EPA/600/8-90/083b
JANUARY 1991

THE REGIONAL OXIDANT MODEL (ROM) USER'S GUIDE
PART 2:
THE ROM PROCESSOR NETWORK

by

L. MILICH · J. EICHINGER · C. MAXWELL · A. VAN METER
T. BOEHM · S. FUDGE · D. OLERUD · R. WAYLAND
C. COATS · D. JORDAN · R. TANG · J. YOUNG

Computer Sciences Corporation
Research Triangle Park, NC 27709

and

O. BULLOCK* · T. PIERCE* · K. SCHERE*
J. NOVAK* · S. ROSELLE*

Atmospheric Sciences Modeling Division
Atmospheric Research and Exposure Assessment Laboratory
Research Triangle Park, NC 27711

Contract No. 68-01-7365

Technical Monitor
Joan Novak*
Atmospheric Sciences Modeling Division
Atmospheric Research and Exposure Assessment Laboratory
Research Triangle Park, NC 27711

*On assignment from the National Oceanic and Atmospheric Administration,
U.S. Department of Commerce

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

EPA/600/8-90/083b
JANUARY 1991

THE REGIONAL OXIDANT MODEL (ROM) USER'S GUIDE
PART 2:
THE ROM PROCESSOR NETWORK

by

L. MILICH
T. BOEHM
C. COATS
J. EICHINGER
S. FUDGE
D. JORDAN
C. MAXWELL
D. OLERUD
R. TANG
A. VAN METER
R. WAYLAND
J. YOUNG

Computer Sciences Corporation
Research Triangle Park, NC 27709

and

O. BULLOCK*
J. NOVAK*
T. PIERCE*
S. ROSELLE*
K. SCHERE*

Atmospheric Sciences Modeling Division
Atmospheric Research and Exposure Assessment Laboratory
Research Triangle Park, NC 27711

Contract No. 68-01-7365

Technical Monitor
Joan Novak*
Atmospheric Sciences Modeling Division
Atmospheric Research and Exposure Assessment Laboratory
Research Triangle Park, NC 27711

*On assignment from the National Oceanic and Atmospheric Administration,
U.S. Department of Commerce

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

Mention of trade names or commercial products
does not constitute endorsement or recommendation for use.

TABLE OF CONTENTS

APPENDICES

LIST OF TABLES

Because we believe that a "normal" List of Tables for Section 2, the ROM Processors, is not useful (due to the large number of tables and the fact that many are repeated), we formatted the List of Tables unconventionally. Tables for all Sections, except for Section 2, are treated in the order of their occurrence. *Preceding* this, in an alphabetized, non-numbered format, are (1) a list of input and output files for the ROM processors where each file is listed only at its first occurrence; (2) a list of control card variables by processor number; and (3) a list of I/O file space requirements by processor number.

LIST OF FIGURES

SECTION 1

I/O SUPPORT FOR THE ROM PROCESSOR NETWORK

1.1 SYSTEM CONCEPTS

1.1.1 G Series Processors

The G Series Processors consist of a set of 35 programs that are run in a sequence of nine stages; they prepare the data required by the Regional Oxidant Model (ROM). Detailed functional specifications for these processors are available in Section 3 of this Part of the User's Guide. Processors are named P*nn*G, where *nn* is the processor number. Each stage of the processor network consists of one or more processors that read files created at an earlier stage and write files that are read at a later stage. The processors create a total of 117 files that are divided into two groups: processor files (PF), and model files (MF).

1.1.2 Processor and Model Files (PF/MF) Database

The PF/MF database consists of 117 files that are written and read only by processors (and not the Core Model). Each file is assigned a number from 000 to 999, and an internal or generic name of the form "PF*nnn*" or "MF*nnn*", where *nnn* is the file number. Detailed descriptions of all files within the database are available in Section 2, "The ROM Processor Network." Note that we use the nomenclature *standard files* for these files.

1.1.3 Use of Internal Names Instead of Unit Numbers

The use of unit numbers within each processor would make it difficult to trace and document the flow of data through the system, since each processor would assign a different unit number to each file. Due to the large number of files flowing through the system, PF/MF files are, by convention, written and read using their internal (generic) name (i.e., "PF*nnn*" or "MF*nnn*"). This permits all processors to read and write all files without regard to variations in naming. Users are relieved of the burden of determining which unit numbers to attach to which files in each processor, and run streams are simplified. The data flow through the system can thus be more easily and consistently traced.

1.1.4 Dynamic Assignment of Unit Numbers

Each file to be written or read by a processor is assigned (via DEC Control Language, DCL) by its external name (i.e., the name of the file as it is stored on disk) to a logical name, and its internal name is attached (via DCL) to a logical name prior to execution. (Refer to any run stream in the User's Guide for an example of this process.) Each processor is coded to use only internal names and not unit numbers. Since unit numbers are not assigned prior to execution, each processor dynamically selects a unit number and attaches the unit number to each file using the file's internal name and the function JUNIT. The I/O routines select, attach, and use unit numbers for each PF/MF file.

1.1.5 Standard File Structures and I/O Routines

During development of the ROM, and to simplify the I/O requirements of the processors, we defined a set of five basic file structures for the PF/MF files. Note that while the second file structure is not used in ROM2.1, it will be included in subsequent versions of the model. We developed standard subprograms to read and write each file structure. The use of standard file structures and I/O routines allows the data to flow through the system in a traceable and well-defined manner, making it easy to perform quality control on the standard files.

1.2 DATA FLOW COMPONENTS FOR G SERIES PROCESSORS

- Processor P*nn*G: Any program that must write to or read from the PF/MF database.
- PF/MF Directory: Contains structure definitions of all files in PF and MF databases.
- WRFT*n*: Subroutine that writes to a processor or model file of type *n*.
- RDFT*n*: Subroutine that reads from a processor or model file of type *n*.
- PF/MF database: The database of standard files that are written or read by the G series processors.
- OPEN2: Subroutine that opens all processor or model files that are to be written to or read from.

1.3 GENERAL I/O GUIDELINES

1.3.1 No Tape Files

All I/O should be directly to disk-resident files. Large disk files may be stored on tape to conserve disk space. These files should be copied from tape to temporary disk storage prior to execution of the ROM processors, and, if necessary, copied from temporary disk storage to tape after program execution and after quality control procedures have been completed.

1.3.2 Assigning Internal File Names

All processor or model files must be assigned by their proper PF internal (logical) name (PF*nnn*) or MF internal (logical) name (MF*nnn*), attached via DCL assignment commands prior to program execution. An internal (logical) file name must not be a unit number. Refer to Sections 1.1.3 and 1.1.4 for further details.

1.3.3 Assigning the PF/MF Directory

The internal (logical) name "PFMFDIR" must be attached (via DCL commands) to the database directory file. This file must be assigned to any program that reads or writes any processor or model file.

1.3.4 Assigning Processor and Model Files

All processor or model files must be assigned by their proper PF internal (logical) name (PF*nnn*) or MF internal (logical) name (MF*nnn*), as well as by their external name. For example,

```
$ ASSIGN MET56:[ROMNET.D85188]PF106A.DAT                PF106
```

where `PF106A.DAT` is the external file name, and `PF106` is the internal (logical) file name.

1.3.5 Assigning Nonstandard Files

All other files must be assigned by their external names and some internal (logical) file name, attached to each via DCL commands prior to program execution. For example,

```
$ ASSIGN MET35:[SCRATCH.VMQ]GUTBIONO.DAT                UROUT
```

where `GUTBIONO.DAT` is the external file name, and `UROUT` is the internal (logical) file name.

1.3.6 Run Date/Time

A call to subroutine PROGID is the first executable statement in all main programs. PROGID saves the current date/time in a common block that is used to update the processor or model files' header records after writing to these files.

1.3.7 Array List of Processor or Model Files in Main Programs

Each main program contains a character array that defines the list of internal (logical) names of all the processor or model files that the program writes or reads. The main programs also contain a logical array that defines the read/write access required for each processor or model file. Input files must be defined as read-only files, and output files must be defined as read/write files.

1.3.8 Opening the PF/MF Directory

The PF/MF Directory must be opened via subroutine OPDIR2 prior to opening any processor or model files. OPDIR2 will dynamically select (via function JUNIT) a unit number; it will attach this unit number to the PF/MF Directory.

1.3.9 Opening Processor or Model Files

All processor or model files must be opened by passing the array list of PF/MF internal (logical) file names to subroutine OPEN2. A processor or model file must be opened prior to writing to or reading from the file. OPEN2 will dynamically select (via function JUNIT) a unit number; it will attach the unit number to each processor or model file.

1.3.10 File Status Summaries

We recommend printing a file status summary both after the processor or model files have been opened and again immediately before the program terminates normally. This is done by passing the array list of PF/MF internal (logical) file names to subroutine FLSMRY.

1.3.11 Opening Nonstandard Sequential Files

To avoid the possibility of duplicate unit numbers, each nonstandard file must be opened by function JFILE2. JFILE2 dynamically selects (via function JUNIT) a unit number, and attaches the unit number to the sequential file. The unit number that is returned must be used on all reads/writes of that file.

1.3.12 Opening Alternate Print Files

To avoid the possibility of duplicate unit numbers, each print file must be opened by function JFILE3. JFILE3 dynamically selects (via function JUNIT) a unit number, and attaches the unit number to the sequential file. The unit number that is returned must be used on all reads/writes of that file.

1.3.13 Opening Nonstandard Direct Access Files

To avoid the possibility of duplicate unit numbers, a unit number for each nonstandard direct access file is obtained by function JUNIT. The FORTRAN OPEN statement is then used to attach the unit number and define the file as a direct access file.

