TIME,IFLXTM)                                          NKM 1008
      IPS = IPS + 1                                                     NKM 1009
      CALL UPSORC(TIME,IPS,NOSTAT,SPEC)                                 NKM 1010
720   CONTINUE                                                          NKM 1011
C                                                                       NKM 1012
C     ------- UPDATE SKY COVER FACTOR                                   NKM 1013
      IF (QRATE.EQ.RNEG) GO TO 730                                      NKM 1014
      IF(QCLOUD.EQ.RNEG)  GO TO 725                                     NKM 1015
      IF(ICLOUD.GE.NCLOUD)  GO TO 725                                   NKM 1016
      BB= CLOUDT(ICLOUD+1)  - UPD4                                      NKM 1017
      IF(TIME.LT. BB)  GO TO 725                                        NKM 1018
      ICLOUD = ICLOUD + 1                                               NKM 1019
      CLOUDY = CLOUDF(ICLOUD)                                           NKM 1020
      WRITE(10,75)  TIME ,  CLOUDY                                      NKM 1021
725   CONTINUE                                                          NKM 1022
C                                                                       NKM 1023
C     -----------  UPDATE THE VARIABLE PHOTODISSOCIATION RATE CONSTANTS NKM 1024
C                                                                       NKM 1025
      ROLDK1 = RATEV(1,1)                                               NKM 1026
      CALL PHOTOR(TIME,NOSTAT,ZEE,CLOUDY,RATEV)                         NKM 1027
      IF(RATEV(1,1).LT..2*ROLDK1.OR.RATEV(1,1).GT.5.*ROLDK1)RECALL=YES  NKM 1028
730   CONTINUE                                                          NKM 1029
C                                                                       NKM 1030
C     ----          UPDATE DIFFUSION COEFFICIENTS                       NKM 1031
C                                                                       NKM 1032
      IF(NHT.EQ.1)  GO TO 770                                           NKM 1033
      AA= HTINV(IHT,1) - UPD4                                           NKM 1034
      IF(TIME.LT.AA)  GO TO 770                                         NKM 1035
```

```
      IF(KHT.GE.NHT)  GO TO 770                                          NKM 1036
      IHT = IHT + 1                                                      NKM 1037
      KHT = KHT + 1                                                      NKM 1038
      WRITE(11,46) TIME                                                  NKM 1039
      K = NOSTAT+1                                                       NKM 1040
      DO 750 I = 1,K                                                     NKM 1041
      DFINIT(I) = HTINV(IHT,I+2)                                         NKM 1042
750   CONTINUE                                                           NKM 1043
      WRITE(11,96) (DFINIT(I),I=1,K)                                     NKM 1044
      CALL DIFCOF(NOSTAT)                                                NKM 1045
      IF (IHT.GT.NHT) IHT = NHT                                          NKM 1046
770   CONTINUE                                                           NKM 1047
C                                                                        NKM 1048
C      -----  UPDATE TEMPERATURE DEPENDENT REACTION RATES                NKM 1049
      IF(VTEMP.NE.YES)  GO TO 780                                        NKM 1050
      IF(ITEMP.GE.NTEMP)  GO TO 780                                      NKM 1051
      IP1 = ITEMP + 1                                                    NKM 1052
      88 = TMTEMP(IP1) - UPD4                                            NKM 1053
      IF(TIME.LT.88)  GO TO 780                                          NKM 1054
      ITEMP = IP1                                                        NKM 1055
      CALL TEMPR(ITEMP,TIME,TEMP)                                        NKM 1056
780   CONTINUE                                                           NKM 1057
C                                                                        NKM 1058
      GO TO 460                                                          NKM 1059
C                                                                        NKM 1060
C     ------------  EXIT FROM INTEGRATION TIME CYCLE -----------------  NKM 1061
C                                                                        NKM 1062
800   CONTINUE                                                           NKM 1063
      IF(F.EQ.YES)  RETURN                                               NKM 1064
      IF(INDEX.LT.0)  WRITE(LOUT,74) TIME, INDEX                         NKM 1065
      IF(NCURV.EQ.0) GO TO 805                                           NKM 1066
      IF(KT.GT.KTMAX) KT = KTMAX                                         NKM 1067
      CALL COPLOT(KT)                                                    NKM 1068
805   CONTINUE                                                           NKM 1069
      IF(NSTEP .GT. NUMSTP)  WRITE(LOUT,26)                              NKM 1070
      RETURN                                                             NKM 1071
C                                                                        NKM 1072
C     FORMATS                                                            NKM 1073
C                                                                        NKM 1074
8     FORMAT (1H ,5X,A4,8H FLUX = ,2X,G15.8)                             NKM 1075
9     FORMAT (1H0,4X,9HHEIGHT-M.,3X,A4,7(11X,A4)/,6X,8(11X,A4),/,        NKM 1076
     1  6X,8(11X,A4),/,6X,8(11X,A4),/,6X,8(11X,A4),/,6X,8(11X,A4),/   ) NKM 1077
13    FORMAT (1H0,  /,  34X,48HCONCENTRATION PROFILES-PARTS PER MILLION  NKM 1078
     1        //45X,5HTIME=,E12.5,8H MINUTES//10X,20HGROUND DIFFUSIVITY NKM 1079
     2=,G15.8,10X, 1HK,I1,3H = ,G12.4,10X,1HK,I2,3H = ,G12.4)           NKM 1080
14    FORMAT (1H0,10X,30HINITIAL DIFFUSION COEFFICIENTS//10X,52HVALUES SNKM 1081
     1HOWN ARE FOR POINTS HALFWAY BETWEEN STATIONS///10X,7HSTATION,10X,1NKM 1082
     25HHEIGHT (METERS),10X,11HDIFFUSIVITY,16X,7HDELTA Z,11X,11HCELL HEINKM 1083
     3GHT  )                                                            NKM 1084
15    FORMAT (1H1,10X,84HNUMBER OF NO2 RATE CONSTANTS EXCEEDED THE DIMENNKM 1085
     1SIONS OF VECTOR -RATK1-, JOB ABORTED.)                            NKM 1086
16    FORMAT (1H0,//,55H  INPUT DATA FOR VERTICAL DIFFUSION WITH PHOTOCHNKM 1087
     1EMISTRY  ///8X,15HINITIAL TIME IS,F25.2,8H MINUTES//8X,           NKM 1088
     1   13HFINAL TIME IS      ,                                        NKM 1089
     1      F27.2,8H MINUTES//8X,25HINITIAL TIME STEP SIZE IS,F15.6,8H MNKM 1090
     3INUTES   //,8X,25HMAXIMUM TIME STEP SIZE IS ,F15.2,              NKM 1091
     3 8H MINUTES  ,//,8X,22HNUMBER OF REACTIONS IS ,I10,               NKM 1092
     4    //8X,20HNUMBER OF SPECIES IS ,I20,//,8X, 35HNUMBER OF SPECIES NKM 1093
```