1.3.14 Writing/Reading Processor or Model Files

All processor or model files of type 1 are written by subroutine WRFT1 and read by subroutine RDFT1. Subroutines WRFT*n* and RDFT*n* are used for writing and reading processor or model files of type n (n = 2 to 5). File types are defined in Section 1.4.1. Note that model file type 2 is not used in ROM2.1.

1.3.15 The 50-File Limit

A maximum of 50 files (PF/MF and all other files) can be written and/or read by any one program.

1.4 PF/MF DESCRIPTIONS

1.4.1 General Description

All Processor or Model files are ASCII FORTRAN, direct access, unformatted files. Each file contains one standard header record followed by a maximum of 72 data records. Each file contains a maximum of 19 parameters. Files are classified by *type*, based on structure and content. Type 1 and 2 files contain upper-air vertical profiles at radiosonde (rawind) stations at hourly or 12-hourly intervals. Type 3 files contain meteorology and air quality parameters at surface stations at hourly intervals. Type 4 files contain time-invariant gridded data. Type 5 files contain time-variant gridded data. The file types are summarized in Table 1.1. Note that model file type 2 is not used in ROM2.1.

TABLE 1.1. FILE TYPE SUMMARY

File Type	File Body Record Length	Max. Records	Contents
1	151+(4500xNP)	73	Rawind stations upper-air vertical profiles
2	121+(30xNP)	73	Rawind stations single level parameters (e.g., mixing heights)
3	1001+(250xNP)	73	Surface meteorology and air quality parameters
4	NCxNRxNP	2	Gridded time-invariant data (e.g., land use)
5	NCxNRxNP	73	Gridded time-variant data

Note: NP is the number of parameters in the file, NC and NR are the number of columns and rows in the model domain.

1.4.2 File Headers for Processor and Model Files

1.4.2.1 File Header Characteristics

Recording Mode: ASCII FORTRAN, direct access, unformatted

Record Size: 539 words

Number of records: 1

1.4.2.2 File Header Descriptions

This record is the first record in every processor or model file, and contains the following information:

- dates and times of (1) file creation, and (2) last update;
- the structural definition of the file;
- a general description of the file;
- the number of parameters in the file, and descriptions of each parameter;
- the grid region definition; and
- the time span contained in the file.

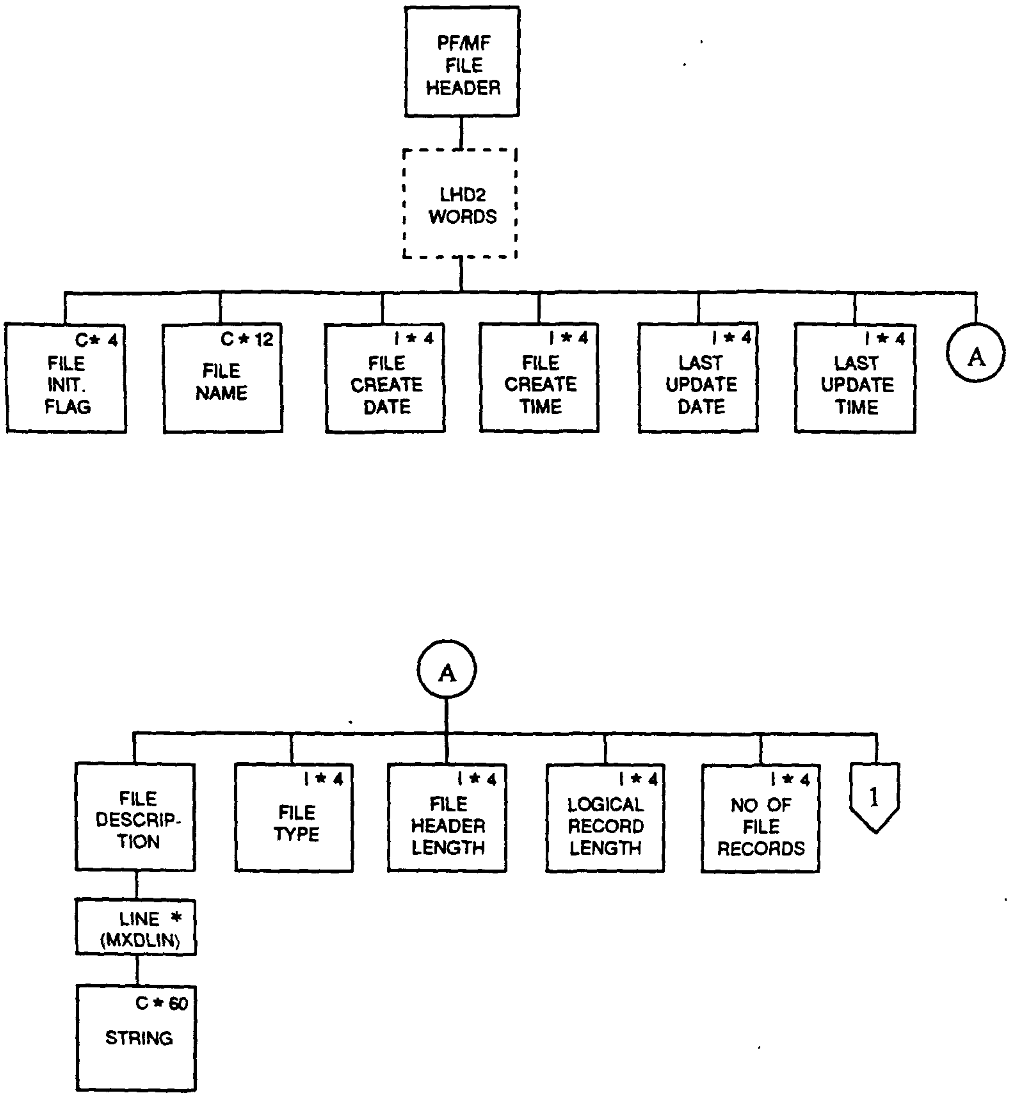
The file header contains a maximum of 19 parameters.

1.4.2.3 Header Record Data Dictionary

Table 1.2 defines the parameters that appear in header records. Figure 1.1 is a structure chart of the processor and model file header records.

TABLE 1.2. HEADER RECORD PARAMETER DEFINITIONS

Parm. Name	Type	Definition
KEY	Char*4	Defines initialization status of file = "INIT" if file has been initialized ≠ "INIT" if file has not been initialized
FNAME	Char*12	Internal (logical) name of file (PF*nnn* or MF*nnn*)
CDATE	Integer	Creation date (*MMDDYY*) of file
CTIME	Integer	Creation time (*HHMMSS*) of file
WDATE	Integer	Date (*MMDDYY*) of last update
WTIME	Integer	Time (*HHMMSS*) of last update
FDESC(i)	Char*60	Line *i* of general file description
FTYPE	Integer	Type of file (1, 2, 3, 4, or 5)
LHEAD	Integer	Length of header record (words)
LRECL	Integer	Logical record length (words)
LBLOCK	Integer	Length of block (words)
MRECS	Integer	Max. no. of records (header+data) that the file contains
NPARM	Integer	Number of parameters in file
PNAMES(i)	Char*12	Name of parameter *i*
UNITS(*i*)	Char*12	Units of parameter *i*
PDESCS(i)	Char*40	Description of parameter *i*
GRDNAM	Char*8	Model domain name (not used on file types 1, 2, and 3)
NCOL	Integer	No. of model domain columns (not used on file types 1, 2, and 3)
NROW	Integer	No. of model domain rows (not used on file types 1, 2, and 3)
JDO	Integer	Julian Date (*YYDDD*) of first hour in scenario (not used on file type 4)
IHRO	Integer	Time (00 to 23 local standard) of first hour in scenario (not used on file type 4)
HRINC	Integer	Number of hours in time interval between records (= 1 or 12; not used on file type 4)
MHRS	Integer	Max. no. of hour records in file (= 72; not used on file type 4)
HOURS(i)	Logical	Status of hour record *i* in file (not used on file type 4) = .FALSE. if hour record has not been written = .TRUE. if hour record has been written



MXPRM1	=	25
NPARM	≤	MXPRM1
MXDLIN	=	3
MHRS	=	72
NTHRS	=	MHRS
LHD2	=	22 + 16 * MXPRM1 + 15 * MXDLIN + NTHRS

Figure 1.1. Structure Chart of the Processor and Model file Header Records (page 1 of 2).

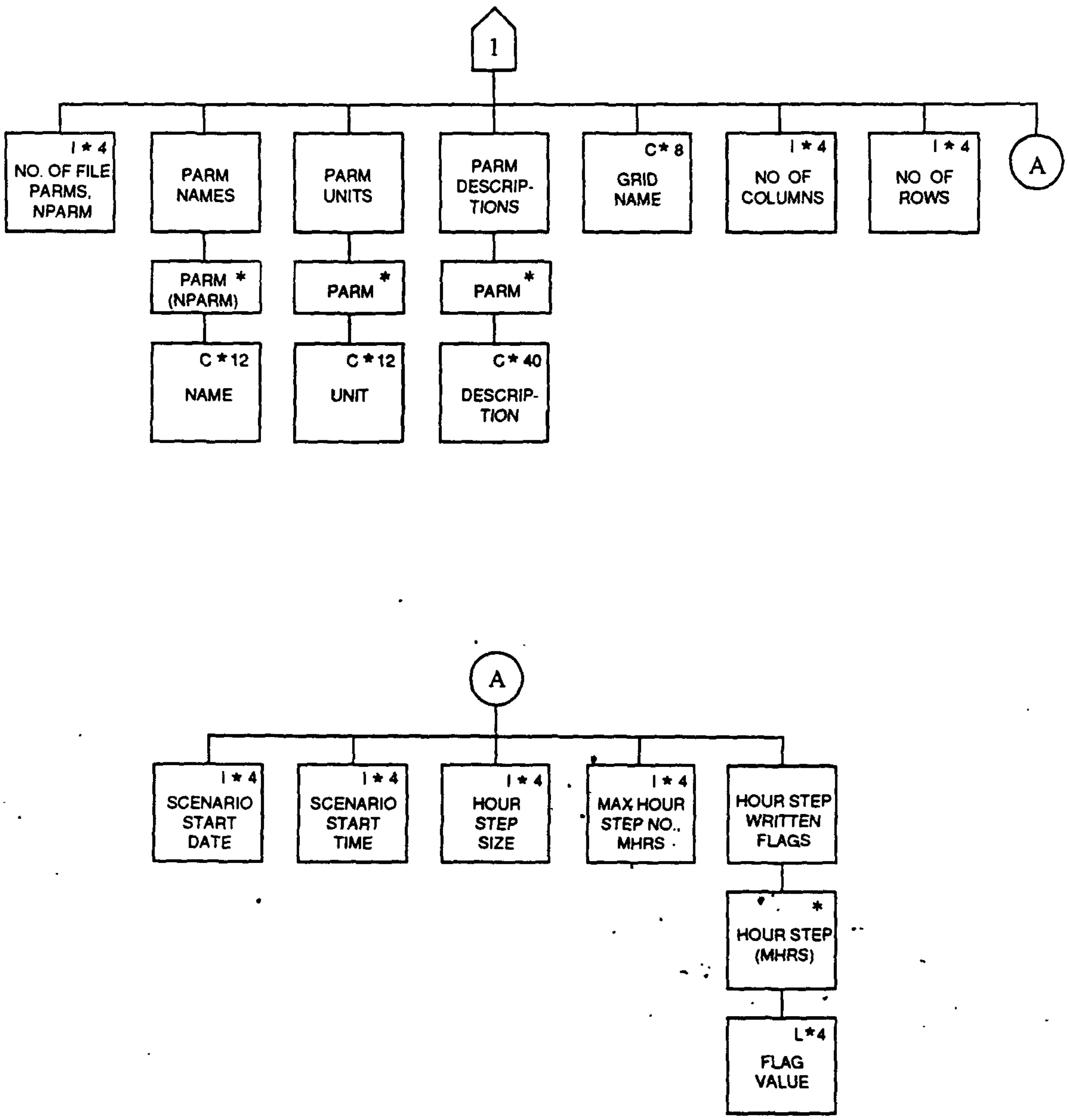


Figure 1.1 (page 2 of 2).

1.4.3 Type 1 PF: Rawind Stations Upper-Air Vertical Profiles

1.4.3.1 File Characteristics

Recording Mode: ASCII FORTRAN, direct access, unformatted

Record Size: 151+(4500xNP) words (NP = number of parameters)

Number of Records: 72 + header record

1.4.3.2 File Description

Type 1 files contain the standard PF/MF header followed by a single record for each hour of (up to) a 72-hour scenario. Each data record contains identifying information for each rawind station that was reporting for that hour, followed by the vertical profiles of upper-air parameters for each station. Each record can contain a maximum of 30 stations, with a maximum of 150 values per vertical profile. Table 1.3 defines the parameters that appear in file-type 1 records.

TABLE 1.3. FILE-TYPE 1 PARAMETER DEFINITIONS

Parm. Name	Type	Definition
NSTA	Integer	Number of rawind stations reporting for hour
STAIDS(i)	Integer	Unique identifier of rawind station *i*
ELEVS(i)	Real	Elevation (meters above sea level) of rawind station *i*
LONS(i)	Real	Longitude (degrees) of rawind station *i*
LATS(i)	Real	Latitude (degrees) of rawind station *i*
NPTS(i)	Integer	Number of readings in vertical profile for rawind station *i*
PROFS(i,j,k)	Real	Value of parameter *k* at rawind station *j* at vertical point *i* for *k*=1,NP

Note: NP is the number of parameters in the file.

Figure 1.2 is a structure chart of Type 1 Processor/Model Files.

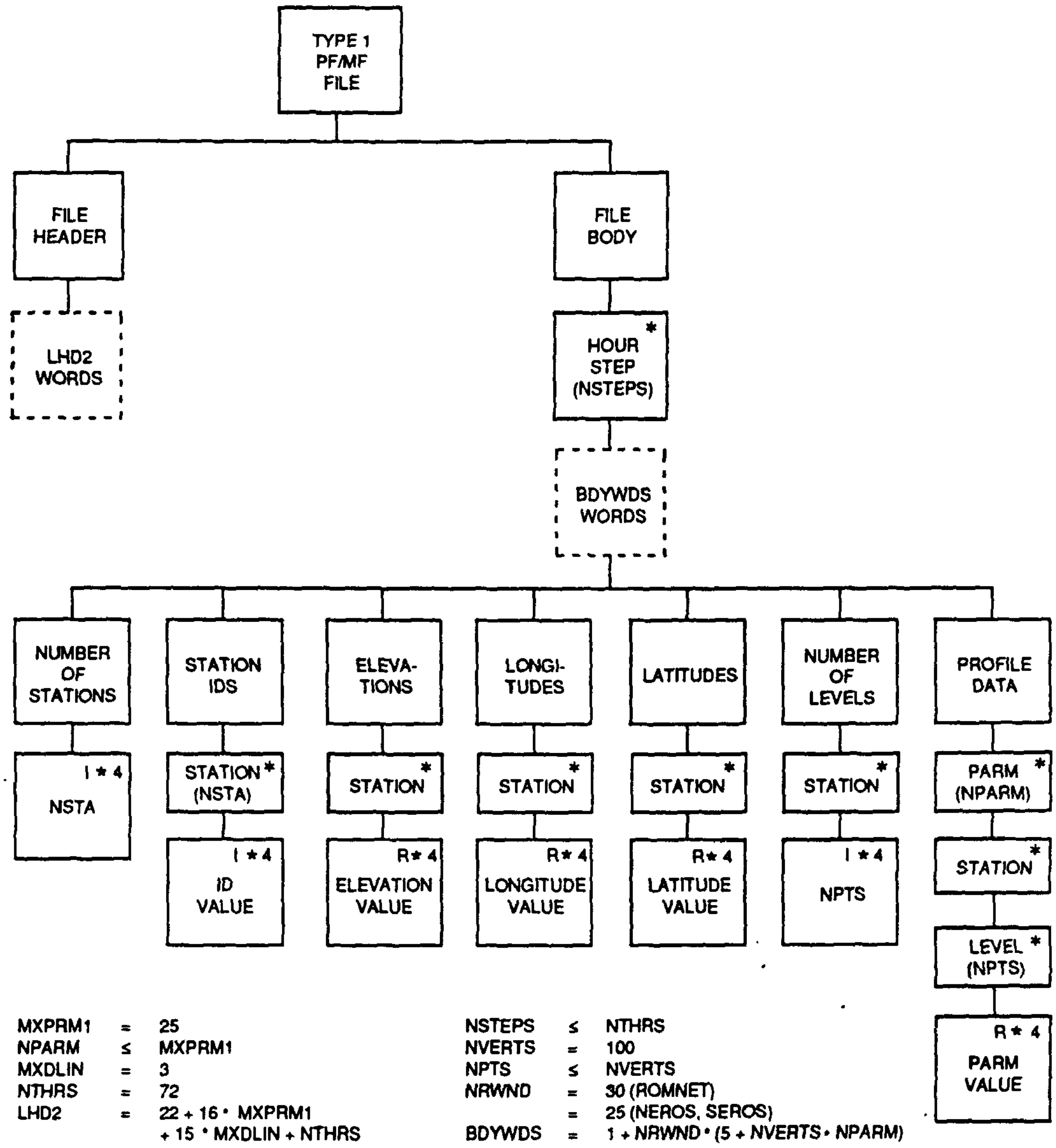


Figure 1.2. Structure Chart of Type 1 Processor/Model Files.

1.4.4 Type 2 PF: Rawind Stations Single Level Parameters (not used in ROM2.1)

1.4.4.1 File Characteristics

Recording Mode: ASCII FORTRAN, direct access, unformatted

Recording Size: 121+(30×NP) words (NP = Number of Parameters)

Number of records: 72 + header record

1.4.4.2 File Description

Type 2 files contain the standard PF/MF header followed by a single record for each hour of (up to) a 72-hour scenario. Each data record contains identifying information for each rawind station that was reporting for that hour, followed by the calculated parameters for each station. Each record can contain a maximum of 30 rawind stations. Table 1.4 defines the parameters that appear in file-type 2 records.

TABLE 1.4. FILE-TYPE 2 PARAMETER DEFINITIONS

Parm. Name	Type	Definition
NSTA	Integer	Number of rawind stations reporting for hour
STAIDS(i)	Integer	Unique identifier of rawind station *i*
ELEVS(i)	Real	Elevation (meters above sea level) of rawind station *i*
LONS(i)	Real	Longitude (degrees) of rawind station *i*
LATS(i)	Real	Latitude (degrees) of rawind station *i*
P(i,j)	Real	Value of calculated parameter *j* at rawind station *i* for $j=1$,NP

Note: NP is the number of parameters in the file.

Figure 1.3 is a structure chart of Type 2 Processor/Model Files.