```
     5IN STEADY STATE IS,I4,                                                  NKM 1094
     5              //8X,33HNUMBER OF VERTICAL MESH POINTS IS,I7,4X,33HNKM 1095
     7INCLUDING THE GROUND AND THE EDGE )                                     NKM 1096
17    FORMAT (1H0,//13X,8(8X,A4),6X,13HRATE CONSTANT,/,13X,8(8X,A4),         NKM 1097
     1 /,13X,8(8X,A4),/,13X,8(8X,A4),/,13X,8(8X,A4) )                         NKM 1098
18    FORMAT(1H0,                                                            NKM 1099
     1/////4X,9HHEIGHT-M.,5X,A4,7(11X,A4)/,7X,8(11X,A4),/,7X,8(11X,A4),     NKM 1100
     2  /,7X,8(11X,A4),/,7X,8(11X,A4)  )                                     NKM 1101
20    FORMAT(1H0,5X,I3,2X,8F12.2,10X,G12.5)                                  NKM 1102
21    FORMAT (1H ,10X,8F12.2,/, 3(11X,8F12.2,/)  )                           NKM 1103
22    FORMAT(50H1  THE READ PHOTOLYTIC RATES OPTION WAS DELETED              )NKM 1104
26    FORMAT (1H1,25(1H /),10X,36HNUMBER OF INTEGRATION STEPS EXCEEDED,2NKM 1105
     14H LIMIT.  JOB TERMINATED.)                                             NKM 1106
27    FORMAT(1H ,I3,3H.  ,A4,14X,F7.1,16X,I2)                                NKM 1107
28    FORMAT(1H0,/6X,7HSPECIES,7X,16HMOLECULAR WEIGHT,5X,9HBOUNDARY          NKM 1108
     1  11HCONDITION    ,/)                                                   NKM 1109
29    FORMAT (40X,A4)                                                        NKM 1110
30    FORMAT (40X,F10.0)                                                     NKM 1111
31    FORMAT (40X,I10)                                                       NKM 1112
32    FORMAT (1H ,2X,F7.2,8E15.5,/, 4(10X,8E15.5,/) )                        NKM 1113
33    FORMAT (1H1,25(/),25X,20A4//25X,8HDATE -- ,A10,12X,10HCP TIME = ,FNKM 1114
     110.2,5H  SEC)                                                           NKM 1115
37    FORMAT(1H0,    20X,17HR E A C T I O N S ,55X,30HRATE CONSTANTS (1/NKM 1116
     1(PPM--MIN))  /)                                                         NKM 1117
38    FORMAT(1H ,3X,I3,2H.  ,20A4,8X,E12.4)                                  NKM 1118
39    FORMAT (11X,A4,5X,11HAT STATION ,I3,5X,25HNEGATIVE OR ZERO AT TIMENKM 1119
     1 ,E10.3,5X,18HCONCENTRATION IS  ,E10.3,5X,8HDELT IS ,E10.3)             NKM 1120
40    FORMAT (20A4)                                                          NKM 1121
41    FORMAT (20X,5E10.4)                                                    NKM 1122
42    FORMAT(1H0,10X,I5,15X,F10.2,55X,F10.2)                                 NKM 1123
43    FORMAT(1H0,50X,G15.8,10X,F10.2)                                        NKM 1124
44    FORMAT(1H0,10X,I5,15X,F10.2,10X,G15.8,30X,F10.2)                       NKM 1125
45    FORMAT (1H1,10X,57HTOO MANY INVERSION HEIGHT ENTRIES IN TABLE.  JONKM 1126
     1B ABORTED.)                                                             NKM 1127
46    FORMAT(13H KZ AT TIME    ,F10.2)                                       NKM 1128
47    FORMAT(1H0,10X,I5,15X,F10.2,10X,G15.8)                                 NKM 1129
51    FORMAT (40X,A4,6X,F10.2,I10)                                           NKM 1130
52    FORMAT(20X,5(I2,8X)/20X,5(I2,8X))                                      NKM 1131
53    FORMAT (40X,I10,2F10.0)                                                NKM 1132
54    FORMAT (20X,5E12.4)                                                    NKM 1133
55    FORMAT (20X,10F6.2)                                                    NKM 1134
57    FORMAT (1H1,10X,45HTOO MANY CLOUD COVER ENTRIES  --  JOB ABORTED) NKM 1135
58    FORMAT(10(/),10X,73HDEPLETION OF  NO  HAS CAUSED FAILURE OF CHEMICNKM 1136
     1AL MODEL.  CASE TERMINATED. )                                           NKM 1137
59    FORMAT (40X,2F10.0)                                                    NKM 1138
60    FORMAT(1H1,/////,20X,41HJOB TERMINATED -- NEGATIVE CONCENTRATIONS NKM 1139
     1   30H OR DEPLETION OF NITRIC OXIDE       )                           NKM 1140
66    FORMAT(1H0,9X,20HINTEGRATION INDEX = ,I4,10X,18HLAST STEP SIZE =  NKM 1141
     1   E10.3,20X,17HLAST ORDER USED = ,I3,/,10X,18HNUMBER OF STEPS = ,NKM 1142
     2 I6,10X,33HNUMBER OF FUNCTION EVALUATIONS = ,I4,10X,34H NUMBER OF NKM 1143
     3JACOBIAN EVALUATIONS = ,I4)                                             NKM 1144
68    FORMAT(1H0, 5X,70HREVISED AREA SOURCE FLUX SCHEDULE      ---       NKM 1145
     1    PPM-METERS/MINUTE  ,//,6X,4HTIME,7(11X,A4),/,10X, 3(11X,A4))  NKM 1146
69    FORMAT(1H ,F10.1,1X,7E15.4)                                            NKM 1147
74    FORMAT (1H0,//,8H  TIME =, F8.2,5X,9HINDEX =   ,I2,                    NKM 1148
     1  36H  INTEGRATION TERMINATED BY EPISODE   )                           NKM 1149
75    FORMAT (1H0,10X,33HCLEARNESS RATIO UPDATED AT TIME =   ,F8.2,5X,  NKM 1150
     1    20HSKY COVER FACTOR  =   , F8.4)                                   NKM 1151
```

```
76    FORMAT(1H1,9X,41HJOB ABORTED BECAUSE OF TOO MANY SPECIES  , //     NKM 1152
     1/,9H NOSPEC =  ,I3,5X,9H  NSTDY=   ,I3,5X,5H NK=   ,I3)           NKM 1153
77    FORMAT(1H1,58HJOB ABORTED BECAUSE OF TOO FEW VERTICAL STATIONS  ,/NKM 1154
     1//,10X,8HNOSTAT = , I2)                                           NKM 1155
79    FORMAT (47H0  TOO MANY TEMPERATURES INPUT -- JOB ABORTED       )   NKM 1156
80    FORMAT (20A4)                                                      NKM 1157
81    FORMAT ( 7(6E12.4,/))                                              NKM 1158
83    FORMAT (1H0,5X,50HPOINT SOURCE EMISSION RATES BY VERTICAL CELL     NKM 1159
     1    10X,10H(PPM/MIN)  ,//,6X,4HTIME,2X,8(8X,A4))                  NKM 1160
84    FORMAT (1H0,F9.1,4X,8E12.3)                                        NKM 1161
85    FORMAT (1H ,13X,8E12.3)                                            NKM 1162
88    FORMAT(1H ,5X,A4,8H SINK = ,2X,G15.8)                              NKM 1163
91    FORMAT(1H0,40X,6HTIME =,F8.2,14H MINUTES     ( ,  I4,              NKM 1164
     1 17H ON 2400-CLOCK)   /,40X,24HSURFACE CELL PARAMETERS            NKM 1165
     2/ 40X,30HNO2 PHOTODISSOCIATION RATE  =   ,F8.3,10H   (/MIN)  ,/,  NKM 1166
     3 40X,38HDIFFUSIVITY COEFFICIENT      =   ,F8.1,13H   (M**2/MIN))  NKM 1167
92    FORMAT(1H ,39X,12HAREA SOURCES,2X,9H(PPM/MIN)3X,13HPOINT SOURCES   NKM 1168
     1  2X,9H(PPM/MIN) )                                                NKM 1169
93    FORMAT(  10(/,43X,A4,3H = ,E11.3,8X,A4,3H = ,E11.3))               NKM 1170
94    FORMAT(      40X,43HCONCENTRATION PROFILES IN PARTS PER MILLION )  NKM 1171
96    FORMAT(1H ,10X,6F15.2)                                             NKM 1172
97    FORMAT(1H0,/,40X,6HTIME =,F8.2,14H MINUTES     ( ,  I4,            NKM 1173
     1 17H ON 2400-CLOCK)   //,                                         NKM 1174
     1/,40X,51HINITIAL CONCENTRATION PROFILES IN PARTS PER MILLION/)     NKM 1175
99    FORMAT(11(//),57X,24HTHE ERT TRAJECTORY MODEL ,/57X,6(4H----),     NKM 1176
     1  //,59X,20H(CODE DATE   2.1.79) ////,63X,12HDEVELOPED BY  ,//     NKM 1177
     1  48X,41HENVIRONMENTAL RESEARCH   TECHNOLOGY, INC.  //             NKM 1178
     1  58X,22HSANTA BARBARA DIVISION   )                                NKM 1179
      END                                                               NKM 1180