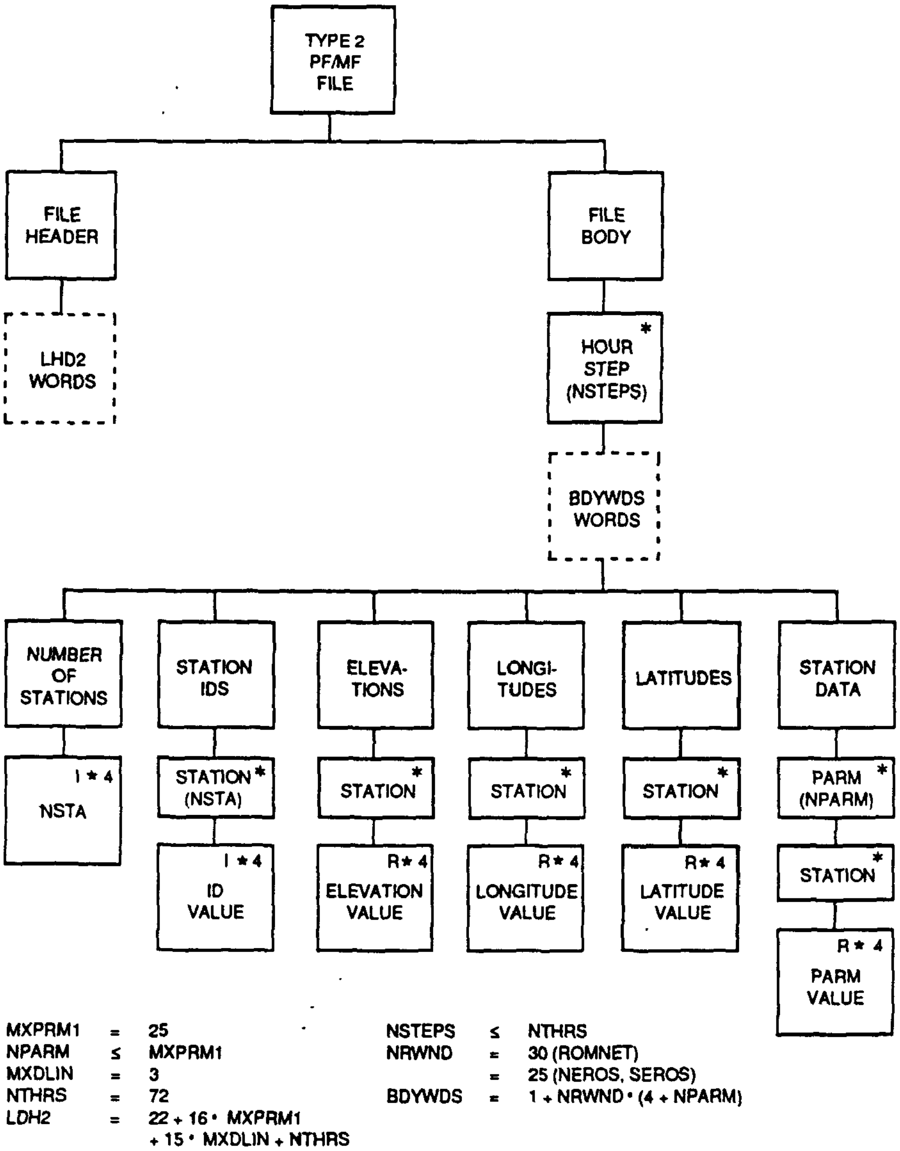


Figure 1.3. Structure Chart of Type 2 Processor/Model Files.

1.4.5 Type 3 PF: Surface Meteorology and Air Quality Parameters

1.4.5.1 File Characteristics

Recording Mode: ASCII FORTRAN, direct access, unformatted

Recording Size: 1001+(250xNP) words (NP = number of parameters)

Number of Records: 72 + header record

1.4.5.2 File Description

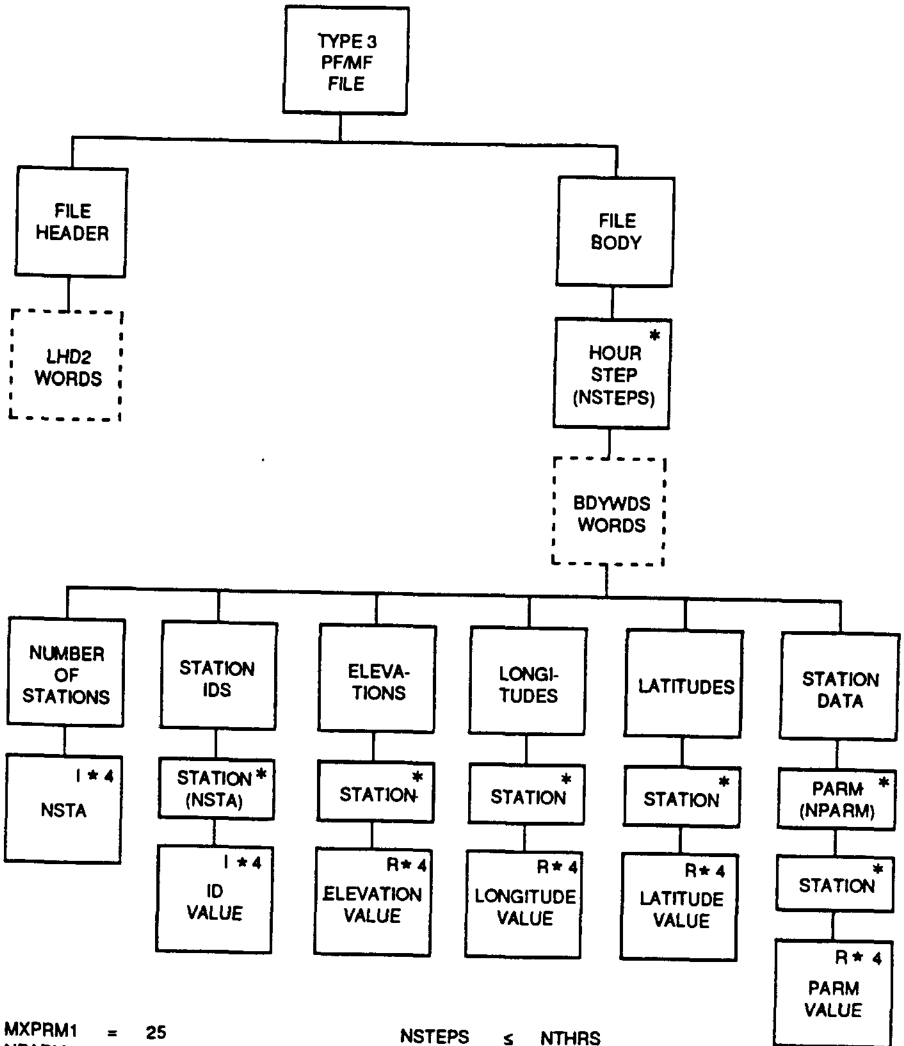
Type 3 files contain the standard PF/MF header followed by a single record for each hour of (up to) a 72-hour scenario. Each data record contains identifying information for each surface station that was reporting for that hour, followed by the observed or calculated parameters for each station. Each record can contain a maximum of 250 surface stations. Table 1.5 defines the parameters that appear in file-type 3 records.

TABLE 1.5. FILE-TYPE 3 PARAMETER DEFINITIONS

Parm. Name	Type	Definition
NSTA	Integer	Number of surface stations reporting for hour
STAIDS(i)	Integer	Unique identifier of surface station *i*
ELEVS(i)	Real	Elevation (meters above sea level) of surface station *i*
LONS(i)	Real	Longitude (degrees) of surface station *i*
LATS(i)	Real	Latitude (degrees) of surface station *i*
SROBS(i,j)	Real	Value of observed or calculated parameter *j* at surface station *i* for $j=1,NP$

Note: NP is the number of parameters in the file.

Figure 1.4 is a structure chart of Type 3 Processor/Model Files.



$$
\begin{aligned}
MXPRM1 &= 25 \\
NPARM &\le MXPRM1 \\
MXDLIN &= 3 \\
NTHRS &= 72 \\
LDH2 &= 22 + 16 \cdot MXPRM1 \\
&\quad + 15 \cdot MXDLIN + NTHRS
\end{aligned}
$$

$$
\begin{aligned}
NSTEPS &\le NTHRS \\
NSURF &= 250\ (ROMNET) \\
&= 200\ (NEROS,\ SEROS) \\
BDYWDS. &= 1 + NSURF \cdot (4 + NPARM)
\end{aligned}
$$

Figure 1.4. Structure Chart of Type 3 Processor/Model Files.

1.4.6 Type 4 PF: Gridded Time-Invariant Parameters

1.4.6.1 File Characteristics

Recording Mode:	ASCII FORTRAN, direct access, unformatted
Recording Size:	NCxNRxNP words (NC and NR are the number of columns and rows in the model domain; NP is the number of parameters)
Number of Records:	1 + header record

1.4.6.2 File Description

Type 4 files contain the standard PF/MF header followed by one record that contains time-invariant gridded parameters (e.g., land use, elevation data). Table 1.6 defines the parameters that appear in file-type 4 records.

TABLE 1.6. FILE-TYPE 4 PARAMETER DEFINITIONS

Parm. Name	Type	Definition
GRIDA(i,j,k)	Real	Value of parameter k in a grid cell at row j and column i for k=1,NP

Note: NP is the number of parameters in the file.

Figure 1.5 is a structure chart of Type 4 Processor/Model Files.

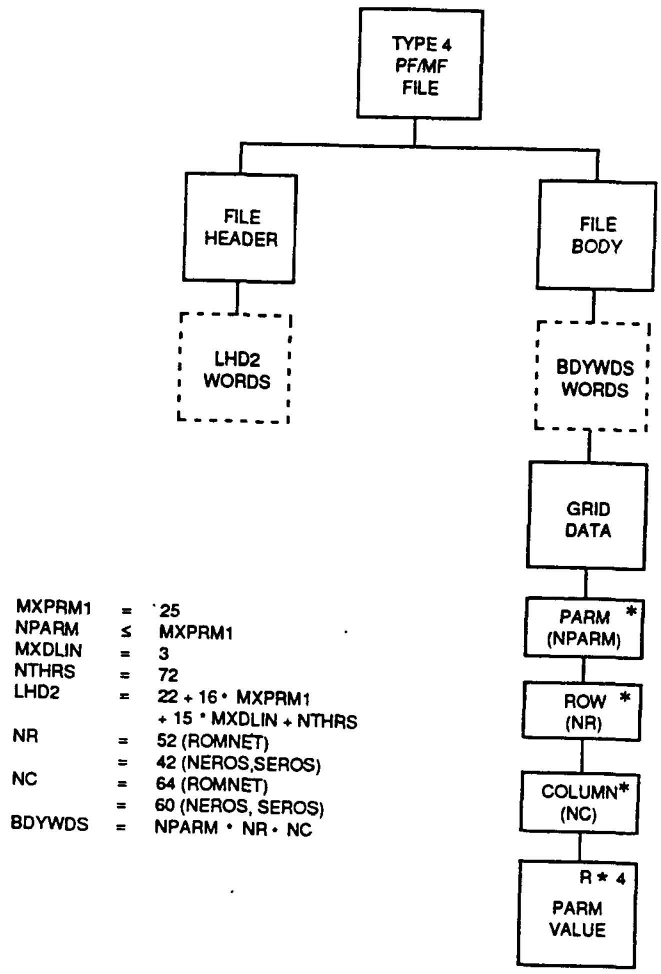


Figure 1.5. Structure Chart of Type 4 Processor/Model Files.

1.4.7 Type 5 PF/MF: Gridded Time-Variant Parameters

1.4.7.1 File Characteristics

Recording Mode:	ASCII FORTRAN, direct access, unformatted
Recording Size:	NC×NR×NP words (NC and NR are the number of columns and rows in the model domain; NP is the number of parameters)
Number of Records:	72 + header record

1.4.7.2 File Description

Type 5 files contain the standard PF/MF header followed by a single record for each hour of (up to) a 72-hour scenario. Each data record contains the gridded parameters for that hour. Table 1.7 defines the parameters that appear in file-type 4 records.

TABLE 1.7. FILE-TYPE 5 PARAMETER DEFINITIONS

Parm. Name	Type	Definition
GRIDA(i,j,k)	Real	Value of parameter k in a grid cell at row j and column i for k=1,NP

Note: NP is the number of parameters in the file.

Figure 1.6 is a structure chart of Type 5 Processor/Model Files.

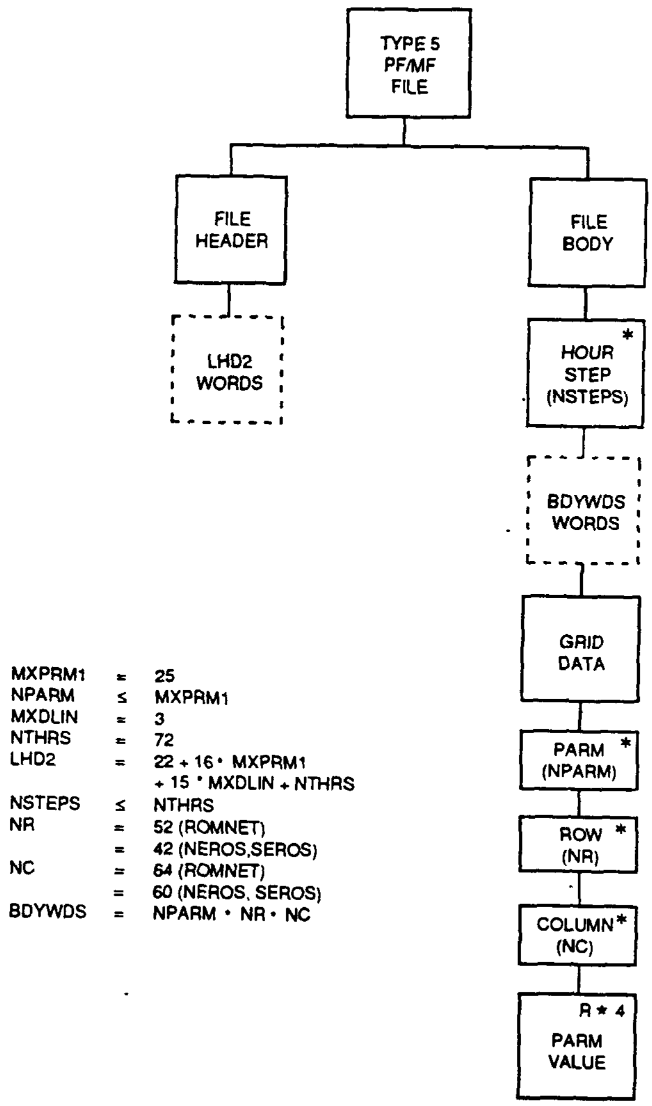


Figure 1.6. Structure Chart of Type 5 Processor/Model Files.

1.5 I/O SUBPROGRAMS

1.5.1 General Description

All subroutines and functions are written in ASCII FORTRAN and use standard FORTRAN I/O. There are many FORTRAN INCLUDE files also used, which contain parameter-specification COMMON BLOCKS that allow data to be easily accessed by several subprograms.

1.5.2 Date/Hour Conventions

Some I/O subprograms require date and time of day specifications as arguments as follows:

All date specifications must be 5-digit integers of the form (*YYDDD*) where:

YY = 2 digit year (e.g. 79)
DDD = 3 digit Julian date (e.g. 215).

Julian dates are numbered from 1 to 366 where:

1 implies January 1
365 implies December 31 of non-leap year
366 implies December 31 of a leap year.

All time of day specifications must be 2-digit integers of the form (*HH*) where:

HH = 2 digit hour in local standard time (e.g. 12)[1]

Hours are numbered from 00 to 23 where:

00 implies hour 00:00 AM to 00:59 AM
23 implies hour 23:00 PM to 23:59 PM

Example: date: 79215, hour: 12

This would represent the hour 12:00 PM to 12:59 PM on Friday, August 3, 1979.

1.5.3 Summary of the Most Frequently Called I/O Routines

DATHR1	Returns the start date/hour for scenario in processor or model file
DATHR2	Determines if date/hour may be written into processor or model file
DATHR3	Determines if date/hour has been written in processor or model file
DATHR4	Returns list of dates/hours that are written in processor or model file
FLSMRY	Prints date/time summary of processor or model files
JFILE2	Opens sequential file and attaches unit number
JFILE3	Opens alternate print file and attaches unit number
JUNIT	Selects an available unit number
OPDIR2	Opens PF/MF directory

1. Note that this User's Guide references ROMNET applications, which use Eastern Standard Time only (EPA, 1990).

OPEN2	Opens processor or model files
PROGID	Prints program name, program version, and current date/time
RDFT1	Reads a single hour record from type 1 PF
RDFT2	Reads a single hour record from type 2 PF
RDFT3	Reads a single hour record from type 3 PF
RDFT4	Reads a single record from type 4 PF
RDFT5	Reads a single hour record from type 5 processor or model files
WRFT1	Writes a single hour record to type 1 PF
WRFT2	Writes a single hour record to type 2 PF
WRFT3	Writes a single hour record to type 3 PF
WRFT4	Writes a single record to type 4 PF
WRFT5	Writes a single hour record to type 5 processor or model files

1.5.3.1 Subroutine DATHR1

```
      SUBROUTINE DATHR1(FNAME, JDATEO, IHOURO, HRINC)
      CHARACTER*12 FNAME
      INTEGER HRINC
C     INPUT ARGUMENTS:
C     FNAME          Internal (logical) file name attached to Processor or Model file
C OUTPUT ARGUMENTS:
C     JDATEO         Julian date (YYDDD) of first hour in scenario
C     IHOURO         Hour (00 to 23 local standard) of first hour in scenario
C     HRINC          Number of hours in the time interval between records
      .
      .
      .
      RETURN
      END
```

This subroutine reads the file header for a processor or model file and returns the start date/hour for the scenario, as well as the number of hours in the time interval between records. The file being read must have been previously opened by routine OPEN2. If the file is a type 4 PF/MF, the output arguments are all set to 0.

1.5.3.2 Function DATHR2

```
      FUNCTION DATHR2(FNAME, JDATEO, IHOURO, HRINC, JDATE, IHOUR)
      LOGICAL DATHR2
      CHARACTER*12 FNAME
      INTEGER HRINC
C INPUT ARGUMENTS:
C     FNAME          Internal (logical) file name attached to Processor or Model file
C     JDATEO         Julian date (YYDDD) of first hour in scenario
C     IHOURO         Hour (00 to 23 local standard) of first hour in scenario
C     HRINC          Number of hours in the time interval between records
C     JDATE          Julian date (YYDDD) of hour to be checked
C     IHOUR          Hour (00 to 23 local standard) to be checked
C OUTPUT ARGUMENTS:
C     DATHR2         Defines if date/hour may be written into Processor or Model file
C                    EQ .TRUE., Date/hour may be written into file
C                    NE .TRUE., Date/hour may not be written into file
      .
      .
      .
      RETURN
      END
```

This function determines if the date/hour given by JDATE/IHOUR is a valid hour that can be written into the processor or model file. If the date/hour is:

- ≥ the first date/hour in the file, and
- ≤ the last date/hour in the file, and
- the date/hour falls at an appropriate date/hour interval (i.e., defined by HRINC),

then the function returns a .TRUE. value. Otherwise, the function returns a .FALSE. value. Routine DATHR1 must be used to get the correct start date/hour and the number of hours in the time interval (HRINC) between records for the file. The file being read must have been previously opened by routine OPEN2. If the file is a type 4 PF/MF, the input arguments are ignored and the function returns a .TRUE. value.