      SUBROUTINE PHOTOR (TIME,NOSTAT,ZEE,CLOUDY,RATEV)                  NKM 1181
C                                                                       NKM 1182
C       PHOTOR CALCULATES PHOTODISSOCIATION RATES FOR THE  REACTIONS    NKM 1183
C                                                                       NKM 1184
C      1.      NO2 + HV   ----  NO + O                                  NKM 1185
C      2.      HONO + HV  ----  NO + OH                                 NKM 1186
C      3.      RCHO + HV  ----  RO2 + HO2 + CO                          NKM 1187
C      4.      HCHO + HV  -----  HO2 + CO                               NKM 1188
C                                                                       NKM 1189
C ---  INPUTS ----                                                      NKM 1190
C    TIME = TIME IN MINUTES FROM MIDNIGHT                               NKM 1191
C    NOSTAT = NUMBER OF VERTICAL MESH POINTS                            NKM 1192
C    ZEE = ELEVATION OF VERTICAL MESH POINTS ABOVE SURFACE (METERS)     NKM 1193
C    CLOUDY = SKY CLEARNESS RATIO                                       NKM 1194
C                                                                       NKM 1195
C --- OUTPUTS ---                                                       NKM 1196
C    RATEV = 2-D ARRAY OF RATE CONSTANTS                                NKM 1197
C                                                                       NKM 1198
      DIMENSION RATEV(4,1),ZEE(1)                                       NKM 1199
      DIMENSION ZINP(10),HTINP(8),R(8,10,4)                             NKM 1200
      DIMENSION RNO2(8,10),RHONO(8,10), RRCHO(8,10), RHCHO(8,10)        NKM 1201
      DIMENSION RMIN(4), REXP(4), NAM(4)                                NKM 1202
C                                                                       NKM 1203
      COMMON /CHEM4/  RLAT, RLONG, TMZONE, HIRATE, JDATE                NKM 1204
C                                                                       NKM 1205
      EQUIVALENCE( RNO2(1,1),R(1,1,1))                                  NKM 1206
```

```
      EQUIVALENCE(RHONO(1,1),R(1,1,2))                                  NKM 1207
      EQUIVALENCE(RRCHO(1,1),R(1,1,3))                                  NKM 1208
      EQUIVALENCE(RHCHO(1,1),R(1,1,4))                                  NKM 1209
C                                                                       NKM 1210
      DATA RMIN /1.E-3, 1.E-4,1.E-6,1.E-5 /                             NKM 1211
      DATA REXP /1.0, 0.100, 0.0060, 0.0100/                            NKM 1212
      DATA NHTI,NZINP,NR /8,10,4/                                       NKM 1213
      DATA YES, NAM /3HYES,3HNO2,4HHONO,4HRCHO,4HHCHO/                  NKM 1214
      DATA ZINP /0.,10.,20.,30.,40.,50.,60.,70.,78.,86. /               NKM 1215
      DATA HTINP /0.,150.,360.,640.,980.,1380.,1840.,4210./             NKM 1216
C                                                                       NKM 1217
      DATA RNO2 /                                                       NKM 1218
     1  .534,.565,.594,.621,.648,.671,.691,.758,                       NKM 1219
     2 .529,.561,.589,.617,.644,.668,.688,.755,                       NKM 1220
     3 .516,.549,.579,.607,.633,.657,.678,.746,                       NKM 1221
     4 .493,.526,.557,.586,.614,.638,.660,.720,                       NKM 1222
     5 .457,.492,.523,.553,.582,.607,.629,.703,                       NKM 1223
     6 .404,.439,.472,.503,.531,.556,.579,.657,                       NKM 1224
     7 .325,.360,.392,.423,.451,.476,.499,.581,                       NKM 1225
     8 .214,.243,.272,.300,.326,.348,.369,.451,                       NKM 1226
     9 .105,.123,.141,.160,.179,.196,.211,.280,                       NKM 1227
     1 .023,.025,.028,.031,.034,.037,.040,.050/                       NKM 1228
      DATA  RHONO/                                                      NKM 1229
     1.958,1.02,1.08,1.13,1.18,1.23,1.27,1.40,                        NKM 1230
     2.950,1.01,1.07,1.12,1.17,1.22,1.26,1.40,                        NKM 1231
     3.925,.987,1.05,1.10,1.15,1.20,1.24,1.38,                        NKM 1232
     4.881,.945,1.00,1.06,1.12,1.16,1.21,1.35,                        NKM 1233
     5.812,.877,.938,.996,1.05,1.10,1.45,1.29,                        NKM 1234
     6.710,.777,.837,.895,.951,1.00,1.04,1.20,                        NKM 1235
     7.563,.626,.685,.741,.795,.843,.886,1.05,                        NKM 1236
     8.361,.411,.460,.508,.554,.596,.634,.794,                        NKM 1237
     9.174,.200,.229,.260,.289,.317,.344,.468,                        NKM 1238
     1.039,.043,.048,.052,.057,.061,.656,.678/                        NKM 1239
      DATA  RRCHO /                                                     NKM 1240
     1.230,.249,.267,.286,.306,.324,.341,.397,                        NKM 1241
     2.227,.246,.264,.283,.301,.319,.367,.393,                        NKM 1242
     3.216,.234,.252,.270,.289,.307,.323,.379,                        NKM 1243
     4.196,.214,.231,.249,.267,.284,.300,.355,                        NKM 1244
     5.169,.186,.202,.218,.235,.251,.266,.320,                        NKM 1245
     6.134,.147,.162,.177,.191,.206,.219,.269,                        NKM 1246
     7.092,.103,.114,.125,.136,.147,.159,.202,                        NKM 1247
     8.047,.054,.060,.067,.074,.081,.088,.118,                        NKM 1248
     9.019,.021,.023,.026,.029,.032,.035,.050,                        NKM 1249
     1.003,.004,.004,.004,.005,.005,.005,.006/                        NKM 1250
      DATA  RHCHO /                                                     NKM 1251
     X .614,.662,.709,.756,.802,.846,.886,1.02,                       NKM 1252
     1 .606,.654,.700,.747,.794,.838,.878,1.01,                       NKM 1253
     2 .581,.629,.675,.721,.767,.810,.850,.985,                       NKM 1254
     3 .538,.584,.629,.675,.719,.762,.801,.937,                       NKM 1255
     4 .474,.519,.562,.605,.648,.689,.726,.861,                       NKM 1256
     5 .388,.428,.469,.508,.547,.585,.620,.750,                       NKM 1257
     6 .280,.313,.346,.379,.412,.444,.474,.594,                       NKM 1258
     7 .156,.178,.199,.222,.244,.266,.288,.361,                       NKM 1259
     7 .067,.076,.086,.096,.107,.117,.128,.181,                       NKM 1260
     8 .013,.015,.016,.018,.019,.021,.022,.029/                       NKM 1261
C                                                                       NKM 1262
      STIME = ITHOUR(TIME)                                              NKM 1263
      IY = JDATE/10000                                                  NKM 1264
```

```
      IM = JDATE/100 - IY*100                                          NKM 1265
      ID = JDATE - IY*10000 - IM*100                                   NKM 1266
      IY = IY + 1900                                                   NKM 1267
C     RETRIEVE SOLAR ZENITH ANGLE ZEN                                  NKM 1268
      CALL SOLAR(RLAT,RLONG,TMZONE,IY,IM,ID,STIME,O,5)                 NKM 1269
      ZEN = 90. - O                                                    NKM 1270
      IF((ZEN.GE.ZINP(1)) .AND. (ZEN.LE.ZINP(NZINP))) GO TO 40         NKM 1271
C                                                                      NKM 1272
      DO 20 K = 1,NOSTAT                                               NKM 1273
      DO 20 J = 1,NR                                                   NKM 1274
      RATEV(J,K) = RMIN(J)                                             NKM 1275
20    CONTINUE                                                         NKM 1276
      GO TO 200                                                        NKM 1277
C                                                                      NKM 1278
40    DO 60 I = 1,NZINP                                                NKM 1279
      IF(ZINP(I).LT.ZEN)  GO TO 60                                     NKM 1280
      I1  = I - 1                                                      NKM 1281
      I2 = I                                                           NKM 1282
      Z1 = ZINP(I1)                                                    NKM 1283
      Z2 = ZINP(I2)                                                    NKM 1284
      GO TO 80                                                         NKM 1285
60    CONTINUE                                                         NKM 1286
80    PZ = (ZEN-Z1)/(Z2-Z1)                                            NKM 1287
      J = 1                                                            NKM 1288
      DO 90 KJ = 1,NHTI                                                NKM 1289
      IF(ZEE(JJ+1).GT.HTINP(J+1))  J = J + 1                           NKM 1290
90    CONTINUE                                                         NKM 1291
C                                                                      NKM 1292
      DO 140 JJ = 1,NOSTAT                                             NKM 1293
      JP = J + 1                                                       NKM 1294
      PH = (ZEE(JJ)-HTINP(J))/(HTINP(JP)-HTINP(J))                     NKM 1295
      DO 100 K = 1,NR                                                  NKM 1296
      R1 = R(J,I1,K) +PZ*(R(J,I2,K)-R(J,I1,K))                         NKM 1297
      R2 = R(JP,I1,K) + PZ*(R(JP,I2,K)-R(JP,I1,K))                     NKM 1298
C                                                                      NKM 1299
      RATEV(K,JJ) = (R1 + PH*(R2-R1))*REXP(K)*CLOUDY                   NKM 1300
C                                                                      NKM 1301
100   CONTINUE                                                         NKM 1302
      IF(HIRATE.NE.YES)  GO TO 160                                     NKM 1303
      IF(JJ.EQ.NOSTAT)  GO TO 140                                      NKM 1304
      KK = J                                                           NKM 1305
      DO 120 KJ = KK,NHTI                                              NKM 1306
      IF(ZEE(JJ+1).GT.HTINP(J+1))  J = J + 1                           NKM 1307
120   CONTINUE                                                         NKM 1308
      J = MIN0(J,NHTI)                                                 NKM 1309
140   CONTINUE                                                         NKM 1310
      GO TO 200                                                        NKM 1311
160   DO 180 K = 2,NOSTAT                                              NKM 1312
      DO 180 J = 1,NR                                                  NKM 1313
      RATEV(J,K) = RATEV(J,1)                                          NKM 1314
180   CONTINUE                                                         NKM 1315
200   WRITE(10,2)  TIME,ZEN                                            NKM 1316
      DO 220 J= 1,NR                                                   NKM 1317
      WRITE  (10,1)       NAM(J),(RATEV(J,K),K=1,NOSTAT)               NKM 1318
220   CONTINUE                                                         NKM 1319
      RETURN                                                           NKM 1320
1     FORMAT(1H ,A4,8E10.3)                                            NKM 1321
2     FORMAT( 30H  PHOTOLYSIS RATES AT TIME =  ,F7.1,5X,               NKM 1322
```

```
     1  15HZENITH ANGLE =      ,F7.2)                                        NKM 1323
        END                                                                  NKM 1324



        SUBROUTINE PRODUK(DT)                                                NKM 1325
C                                                                            NKM 1326
C          COMPUTES APPROXIMATE PRODUCT SPECIES CONCENTRATIONS               NKM 1327
C                                                                            NKM 1328
C          HNO3  OZID  H2O2  NTRA  ANO2  ACHO                                NKM 1329
C                                                                            NKM 1330
C          THIS ROUTINE IS SPECIFIC TO THE ERT PHOTOCHEMICAL MECHAMISM       NKM 1331
C     39 SPECIES  X  64 REACTIONS     11.1.78                                NKM 1332
        COMMON/CHEM1/ NOSTAT,            NOSTM1,          NOREAC,           NKM 1333
     1                NOSPEC,           NSTDY,           NK                NKM 1334
        COMMON/CHEM2/ CONIN(40,5),       WTMOLE(40),      RATKON(65),       NKM 1335
     1                RATEFF(65),       RATEV(4,5),      QRATE,            NKM 1336
     2                NVRATE,           LOCVRT(4)                          NKM 1337
        DO 10 K = 1,NOSTAT                                                   NKM 1338
C          ----------  HNO3    ---------                                     NKM 1339
        RATE = RATEFF(8)*CONIN(2,K)*CONIN(31,K)  +  RATEFF(16)*CONIN(21,K)   NKM 1340
     1 *CONIN(33,K)                                                          NKM 1341
        CONIN(34,K) = CONIN(34,K) + DT*RATE                                  NKM 1342
C          ----------  OZID  ----------                                      NKM 1343
        RATE = RATEFF(31)*CONIN(6,K)*CONIN(25,K) + RATEFF(32)*CONIN(7,K)*    NKM 1344
     1  CONIN(25,K) + RATEFF(33)*CONIN(6,K)*CONIN(26,K) + RATEFF(34)*       NKM 1345
     2  CONIN(7,K)*CONIN(26,K)                                              NKM 1346
        CONIN(35,K) = CONIN(35,K) + RATE*DT                                  NKM 1347
C          ----------  H2O2  ----------                                      NKM 1348
        RATE = (CONIN(22,K)**2)*RATEFF(12) + CONIN(22,K)*CONIN(13,K)*        NKM 1349
     1                                                  RATEFF(55)          NKM 1350
        CONIN(36,K) = CONIN(36,K) + RATE*DT                                  NKM 1351
C          ----------  NTRA  ----------                                      NKM 1352
        RATE = CONIN(1,K)*CONIN(28,K)*RATEFF(37) + CONIN(2,K)*CONIN(27,K)    NKM 1353
     1                                          *RATEFF(41)*.85             NKM 1354
        CONIN(37,K) = CONIN(37,K) + RATE*DT                                  NKM 1355
C          ----------  ANO2  ----------                                      NKM 1356
        RATE = CONIN(2,K)*CONIN(13,K)*RATEFF(54)                             NKM 1357
        CONIN(38,K) = CONIN(38,K) + RATE*DT                                  NKM 1358
C          ----------  ACHO  ----------                                      NKM 1359
        RATE = CONIN(1,K)*CONIN(16,K)*RATEFF(58)                             NKM 1360
        CONIN(39,K) = CONIN(39,K) + RATE*DT                                  NKM 1361
   10   CONTINUE                                                             NKM 1362
        RETURN                                                               NKM 1363
        END                                                                  NKM 1364



        SUBROUTINE RATEHI(K)                                                 NKM 1365
C                                                                            NKM 1366
C       RETRIEVES PHOTODISSOCIATION RATES FOR THE KTH MESH POINT             NKM 1367
C                                                                            NKM 1368
        COMMON /CHEM2/ CONIN(40,5),  WTMOLE(40),  RATKON(65),                NKM 1369
     1                 RATEFF(65),  RATEV(4,5),  QRATE, NVRATE, LOCVRT(4)   NKM 1370
C                                                                            NKM 1371
        DO 10 I = 1,NVRATE                                                   NKM 1372
        J = LOCVRT(I)                                                        NKM 1373
        RATKON(I) = RATEV(I,K)                                               NKM 1374
```