1.5.3.3 Function DATHR3

```
      FUNCTION DATHR3(FNAME, JDATE, IHOUR)
      LOGICAL DATHR3
      CHARACTER*12 FNAME
C     INPUT ARGUMENTS:
C     FNAME           Internal (logical) name attached to Processor or Model file
C     JDATE           Julian date (YYDDD) of hour to checked
C     IHOUR           Hour (00 to 23 local standard) to be checked
C OUTPUT ARGUMENTS:
C     DATHR3         Defines availability of date/hour in Processor or Model file
C                    EQ .TRUE., Date/hour has been written
C                    NE .TRUE., Date/hour has not been written
      .
      .
      .
      RETURN
      END
```

This function reads the file header for a processor or model file and determines if JDATE and IHOUR have been written to the file. The file must have been previously opened by subroutine OPEN2. If the file is a type 4 PF/MF, the date/hour arguments will be ignored. The function returns a .TRUE. value if the record has been written and a .FALSE. value if it has not been written.

1.5.3.4 Subroutine DATHR4

```
       SUBROUTINE DATHR4(FNAME, JDATEO, IHOURO, HRINC, NHRS, JDATES, IHOURS)
       CHARACTER*12 FNAME
       INTEGER HRINC
       DIMENSION JDATES(72), IHOURS(72)
C INPUT ARGUMENTS:
C      FNAME         Internal (logical) file name attached to Processor or Model file
C      JDATEO        Julian date (YYDDD) of first hour in scenario
C      IHOURO        Hour (00 to 23 local standard) of first hour in scenario
C      HRINC         Number of hours in the time interval between records
C      JDATE         Julian date (YYDDD) of hour to be checked
C      IHOUR         Hour (00 to 23 local standard) to be checked
C      JDATES(i)     Julian date (YYDDD) of hour i that has been written
                     in the file
                     Array must contain 72 words.
```

```
C     IHOURS(i)      Hour (00 to 23 local standard) of hour i that has been
                     written in file
                     Array must contain 72 words.
       .
       .
       .
       RETURN
       END
```

This subroutine reads the file header for a processor or model file and returns a list of dates/hours that have been written to the file. The file must have been previously opened by subroutine OPEN2. If the file is a type 4 PF/MF, NHRS is set to 1 if the record has been written, 0 if it has not been written; other output arguments are set to 0.

1.5.3.5 Subroutine FLSMRY

```
        SUBROUTINE FLSMRY (NF, FNAMES)
        CHARACTER*12 FNAMES
        DIMENSION FNAMES(NF)
C     INPUT ARGUMENTS:
C     NF             Number of names in list
C     FNAMES(i)      Internal (logical) file name attached to Processor or Model file i
      .
      .
      .
      RETURN
      END
```

This subroutine prints the following information for each processor or model file in its list: external file name, internal (logical) file name, creation date/time, and last update date/time. This subroutine is called after the files have been opened and before normal termination of the main program.

1.5.3.6 Function JFILE2

```
        FUNCTION JFILE2 (FNAME, FORM, READ)
        LOGICAL FORM, READ
        CHARACTER*12 FNAME
C INPUT ARGUMENTS:
C     FNAME          Internal (logical) file name attached to sequential file
C     FORM           Determines type of records in file
                     EQ .TRUE., records are formatted
                     NE .TRUE., records are unformatted
C     READ           Determines read/write access for file
C                    EQ .TRUE., read only access
C                    NE .TRUE., read/write access
C     OUTPUT ARGUMENTS:
C     JFILE2         Unit number attached to sequential file
      .
      .
      .
      RETURN
      END
```

This function selects an available unit number, opens the sequential file, and attaches the selected unit number. The unit number selected is returned by the function. A message is printed to the log that echoes the internal (logical) file name and unit number. The unit number returned is used on all writes/reads of the file. We recommend that (1) input files be defined with read-only access, and (2) output files be defined with read/write access.

1.5.3.7 Function JFILE3

```
        FUNCTION JFILE3 (FNAME)
        CHARACTER*12 FNAME
C       INPUT ARGUMENTS:
C       FNAME           Internal (logical) file name attached to alternate print file
C       OUTPUT ARGUMENTS:
C       JFILE3          Unit number attached to alternate print file
        .
        .
        .
        RETURN
        END
```

This function selects an available unit number, opens the alternate print file, and attaches the selected unit number. The unit number selected is returned by the function. A message is printed to the log that echoes the internal (logical) file name and unit number. The unit number returned is used on all writes of the file.

1.5.3.8 Function JUNIT

```
        FUNCTION JUNIT ()
C       OUTPUT ARGUMENTS:
C       JUNIT           Unit number selected
        .
        .
        .
        RETURN
        END
```

This function maintains a list of unit numbers already assigned. When called, it returns the next unit number not in use from a total list of 50 available unit numbers. This next unit number is returned to the calling program.

1.5.3.9 Subroutine OPDIR2

```
        SUBROUTINE OPDIR2
        CHARACTER*12 FNAME
C       INPUT ARGUMENTS:
C       FNAME           Internal (logical) file name attached to PF/Mf file directory
        .
        .
        .
        RETURN
        END
```

This subroutine opens the PF/MF directory, and reads and prints the file header. OPDIR2 is called prior to calling subroutine OPEN2 which opens the processor and model files. OPDIR2 assumes that the internal (logical) file name "PFMFDIR" has been attached to the PF/MF directory.

1.5.3.10 Subroutine OPEN2

```
        SUBROUTINE OPEN2 (NF, FNAMES, READ)
        LOGICAL READ
        CHARACTER*12 FNAMES
        DIMENSION FNAMES(NF), READ(NF)
C      INPUT ARGUMENTS:
C      NF              Number of names in list
C      FNAMES(i)       Internal (logical) file name attached to Processor or Model file i
C      READ(i)         Determines read/write access for Processor or Model file i
C                      EQ .TRUE., read only access
C                      NE .TRUE., read/write access
       .
       .
       .
       RETURN
       END
```

This subroutine opens each processor or model file in its list, and reads and prints each file header. Subroutine OPDIR2 must be called prior to calling OPEN2.

1.5.3.11 Subroutine PROGID

```
        SUBROUTINE PROGID (PNAME, PVERS)
        CHARACTER*8 PNAME
        INTEGER PVERS
C      INPUT ARGUMENTS:
C      PNAME           Program name
C      PVERS           Program Version (YYDDD)
        CHARACTER*8 PRNAME
        INTEGER PRVERS, RUNDTE, RUNTIM
        COMMON /PRID/ PRNAME, PRYERS, RUNDTE, RUNTIM
C      PRNAME          Program name
C      PRVERS          Program Version (YYDDD)
C      RUNDTE          Run date (MMDDYY)
C      RUNTI11         Run time (HHMMSS)
       .
       .
       .
       RETURN
       END
```

This subroutine skips to the top of a new page and prints the program name, program version, and current date/time. This information is saved in common block PRID.

1.5.3.12 Subroutine RDFT1

```
        SUBROUTINE RDFT1 (FNAME, NPARM, JDATE, IHOUR, RECORD)
        CHARACTER*12 FNAME
        DIMENSION RECORD (151+4500*NPARM)
C      INPUT ARGUMENTS:
C      FNAME Internal (logical) file name attached to Processor or Model file
C      NPARM Number of parameters in file
C      JDATE Julian date (YYDDD) of hour to be read
C      IHOUR Hour (00 to 23 local standard) to be read
```

```
C     OUTPUT ARGUMENTS:
C     RECORD Address of first word in common block or buffer
C                   area that contains (151+4500*NPARM) words.
      .
      .
      .
      RETURN
      END
```

This subroutine reads a single hour's record into array RECORD from a type 1 PF file.

1.5.3.13 Subroutine RDFT2

```
      SUBROUTINE RDFT2 (FNAME, NPARM, JDATE, IHOUR, RECORD)
      CHARACTER*12 FNAME
      DIMENSION RECORD (121+30*NPARM)
C     INPUT ARGUMENTS:
C     FNAME Internal (logical) file name attached to Processor or Model file
C     NPARM Number of parameters in file
C     JDATE Julian date (YYDDD) of hour to be read
C     IHOUR Hour (00 to 23 local standard) to be read
C     OUTPUT ARGUMENTS:
C     RECORD Address of first word in common block or buffer
C                   area that contains (121+30*NPARM) words
      .
      .
      .
      RETURN
      END
```

This subroutine reads a single hour's record into array RECORD from a type 2 PF file. Note that model file type 2 is not used in ROM2.1.

1.5.3.14 Subroutine RDFT3

```
      SUBROUTINE RDFT3 (FNAME, NPARM, JDATE, IHOUR, RECORD)
      CHARACTER*12 FNAME
      DIMENSION RECORD (1001+250*NPARM)
C     INPUT ARGUMENTS:
C     FNAME Internal (logical) file name attached to Processor or Model file
C     NPARM Number of parameters in file
C     JDATE Julian date (YYDDD) of hour to be read
C     IHOUR Hour(00 to 23 local standard) to be read
C     OUTPUT ARGUMENTS:
C     RECORD Address of first word in common block or buffer
C                   area that contains (1001+250*NPARM) words
      .
      .
      .
      RETURN
      END
```

This subroutine reads a single hour's record into array RECORD from a type 3 PF file.

1.5.3.15 Subroutine RDFT4

```
      SUBROUTINE RDFT4 (FNAME, NCOL, NROW, NPARM, GRIDA)
      CHARACTER*12 FNAME
      DIMENSION GRIDA (NCOL, NROW, NPARM)
```

```
C       INPUT ARGUMENTS:
C       FNAME Internal (logical) file name attached to Processor or Model File
C       NPARM Number of parameters in file
C       NCOL Number of columns in grid region
C       NROW Number of rows in grid region
C       OUTPUT ARGUMENTS:
C       GRIDA Address of first word in buffer
C                     area that contains (NCOL*NROW*NPARM) words
        .
        .
        .
        RETURN
        END
```

This subroutine reads a single record into array GRIDA from a type 4 PF file.

1.5.3.16 Subroutine RDFT5

```
          SUBROUTINE RDFT5(FNAME, NCOL, NROW, NPARM, JDATE, IHOUR, GRIDA)
          CHARACTER*12 FNAME
          DIMENSION GRIDA (NCOL, NROW, NPARM)
C INPUT ARGUMENTS:
C       FNAME          Internal (logical) file name attached to Processor or Model File
C       NCOL           Number of columns in grid region
C       NROW           Number of rows in grid region
C       NPARM          Number of parameters in file
C       JDATE          Julian Date (YYDDD) of hour to be read
C       IHOUR          Hour (00 to 23 local standard) to be read
C       OUTPUT ARGUMENTS:
C       GRIDA          Address of first word in buffer
C                      area that contains (NCOL*NROW*NPARM) words
        .
        .
        .
        RETURN
        END
```

This subroutine reads a single hour's record into array GRIDA from a type 5 PF file.

1.5.3.17 Subroutine WRFT1

```
          SUBROUTINE WRFT1 (FNAME, NPARM, JDATE, IHOUR, RECORD)
          CHARACTER*12 FNAME
          DIMENSION RECORD (151+4500*NPARM)
C       INPUT ARGUMENTS:
C       FNAME          Internal (logical) file name attached to Processor or Model File
C       NPARM          Number of parameters in file
C       JDATE          Julian date (YYDDD) of hour to be written
C       IHOUR          Hour (00 to 23 local standard) to be written
C       RECORD         Address of first word in common block or buffer
C                      area that contains (151+4500*NPARM) words
        .
        .
        .
        RETURN
        END
```

This subroutine writes a single hour's record from array RECORD into a type 1 PF file.

1.5.3.18 Subroutine WRFT2

```
      SUBROUTINE WRFT2 (FNAME, NPARM, JDATE, IHOUR, RECORD)
      CHARACTER*12 FNAME
      DIMENSION RECORD (121+30*NPARM)
C INPUT ARGUMENTS:
C     FNAME          Internal (logical) file name attached to Processor or Model File
C     NPARM          Number of parameters in file
C     JDATE          Julian date (YYDDD) of hour to be written
C     IHOUR          Hour (00 to 23 local standard) to be written
C     RECORD         Address of first word in common block or buffer
C                    area that contains (121+30*NPARM) words
      .
      .
      .
      RETURN
      END
```

This subroutine writes a single hour's record from array RECORD into a type 2 PF file. Note that model file type 2 is not used in ROM2.1.

1.5.3.19 Subroutine WRFT3

```
      SUBROUTINE WRFT3 (FNAME, NPARM, JDATE, IHOUR, RECORD)
      CHARACTER*12 FNAME
      DIMENSION RECORD (1001+250*NPARM)
C INPUT ARGUMENTS:
C     FNAME          Internal (logical) file name attached to Processor or Model File
C     NPARM          Number of parameters in file
C     JDATE          Julian date (YYDDD) of hour to be written
C     IHOUR          Hour (00 to 23 local standard) to be written
C     RECORD         Address of first word in common block or buffer
C                    area that contains (1001+250*NPARM) words
      .
      .
      .
      RETURN
      END
```

This subroutine writes a single hour's record from array RECORD into a type 3 PF file.

1.5.3.20 Subroutine WRFT4

```
      SUBROUTINE WRFT4 (FNAME, NCOL, NROW, NPARM, GRID)
      CHARACTER*12 FNAME
      DIMENSION GRIDA (NCOL, NROW, NPARM)
C INPUT ARGUMENTS:
C     FNAME          Internal (logical) file name attached to Processor or Model File
C     NCOL           Number of columns in grid region
C     NROW           Number of rows in grid region
C     NPARM          Number of parameters in file
C     GRIDA          Address of first word in buffer
C                    area that contains (NCOL*NROW*NPARM) words
      .
      .
      .
      RETURN
      END
```

This subroutine writes a single record from array GRIDA into a type 4 PF file.

1.5.3.21 Subroutine WRFT5

```
         SUBROUTINE WRFT5(FNAME, NCOL, NROW, NPARM, JDATE, IHOUR, GRIDA)
         CHARACTER*12 FNAME
         DIMENSION GRIDA (NCOL, NROW, NPARM)
C INPUT ARGUMENTS:
C       FNAME           Internal (logical) file name attached to Processor or Model File
C       NCOL            Number of columns in grid region
C       NROW            Number of rows in grid region
C       NPARM           Number of parameters in file
C       JDATE           Julian Date (YYDDD) of hour to be written
C       IHOUR           Hour (00 to 23 local standard) to be written
C       GRIDA           Address of first word in buffer
C                       area that contains (NCOL*NROW*NPARM) words
        .
        .
        .
        RETURN
        END
```

This subroutine writes a single hour's record from array GRIDA into a type 5 PF file.

1.5.4 A Brief Description of Other Utility and I/O Subprograms

ADATE	returns the current system date and time.
AGRID1	grids one scalar variable and produces a 2-dimensional array.
BGRID1	uses the Barnes (1973) technique to grid a given variable.
BGRID2	uses the Barnes (1973) technique to grid two variables.
BGRID3	uses the Barnes (1973) technique to grid three or more variables.
CELL0	defines a ROM grid region, and sets up a BLOCK of parameters for this region.
CELL1	returns the grid column and row numbers for given latitude and longitude coordinates.
CELL2	returns the latitude and longitude coordinates for the center of a given grid cell (column and row number inputs).
CELLAR	returns the area in m^2 of a given grid cell.
CHAIN2	inserts/retrieves a key in/from a hashing table.
CLOCK1	returns the elapsed scenario time and the step number since the start of the run.
CLOCK2	returns the current scenario date and time relative to the start of the run.
CLRHDR	sets (clears) the HOUR array in a PF/MF header to .FALSE.

CPUTOT	returns the total amount of CPU seconds since the start of the current process/job.
DATEHR	returns a list of date/hours when provided with a start/end date and interval size.
DATHR5	returns the creation and update date and time of a given file.
DATHR6	returns a list of date/hours that have been written to a given file.
DATTIM	returns a current date and time in *MMDDYY, HHMMSS* format.
ERROUT	processes an error condition when one is encountered.
FSKIP1	positions a pointer in a sequential file either forward or backward (by a given number of records).
FSTAT1	determines if a given file name is valid or invalid according to the PF/MF directory file.
GETMEM	dynamically allocates system memory for an array of given size.
GETMSG	returns the message text for a given system status code.
GETYN	prompts users with a text question that is used for input.
GMTLCL	converts Greenwich Mean Time to local time.
GPRIME	solves a chemical differential equation used in ROM.
GREG	returns the day and month for a given Julian date.
IBLKR	computes the number of blocks in a file header and/or record.
INDEX1	searches a character array for a given character string.
IOCL	extracts the clause field in a given I/O status word.
IOERR	writes I/O error information to the current job log.
JFILE1	returns the unit number attached to a given processor or model file.
JFILE4	opens and attaches a given sequential file to a given unit.
JFILE5	opens and attaches a unit number to a given fixed-length sequential file.
JFILE6	opens and attaches a unit number to a given sequential file with a 'variable' RECORDTYPE attribute.
JFILEX	opens and attaches a unit number to a given sequential file with a given record length (other than the default length).

JHOUR1	computes the hour index in a sequential list of Julian dates/hours.
JULIAN	returns the Julian date for a set of year, month, and day arguments.
KDAY	returns the day of the week (1-7) for a given year and Julian day.
LEN1	returns the number of trailing blanks in the input argument string.
MATCH	compares and determines whether two arrays are identical.
NEXTHR	increments the date and hour being processed by a given temporal interval.
POINTS	uses the Barnes (1973) technique to analyze multiple variables where the values for each variable occur at common locations.
PQ1CB42	calculates the rates of decay and production for the CBM-4.2 chemical species.
PQCOEFCB42	calculates the product [reaction rate] • [advection solution] for each reaction in CBM-4.2.
RATEDCB42	calculates the CBM-4.2 reaction rates.
RDDIR1	retrieves structure information concerning a given processor or model file from the database directory file.
RDFHD1	reads a given PF/MF header record.
RDFHD2	returns the information in a PF/MF header record.
RDFILE	reads a given number of unformatted words from a file into a buffer.
RDREC	reads one logical record from a processor or model file.
RDSRF1	reads temporally-windowed data from a surface meteorology file.
RDWNDO	reads spatially- and temporally-windowed data from a surface meteorology file.
SOLAR	computes solar elevation angles.
SOLELV	calculates the angle of the sun above/below the horizon for a specific location and time.
TOTAL	writes total nitrogen concentrations (from input N-species) to the log.
TRNSLT	translates VAX F_floating format to IBM floating point format or translates ASCII format to EBCDIC format.
WRCHAR	writes a character-string buffer to a file.

WRFILE	writes a buffer of unformatted data to a file.
WRREC	writes a logical record from a buffer to a processor or model file.
XATAN2	computes the ARCTAN for a given pair of components (in radians).