```
      RATEFF(J) = RATEV(I,K)                                                NKM 1375
10    CONTINUE                                                              NKM 1376
      RETURN                                                                NKM 1377
      END                                                                   NKM 1378



      SUBROUTINE RATES(Y,YDOT)                                              NKM 1379
C                                                                           NKM 1380
C          CALCULATION OF CHEMICAL RATES                                    NKM 1381
C                                                                           NKM 1382
      INTEGER BCFLAG                                                        NKM 1383
      COMMON/CHEM1/ NOSTAT,          NOSTM1,          NOREAC,               NKM 1384
     1              NOSPEC,          NSTOY,           NK                    NKM 1385
      COMMON/CHEM2/ CONIN(40,5),     HTMOLE(40),      RATKON(65),           NKM 1386
     1              RATEFF(65),      RATEV(4,5),      QRATE,                NKM 1387
     2              NVRATE,          LOCVRT(4)                              NKM 1388
      DIMENSION Y(1), YDOT(1)  ,  RATE(40),  C(40),  R(65)                  NKM 1389
      EQUIVALENCE (R,RATEFF)                                                NKM 1390
      DATA  YES  /3HYES/                                                    NKM 1391
      NKP1 = NK + 1                                                         NKM 1392
      DO 130 K = 1,NOSTAT                                                   NKM 1393
      KS = NK*(K-1)                                                         NKM 1394
      DO 90  I=1,NK                                                         NKM 1395
      C(I) = Y(I+KS)                                                        NKM 1396
  90  CONTINUE                                                              NKM 1397
      IF(NSTOY.EQ.0)  GO TO 105                                             NKM 1398
      DO 100 I=NKP1,NOSPEC                                                  NKM 1399
      C(I) = CONIN(I,K)                                                     NKM 1400
100   CONTINUE                                                              NKM 1401
105   CONTINUE                                                              NKM 1402
C                                                                           NKM 1403
      IF(QRATE.EQ.YES)  CALL RATEHI(K)                                      NKM 1404
      CALL UNMIXR(K)                                                        NKM 1405
C                                                                           NKM 1406
C      EXPLICIT CHEMICAL RATE EQUATIONS                                     NKM 1407
C                                                                           NKM 1408
C         RATES FOR THE ERT PHOTOCHEMICAL MECHANISM  (39X64)  1.31.79       NKM 1409
C                                                                           NKM 1410
      RATE(  1) = + R(  1)*C(  2) - R(  3)*C(  1)*C(  3) - R(  4)*C(  1)NKM 1411
     *     *C(  2)*C( 33) + R(  5)*C(  4)** 2 + R(  6)*C(  4) - R(  7)    NKM 1412
     *     *C(  1)*C( 31) - R( 10)*C(  1)*C( 22) - R( 14)*C(  1)*C( 19)   NKM 1413
     *      - R( 20)*C(  1)*C( 18) - R( 27)*C(  1)*C( 25) - R( 28)*C(  1)NKM 1414
     *     *C( 26) - R( 36)*C(  1)*C( 28) - R( 37)*C(  1)*C( 28) - R( 39)NKM 1415
     *     *C(  1)*C( 23) - R( 44)*C(  1)*C( 24) - R( 53)*C(  1)*C( 13)   NKM 1416
     *      - R( 58)*C(  1)*C( 16)                                         NKM 1417
      RATE(  2) = - R(  1)*C(  2) + R(  3)*C(  1)*C(  3) - R(  4)*C(  1)NKM 1418
     *     *C(  2)*C( 33) + R(  5)*C(  4)** 2 - R(  8)*C(  2)*C( 31) +    NKM 1419
     *     R( 10)*C(  1)*C( 22) - R( 11)*C(  2)*C( 22) - R( 13)*C(  2)    NKM 1420
     *     *C(  3) +  2.00*R( 14)*C(  1)*C( 19) - R( 15)*C(  2)*C( 19) +  NKM 1421
     *     R( 17)*C( 21) + R( 20)*C(  1)*C( 18) + R( 22)*C( 20) + R( 27)  NKM 1422
     *     *C(  1)*C( 25) + R( 28)*C(  1)*C( 26) - R( 29)*C(  2)*C( 25)   NKM 1423
     *      - R( 30)*C(  2)*C( 26) + R( 36)*C(  1)*C( 28) + R( 39)*C(  1)NKM 1424
     *     *C( 23) - R( 41)*C(  2)*C( 27) + R( 44)*C(  1)*C( 24) - R( 45)NKM 1425
     *     *C(  2)*C( 24) + R( 46)*C( 11) + R( 53)*C(  1)*C( 13) - R( 54)NKM 1426
     *     *C(  2)*C( 13) + R( 58)*C(  1)*C( 16)                          NKM 1427
      RATE(  3) = + R(  2)*C( 32) - R(  3)*C(  1)*C(  3) - R( 13)*C(  2)NKM 1428
     *     *C(  3) - R( 25)*C(  3)*C(  8) - R( 26)*C(  3)*C(  9)          NKM 1429
```

```
 RATE(  4) = +  2.00*R(  4)*C(  1)*C(  2)*C( 33) -  2.00*R(  5)     NKM 1430
*     *C(  4)** 2 - R(  6)*C(  4) + R(  7)*C(  1)*C( 31) +          NKM 1431
*      0.15*R( 41)*C(  2)*C( 27)                                    NKM 1432
 RATE(  5) = - R(  9)*C(  5)*C( 31) +  0.40*R( 24)*C(  9)*C( 32) + NKM 1433
*     R( 43)*C(  7) + R( 47)*C(  6) + R( 48)*C(  6)*C( 31)         NKM 1434
 RATE(  6) = +  1.50*R( 21)*C( 17) +  0.50*R( 25)*C(  3)*C(  8) + NKM 1435
*     R( 26)*C(  3)*C(  9) + R( 27)*C(  1)*C( 25) + R( 29)*C(  2)  NKM 1436
*     *C( 25) - R( 31)*C(  6)*C( 25) - R( 33)*C(  6)*C( 26) +      NKM 1437
*      0.50*R( 38)*C( 27) - R( 47)*C(  6) - R( 48)*C(  6)*C( 31) + NKM 1438
*     R( 61)*C( 25)*C( 29)                                         NKM 1439
 RATE(  7) = +  0.50*R( 21)*C( 17) +  0.30*R( 23)*C(  8)*C( 32) + NKM 1440
*      0.30*R( 24)*C(  9)*C( 32) +  0.50*R( 25)*C(  3)*C(  8) +    NKM 1441
*     R( 28)*C(  1)*C( 26) + R( 30)*C(  2)*C( 26) - R( 32)*C(  7)  NKM 1442
*     *C( 25) - R( 34)*C(  7)*C( 26) +  0.50*R( 38)*C( 27) + R( 40) NKM 1443
*     *C( 27) +  0.15*R( 41)*C(  2)*C( 27) - R( 42)*C(  7)*C( 31) - NKM 1444
*     R( 43)*C(  7) + R( 62)*C( 26)*C( 29)                         NKM 1445
 RATE(  8) = - R( 18)*C(  8)*C( 31) - R( 23)*C(  8)*C( 32) - R( 25)NKM 1446
*     *C(  3)*C(  8)                                               NKM 1447
 RATE(  9) = - R( 19)*C(  9)*C( 31) - R( 24)*C(  9)*C( 32) - R( 26)NKM 1448
*     *C(  3)*C(  9)                                               NKM 1449
 RATE( 10) = - R( 35)*C( 10)*C( 31)                                NKM 1450
 RATE( 11) = + R( 45)*C(  2)*C( 24) - R( 46)*C( 11)                NKM 1451
 RATE( 12) = - R( 50)*C( 12)*C( 31) - R( 51)*C( 12)*C( 31) - R( 57)NKM 1452
*     *C( 12)*C( 31)                                               NKM 1453
 RATE( 13) = + R( 51)*C( 12)*C( 31) - R( 53)*C(  1)*C( 13) - R( 54)NKM 1454
*     *C(  2)*C( 13) - R( 55)*C( 13)*C( 22)                        NKM 1455
 RATE( 14) = + R( 50)*C( 12)*C( 31) - R( 52)*C( 14)*C( 31) + R( 55)NKM 1456
*     *C( 13)*C( 22)                                               NKM 1457
 RATE( 15) = + R( 53)*C(  1)*C( 13) - R( 56)*C( 15)                NKM 1458
 RATE( 16) = + R( 57)*C( 12)*C( 31) - R( 58)*C(  1)*C( 16)         NKM 1459
 RATE( 17) = + R( 20)*C(  1)*C( 18) - R( 21)*C( 17) + R( 64)*C( 18)NKM 1460
*     *C( 29)                                                      NKM 1461
 RATE( 18) = + R( 18)*C(  8)*C( 31) + R( 19)*C(  9)*C( 31) - R( 20)NKM 1462
*     *C(  1)*C( 18) - R( 64)*C( 18)*C( 29)                        NKM 1463
 RATE( 19) = + R( 13)*C(  2)*C(  3) - R( 14)*C(  1)*C( 19) - R( 15)NKM 1464
*     *C(  2)*C( 19) + R( 17)*C( 21) + R( 29)*C(  2)*C( 25) + R( 30)NKM 1465
*     *C(  2)*C( 26)                                               NKM 1466
 RATE( 20) = + R( 11)*C(  2)*C( 22) - R( 22)*C( 20)                NKM 1467
 RATE( 21) = + R( 15)*C(  2)*C( 19) - R( 16)*C( 21)*C( 33) - R( 17)NKM 1468
*     *C( 21)                                                      NKM 1469
 RATE( 22) = + R(  9)*C(  5)*C( 31) - R( 10)*C(  1)*C( 22) - R( 11)NKM 1470
*     *C(  2)*C( 22) -  2.00*R( 12)*C( 22)** 2 + R( 21)*C( 17) +   NKM 1471
*     R( 22)*C( 20) +  0.40*R( 23)*C(  8)*C( 32) +  0.40*R( 24)    NKM 1472
*     *C(  9)*C( 32) +  0.40*R( 26)*C(  3)*C(  9) + R( 40)*C( 27) + NKM 1473
*     R( 43)*C(  7) +       R( 47)*C(  6) + R( 48)*C(  6)*C( 31) - NKM 1474
*     R( 49)*C( 22)*C( 23) + R( 50)*C( 12)*C( 31) + R( 52)*C( 14)  NKM 1475
*     *C( 31) - R( 55)*C( 13)*C( 22) + R( 58)*C(  1)*C( 16) - R( 60)NKM 1476
*     *C( 22)*C( 29)                                               NKM 1477
 RATE( 23) = +  0.40*R( 23)*C(  8)*C( 32) +  0.40*R( 24)*C(  9)    NKM 1478
*     *C( 32) +  0.10*R( 25)*C(  3)*C(  8) +  0.15*R( 36)*C(  1)   NKM 1479
*     *C( 28) + R( 38)*C( 27) - R( 39)*C(  1)*C( 23) + R( 43)*C(  7)NKM 1480
*      + R( 44)*C(  1)*C( 24) - R( 49)*C( 22)*C( 23) + R( 56)*C( 15)NKM 1481
*      - R( 63)*C( 23)*C( 29)                                      NKM 1482
 RATE( 24) = + R( 42)*C(  7)*C( 31) - R( 44)*C(  1)*C( 24) - R( 45)NKM 1483
*     *C(  2)*C( 24) + R( 46)*C( 11)                               NKM 1484
 RATE( 25) = +  0.40*R( 25)*C(  3)*C(  8) +  0.80*R( 26)*C(  3)    NKM 1485
*     *C(  9) - R( 27)*C(  1)*C( 25) - R( 29)*C(  2)*C( 25) - R( 31)NKM 1486
*     *C(  6)*C( 25) - R( 32)*C(  7)*C( 25) - R( 61)*C( 25)*C( 29)  NKM 1487
```

```
      RATE( 26) = +  0.40*R( 25)*C(  3)*C(  8) - R( 28)*C(  1)*C( 26) - NKM 1488
     *    R( 30)*C(  2)*C( 26) - R( 33)*C(  6)*C( 26) - R( 34)*C(  7)   NKM 1489
     *    *C( 26) - R( 62)*C( 26)*C( 29)                                NKM 1490
      RATE( 27) = +  0.85*R( 36)*C(  1)*C( 28) - R( 38)*C( 27) + R( 39) NKM 1491
     *    *C(  1)*C( 23) - R( 48)*C( 27) - R( 41)*C(  2)*C( 27) + R( 63)NKM 1492
     *    *C( 23)*C( 29)                                                NKM 1493
      RATE( 28) = + R( 35)*C( 10)*C( 31) - R( 36)*C(  1)*C( 28) - R( 37)NKM 1494
     *    *C(  1)*C( 28)                                                NKM 1495
C                                                                        NKM 1496
      IF(NK.LE.28)  GO TO 110                                           NKM 1497
C                                                                        NKM 1498
C         RATES FOR SO2 AND SO4                                          NKM 1499
C                                                                        NKM 1500
      RATE( 29) = - R( 59)*C( 29)*C( 31) - R( 60)*C( 22)*C( 29) - R( 61)NKM 1501
     *    *C( 25)*C( 29) - R( 62)*C( 26)*C( 29) - R( 63)*C( 23)*C( 29)  NKM 1502
     *     - R( 64)*C( 18)*C( 29)                                        NKM 1503
      RATE( 30) = + R( 59)*C( 29)*C( 31) + R( 60)*C( 22)*C( 29) + R( 61)NKM 1504
     *    *C( 25)*C( 29) + R( 62)*C( 26)*C( 29) + R( 63)*C( 23)*C( 29)  NKM 1505
     *     + R( 64)*C( 18)*C( 29)                                        NKM 1506
C                                                                        NKM 1507
110   CONTINUE                                                           NKM 1508
      DO 120 J = 1,NK                                                    NKM 1509
      YDOT(J+KS) = RATE(J)                                               NKM 1510
120   CONTINUE                                                           NKM 1511
C                                                                        NKM 1512
130   CONTINUE                                                           NKM 1513
      RETURN                                                             NKM 1514
      END                                                                NKM 1515
```

```
      SUBROUTINE STEADY(Y,N)                                             NKM 1516
C                                                                        NKM 1517
C     THIS VERSION OF STEADY COMPUTES THE STEADY STATE CONCENTRATIONS     NKM 1518
C     OF O AND OH FOR THE ERT PHOTOCHEMICAL MECHANISM   (39X64) 1.31.79 NKM 1519
C                                                                        NKM 1520
      COMMON /CHEM1/ NOSTAT, NOSTM1, NOREAC, NOSPEC, NSTDY, NK           NKM 1521
      COMMON/CHEM2/ CONIN(40,5),        WTMOLE(40),         RATKON(65), NKM 1522
     1              R(65) ,             RATEV(4,5),         QRATE,      NKM 1523
     2              NVRATE,             LOCVRT(4)                       NKM 1524
      DIMENSION Y(1)                                                     NKM 1525
      DATA YES /3HYES /                                                  NKM 1526
C                                                                        NKM 1527
      DO 50 K = 1,NOSTAT                                                 NKM 1528
      IF(QRATE.EQ.YES)  CALL RATEMI(K)                                   NKM 1529
      J = (K-1)*NK                                                       NKM 1530
C     --------   O   ---------                                           NKM 1531
      FORM = R(1)*Y(2+J)                                                 NKM 1532
      DSTROY = R(2) + R(23)*Y(8+J) + R(24)*Y(9+J)                        NKM 1533
      CONIN(32,K) = FORM/DSTROY                                          NKM 1534
C     ------   OH   ------                                               NKM 1535
      FORM = R(6)*Y(4+J) + R(10)*Y(1+J)*Y(22+J) +R(55)*Y(15+J)           NKM 1536
      DSTROY = R(7)*Y(1+J) + R(8)*Y(2+J) + R(9)*Y(5+J) + R(18)*Y(8+J)    NKM 1537
     1  +R(19)*Y(9+J) + R(35)*Y(10+J) + R(42)*Y(7+J) + R(48)*Y(6+J)      NKM 1538
     2  + R(52)*Y(14+J) + (R(50)+R(51)+R(57))*Y(12+J)                   NKM 1539
      CONIN(31,K) = FORM/DSTROY                                          NKM 1540
50    CONTINUE                                                           NKM 1541
      RETURN                                                             NKM 1542
```

```
                                                                                NKM 1543
      END
                                                                                NKM 1544
      SUBROUTINE TEMPR(IT,TIME,T)                                               NKM 1545
C     DETERMINES TEMPERATURE DEPENDENT REACTION RATES                           NKM 1546
C                                                                               NKM 1547
C     DIMENSION T(1)                                                            NKM 1548
      COMMON/CHEM2/ CONIN(40,5),          WTMOLE(40),          RATKON(65),      NKM 1549
     1              RATEFF(65),           RATEV(4,5),          QRATE,           NKM 1550
     2              NVRATE,               LOCVRT(4)                             NKM 1551
      RT = 1.986*T(IT)                                                          NKM 1552
C     HNO4 DISSOCIATION RATE                                                    NKM 1553
      RATKON(22) = 7.8E15 * EXP(-20700./RT)                                     NKM 1554
      RATEFF(22) = RATKON(22)                                                   NKM 1555
C     N2O5 DISSOCIATION RATE                                                    NKM 1556
      RATKON(17) = 60.*5.7E14*EXP(-21051./RT)                                   NKM 1557
      RATEFF(17) = RATKON(17)                                                   NKM 1558
C     PAN DISSOCIATION RATE                                                     NKM 1559
      RATKON(46) = 1.17E17 * EXP(-26910./RT)                                    NKM 1560
      RATEFF(46) = RATKON(46)                                                   NKM 1561
      WRITE(10,1) TIME, T(IT), RATKON(17), RATKON(22), RATKON(46)               NKM 1562
      RETURN                                                                    NKM 1563
      FORMAT(1H0, 1X,40HTEMPERATURE DEPENDENT RATES  TIME =                     NKM 1564
     1  ,F8.2,5X,6HTEMP =,F8.2,5X,7HRATES = ,3E10.3)                            NKM 1565
      END                                                                       NKM 1566

      SUBROUTINE UNMIXR(KSTN)                                                   NKM 1567
C     DETERMINES RATE CONSTANTS AS A FUNCTION OF ELEVATION (MESH POINT)         NKM 1568
C     FOR REACTIONS INVOLVING ALKE (HIGHER OLEFINS).                            NKM 1569
C                                                                               NKM 1570
C     THIS ROUTINE IS SPECIFIC TO THE ERT PHOTOCHEMICAL MECHANISM               NKM 1571
C     39 SPECIES X 64 REACTIONS   1.31.79                                       NKM 1572
C                                                                               NKM 1573
C     COMMON /CHEM2/ CONIN(40,5),WTMOLE(40),RATKON(65),RATEFF(65),              NKM 1574
     X  RATEV(4,5),QRATE,NVRATE,LOCVRT(4)                                       NKM 157
C     IF(KSTN.GT.2)  GO TO 10                                                   NKM 157
C                ---   50/50  -------                                           NKM 15
      RATEFF(18) = 5.8E+4                                                       NKM 1
      RATEFF(23) = 1.46E+4                                                      NKM
      RATEFF(25) = 4.58E-2                                                      NKM
      RETURN                                                                    NK
      IF(KSTN.GT.4)  GO TO 20                                                   NK
   10          ----  75/25  ------                                              N
C     RATEFF(18) = 4.75E+4
      RATEFF(23) =   9.9E+3
      RATEFF(25) =  2.37E-2
      RETURN
   20 CONTINUE
C                --  100/0  ---
      RATEFF(18) = 3.7E+4
      RATEFF(23) = 5.2E+3
```

```
      RETURN
      END

C     SUBROUTINE UPFLX1(TIME,J)                                          NKM 1595
C                                                                        NKM 1596
C       *UPFLX1* UPDATES AREA SOURCE EMISSION FLUXES
C
C       THIS ROUTINE HAS BEEN MODIFIED TO PARTITION A SINGLE             NKM 1597
C       ALDEHYDE FLUX INTO 60% HCHO AND 40% RC=O.                        NKM 1598
C       OLEFIN FLUX INTO HALF C2H4 AND HALF ALKE                         NKM 1599
      COMMON/FLUXES/FLXIN( 7,200),  FLXTIM(200),   NFLUX                 NKM 1600
     1 COMMON/CHEM3/ ZEE(5),              DELZ(4),                       NKM 1601
     2                TDELZ(2) ,          DFINIT(6),                     NKM 1602
     3                DCOF(5),            FLXHAL(40),                    NKM 1603
     4                BCFLAG(40),         DPRATE(10),                    NKM 1604
      COMMON /CHEM5/  LOCDPF(10),         NDPFLX ,                       NKM 1605
     1                REACT(20,55),SPEC(40),LOCFLX(10),                  NKM 1606
      INTEGER BCFLAG  NASFLX,FLXW1(30),FLXW2(30)                         NKM 1607
      IF(NASFLX.LE.0)  RETURN                                            NKM 1608
      WRITE(10,1)TIME                                                    NKM 1609
      JP1 =MIN0(J+1,NFLUX)                                               NKM 1610
      DO 100 I = 1,NASFLX                                                NKM 1611
      K = LOCFLX(I)                                                      NKM 1612
      FLXW1(K) = FLXIN(I,J)                                              NKM 1613
  100 FLXW2(K) = FLXIN(I,JP1)                                            NKM 1614
C     CONTINUE                                                           NKM 1615
C                                                                        NKM 1616
C         SPLIT ALDEHYDES                                                NKM 1617
                                                                         NKM 1618
      FLXW1(7) = 0.40*FLXW1(6)                                           NKM 1619
      FLXW2(7) = 0.40*FLXW2(6)                                           NKM 1620
      FLXW1(6) = 0.60*FLXW1(6)                                           NKM 1621
C     FLXW2(6) = 0.60*FLXW2(6)                                           NKM 1622
C                                                                        NKM 1623
C       SPLIT NOX                                                        NKM 1624
                                                                         NKM 1625
      FLXW1(2) = 0.10*FLXW1(1)                                           NKM 1626
      FLXW2(2) = 0.10*FLXW2(1)                                           NKM 1627
      FLXW1(1) = 0.90*FLXW1(1)                                           NKM 1628
C     FLXW2(1) = 0.90*FLXW2(1)                                           NKM 1629
C                                                                        NKM 1630
C         SPLIT OLEFINS                                                  NKM 1631
                                                                         NKM 1632
      FLXW1(9) = .5*FLXW1(8)                                             NKM 1633
      FLXW2(9) = .5*FLXW2(8)                                             NKM 1634
      FLXW1(8) = .5*FLXW1(8)                                             NKM 1635
      FLXW2(8) = .5*FLXW2(8)                                             NKM 1636
      DO 200 I = 1,NASFLX                                                NKM 1637
      K = LOCFLX(I)                                                      NKM 1638
      WRITE(10,2) SPEC(K),FLXW1(K)                                       NKM 1639
0     CONTINUE                                                           NKM 1640
      K = 7                                                              NKM 1641
      WRITE(10,2) SPEC(K),FLXW1(K)                                       NKM 1642
      K = 2                                                              NKM 1643
                                                                         NKM 1644
      HTCELL(6),                                                         NKM 1645
      SCALOW(4),                                                         NKM 1646
      FLXDGE(40),                                                        NKM 1647
      DEPOWR(10),                                                        NKM 1648
      SCALUP(4)                                                          NKM 1649
```

```
      WRITE(10,2) SPEC(K),FLXW1(K)                                      NKM 1650
      K = 9                                                             NKM 1651
      WRITE(10,2) SPEC(K), FLXW1(K)                                     NKM 1652
    1 FORMAT(1H0,10X,38HAREA SOURCE FLUXES UPDATED AT TIME   ,F10.2)    NKM 1653
    2 FORMAT(1H ,10X,A4,3H = ,2E15.5)                                   NKM 1654
      RETURN                                                            NKM 1655
      END                                                               NKM 1656


      SUBROUTINE UPSORC(TIME,IPS,NOSTAT,SPEC)                           NKM 1657
C                                                                       NKM 1658
C     UPDATES POINT SOURCE EMISSION RATES                               NKM 1659
C                                                                       NKM 1660
      DIMENSION SPEC(1)                                                 NKM 1661
      COMMON /PS1/  TPASS(200),    PS(7,5,75),    FRACT(3),             NKM 1662
     1              NPTSR,         NPSFLX,        LOCPSF(7)             NKM 1663
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT   NKM 1664
C                                                                       NKM 1665
      IF(NPSFLX.LE.0)  GO TO 5                                          NKM 1666
      IF(IPS.LE.NPTSR)  GO TO 10                                        NKM 1667
      CALL XMIT(-150,0.0,PSRATE)                                        NKM 1668
    5 RETURN                                                            NKM 1669
   10 CONTINUE                                                          NKM 1670
      IPSP1 = MIN0(IPS+1,NPTSR)                                         NKM 1671
      DO 20 J = 1,NOSTAT                                                NKM 1672
      DO 20 I = 1,NPSFLX                                                NKM 1673
      K = LOCPSF(I)                                                     NKM 1674
      PSR1(K,J) = PS(I,J,IPS)                                           NKM 1675
      PSR2(K,J) = PS(I,J,IPSP1)                                         NKM 1676
   20 CONTINUE                                                          NKM 1677
C                                                                       NKM 1678
C     SPLIT ALDEHYDES, NOX, AND OLEFIN  EMISSIONS                       NKM 1679
C                                                                       NKM 1680
      DO 30 J = 1,NOSTAT                                                NKM 1681
      PSR1(7,J) = 0.40*PSR1(6,J)                                        NKM 1682
      PSR2(7,J) = 0.40*PSR2(6,J)                                        NKM 1683
      PSR1(6,J) = 0.60*PSR1(6,J)                                        NKM 1684
      PSR2(6,J) = 0.60*PSR2(6,J)                                        NKM 1685
      PSR1(2,J) = 0.10*PSR1(1,J)                                        NKM 1686
      PSR2(2,J) = 0.10*PSR2(1,J)                                        NKM 1687
      PSR1(1,J) = 0.90*PSR1(1,J)                                        NKM 1688
      PSR2(1,J) = 0.90*PSR2(1,J)                                        NKM 1689
      PSR1(9,J) = .5*PSR1(8,J)                                          NKM 1690
      PSR2(9,J) = .5*PSR2(8,J)                                          NKM 1691
      PSR1(8,J) = .5*PSR1(8,J)                                          NKM 169
      PSR2(8,J) = .5*PSR2(8,J)                                          NKM 169
   30 CONTINUE                                                          NKM 169
C                                                                       NKM 169
C     WRITE UPDATES ON TAPE10                                           NKM 169
C                                                                       NKM 16
      WRITE(10,1)  TIME                                                 NKM 16
      DO 40 K = 1,NPSFLX                                                NKM 16
      ID = LOCPSF(K)                                                    NKM 16
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)                      NKM 16
   40 CONTINUE                                                          NKM 1
      ID = 2                                                            NKM 1
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)                      NKM 1
```

```
      ID = 7
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)
      ID = 9
      WRITE(10,2) SPEC(ID), (PSR1(ID,J),J=1,NOSTAT)
1     RETURN
2     FORMAT(1H0,10X,37HPOINT SOURCE RATES UPDATED AT TIME =  ,F8.2)
      FORMAT(1H ,10X,A4,4H =      ,5E12.3)
      END
```

NKM 1705
NKM 1706
NKM 1707
NKM 1708
NKM 1709
NKM 1710
NKM 1711
NKM 1712

TECHNICAL REPORT DATA
(Please read Instructions on the reverse before completing)

1. REPORT NO.	2.	3. RECIPIENT'S ACCESSION NO.
EPA-600/8-79-015b		

4. TITLE AND SUBTITLE	5. REPORT DATE
A LAGRANGIAN PHOTOCHEMICAL AIR QUALITY SIMULATION MODEL Adaptation to the St. Louis - RAPS Data Base Volume II. User's Manual	June 1979 6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) Fred Lurmann, Daniel Godden, Alan C. Lloyd and Richard A. Nordsieck	8. PERFORMING ORGANIZATION REPORT NO.
9. PERFORMING ORGANIZATION NAME AND ADDRESS Environmental Research and Technology, Inc. 2625 Townsgate Road Westlake Village, CA 91361	10. PROGRAM ELEMENT NO. 1AA603A AA-045 (FY-79) 11. CONTRACT/GRANT NO. 68-02-2765
12. SPONSORING AGENCY NAME AND ADDRESS Environmental Sciences Research Laboratory - RTP, NC Office of Research and Development U.S. Environmental Protection Agency Research Triangle Park, NC 27711	13. TYPE OF REPORT AND PERIOD COVERED Final 14. SPONSORING AGENCY CODE EPA/600/09

15. SUPPLEMENTARY NOTES

Volume I. Model Formulation -- EPA-600/8-79-015a, June 1979

16. ABSTRACT

A set of instructions have been compiled for use of a Lagrangian photochemical air quality simulation model adapted to the St. Louis, Missouri/Illinois metropolitan region and the Regional Air Pollution Study (RAPS) data base. The computer model, developed by Environmental Research and Technology, Inc., consists of a set of computer programs for the simulation of atmospheric transport, turbulent diffusion, and chemical kinetics of photochemical pollutants. The model is used to predict atmospheric concentrations of ozone, nitrogen dioxide, carbon monoxide, sulfur dioxide, and sulfate within an air column moving at the mean wind speed.

Descriptions of the meteorological, source emissions, and air quality data requirements, as well as sample input and output files, are provided. The computational procedures for using the model and a listing of the computer code are included.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group
* Air pollution * Hydrocarbons * Nitrogen oxides * Ozone Data * Adaptation * Mathematical models * Photochemical reactions * Manuals		13B 07C 07B 12A 07E 05B

18. DISTRIBUTION STATEMENT	19. SECURITY CLASS *(This Report)*	21. NO. OF PAGES
RELEASE TO PUBLIC	UNCLASSIFIED	454
	20. SECURITY CLASS *(This page)*	22. PRICE
	UNCLASSIFIED	

EPA Form 2220-1 (9-73)

U S ENVIRONMENTAL PROTECTION AGENCY
Office of Research and Development
Environmental Research Information Center
Cincinnati, Ohio 45268

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE, $300
AN EQUAL OPPORTUNITY EMPLOYER

POSTAGE AND FEES PAID
U S ENVIRONMENTAL PROTECTION AGENCY
EPA-335

If your address is incorrect, please change on the above label tear off, and return to the above address.
If you do not desire to continue receiving these technical reports, CHECK HERE ☐, *tear off label, and return it to the above address,*

EPA-600/8-79-015b