SECTION 2

THE ROM PROCESSOR NETWORK

The ROM has been programmed using the American National Standards Institute (ANSI) FORTRAN-77 full language specification. Development of the system began on the EPA's Sperry UNIVAC®1100 mainframe computer. It was then shifted to a VAX™ 11/780 minicomputer in May 1983. Currently, we are running the Core Model on the EPA-NCC's IBM®3090 and the processor network on the EPA-NCC VAX cluster, consisting of two VAX 8650s and one VAX 11/785. Running the model system requires a significant amount of CPU time; for example, for one three-day execution the ROM2.1 Core Model required about 9.5 h of IBM 3090 CPU time and the ROM2.1 processor network used about 12 h of VAX 8650 CPU time.

Descriptions of each processor begin in Section 2.1; the processors are documented in numerical order. Each processor description (1) defines what the program does, (2) shows input and output file parameters, and lists READ, WRITE, and FORMAT statements, (3) shows and defines control card variables, (4) summarizes the computer resources required, (5) lists the link stream commands, (6) lists the run stream commands, (7) lists all subroutines, functions, and block data used, (8) lists the I/O or utility library routines used, and (9) lists the INCLUDE files used. The documentation of the processor network is specific to the ROMNET region.

To help you identify the acronyms and terms we use in this document, we include a short glossary below:

AEROS	Aerometric and Emissions Reporting System; refer to EPA 450/2-76-005a.
DDD	Three-digit Julian day number.
episode	A model period made up of several three-day scenarios.
EST	Eastern Standard Time (GMT - 5 hours)
GMT	Greenwich Mean Time
HH	Two-digit hour on a 24-hour clock, e.g., 23 = 11 p.m.; <u>unless we stipulate otherwise, all *HH* hours are EST.</u>
Julian	Cumulative day-number calendar; 001 = January 1, 365 = December 30 or 31.
MM	Two-digit month, e.g., 10 represents October; or Two-digit minutes in *HHMMSS* string.
MMM	Three-letter month, e.g., AUG represents August.
MSL	Mean Sea Level
scenario	A model period starting at 1200 EST and continuing for 72 hours.
SS	Two-digit seconds.
YY	Two-digit year, e.g., 85 represents the year 1985.
Z	"Zulu" time, i.e., GMT; used as a suffix on the time of meteorological observations.

2.1 PROCESSOR P01G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 2 processor produces four sets of profiles of meteorological parameters at vertical intervals of 50 m; these are:

- *u* and *v* wind components vs. height;
- dew point temperature, virtual temperature, mixing ratio, and relative humidity vs. height;
- pressure vs. height;
- potential temperature, density, and static stability vs. height.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

   PF111 from P16G

2. Nonstandard Input Files

   None

B. Output Files

1. Standard Output Files

   a. PF101

   This file contains rawinsonde vertical profiles interpolated to hourly time intervals, and interpolated vertically to every 50 m. PF101 is used by P07G and P11G, and its parameters are shown in Table 2.1.

TABLE 2.1. PF101 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	Zm_h	m above MSL	Real*4	Interpolation height for parameters
2	$\hat{}um_{z,h}$	$m \cdot s^{-1}$	Real*4	West-to-east wind component
3	$\hat{}vm_{z,h}$	$m \cdot s^{-1}$	Real*4	South-to-north wind component

Note: *m* = upper-air station number; *z* = height, *h* = hour.

b. PF102

This file contains rawinsonde vertical profiles interpolated to hourly time intervals, and interpolated vertically to every 50 m. PF102 is used by P09G, and its parameters are shown in Table 2.2.

TABLE 2.2. PF102 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	Zm_h	m above MSL	Real*4	Interpolation height for parameters
2	$Tdm_{z,h}$	K	Real*4	Dew point temperature
3	$Tvm_{z,h}$	K	Real*4	Virtual temperature
4	$qm_{z,h}$		Real*4	Fractional mixing ratio
5	$RHm_{z,h}$		Real*4	Fractional relative humidity

Note: m = upper-air station number; z = height, h = hour.

c. PF104

This file contains rawinsonde upper-air vertical profiles interpolated to hourly time intervals, and interpolated vertically to every 50 m. PF104 is used by P09G and P11G, and its parameters are shown in Table 2.3.

TABLE 2.3. PF104 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	Zm_h	m above MSL	Real*4	Interpolation height for parameters
2	$Pm_{z,h}$	mb	Real*4	Atmospheric pressure level

Note: m = upper-air station number; z = height, h = hour.

d. PF131

This file contains rawinsonde upper-air vertical profiles interpolated to hourly time intervals, and interpolated vertically to every 50 m. PF131 is used by P09G, and its parameters are shown in Table 2.4.

TABLE 2.4. PF131 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	Zm_h	m above MSL	Real*4	Interpolation height for parameters
2	$Hm_{z,h}$	K	Real*4	Potential temperature
3	$RHOm_{z,h}$	$kg \cdot m^{-3}$	Real*4	Air density
4	$SIGMAm_{z,h}$	$m^4 \cdot s^2 \cdot kg^{-2}$	Real*4	Static stability

Note: m = upper-air station number; z = height, h = hour.

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 2.5 defines the control card variables.

JDT1 JDT2

TABLE 2.5. CONTROL CARD VARIABLES

Variable Name	Unit	Description
JDT1	EST	Julian scenario start date and time as *YYDDDHH*
JDT2	EST	Julian scenario end date and time as *YYDDDHH*

Example:

8819612 8819911

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	4	files	21	blocks
FORTRAN INCLUDE files:	3	files	15	blocks
Object files:	4	files	18	blocks
Executable file:	1	file	45	blocks
	12	files	99	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:00:30
Buffered I/O count:	282
Direct I/O count:	5003
Peak working-set size:	1240
Peak virtual size:	4411

C. Space Requirements: Log and Print Files

P01GX1.LOG:	21 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.6 shows the input file and output file space requirements.

TABLE 2.6. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF111	1	8904	72 hours
Output	PF101	1	6465	72 hours
	PF102	1	8904	72 hours
	PF104	1	4347	72 hours
	PF131	1	7623	72 hours
			27339	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P01GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P01GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P01GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P01GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]P01G.OBJ          A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]RLEVEL.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]PARAMS.OBJ        A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]PSHIFT.OBJ        A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ         A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ        A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ        A7
```

```
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ        A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ         A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ       A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ       A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ        A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]NEXTHR.OBJ       A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ       A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ        A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ       A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ       A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT1.OBJ        A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ        A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT1.OBJ        A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ        A21
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P01GV1A.EXE -
  /MAP = MET1:[S01S]P01GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A19, -
  A20, -
  A21
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P01GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P01GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P01GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                    PFMFDIR
$ ASSIGN MET56:[ROMNET1.D85188]PF111A.DAT            PF111
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF101A.DAT            PF101
$ ASSIGN MET56:[ROMNET1.D85188]PF102A.DAT            PF102
$ ASSIGN MET56:[ROMNET1.D85188]PF104A.DAT            PF104
$ ASSIGN MET56:[ROMNET1.D85188]PF131A.DAT            PF131
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P01GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P01GX1.CPR    SYS$INPUT
$ !
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P01GV1A.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P01GV1A

B. Subroutines

RLEVEL
PARAMS
PSHIFT

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	NEXTHR	RDFT1
FLSMRY	OPDIR2	RDREC
FSTAT1	OPEN2	WRFT1
IBLKR	PROGID	WRREC
JFILE1	RDDIR1	

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JHOUR1
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

INDATA.EXT
INTERN.EXT
OUTDAT.EXT

2.2 PROCESSOR P02G (CHEMISTRY)

I. PROCESSOR FUNCTION

This stage 1 processor generates ICON, the initial conditions file, which contains the "clean air" (background) layered gridded species concentrations for the modeling domain at time step zero. All cells in each layer have identical concentration values for each species.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

P02G uses one nonstandard input file output from P21G, CLEANAIR.DAT, that contains layered background species concentrations for 35 species, generated during both day (photolytic) and night (nonphotolytic) conditions. The rate constants for the input file are output by CVCHEM. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 2.7.

```
      READ (JCONC,1005) HDTEXT
      READ (JCONC,1005) HDTEXT
 1005 FORMAT (A100)
      READ (JCONC,1009) TMPVAR
 1009 FORMAT (A8)
      DO 201 LEV=1,NLEVS
         READ (JCONC,1009) TMPVAR
         READ (JCONC,1011) SPNMIC(ISPC),CONCS(ISPC,LEV)
     &         ISPC=1,NSPECS)
 1011    FORMAT (5E15.0)
  201 CONTINUE
```

TABLE 2.7. CLEANAIR.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	HDTEXT		Char*100	Header text record
2	TMPVAR		Char*8	Day/night flag or layer number
3	$SPNMIC_{L,i}$		Char*4	Species name (see Table 2.8)
4	$CONCS_{L,i}$	ppm	Real*4	Species concentration

Note: L = layer number; i = 1 (day) or 2 (night).

The 35 species used in the ROM are shown in Table 2.8.

TABLE 2.8. ROM BACKGROUND SPECIES AND REACTIONS

Symbol	Description	Symbol	Description
ALD2	High MW aldehydes	O1D	O^1D atom
C2O3	Peroxyacetyl radical	O3	Ozone
CO	Carbon monoxide	OH	Hydroxyl radical
CRES	Cresol and high MW phenols	OLE	Olefinic carbon bond
CRO	Methylphenoxy radical	OPEN	High MW aromatic oxidation ring fragment
ETH	Ethene		
FORM	Formaldehyde	PAN	Peroxyacetyl nitrate
H2O2	Hydrogen peroxide	PAR	Paraffinic carbon bond
HNO2	Nitrous acid	PNA	Peroxynitric acid
HNO3	Nitric acid	ROR	Secondary organic oxy radical
HO2	Hydroperoxy radical	TO2	Toluene-hydroxyl radical adduct
ISOP	Isoprene structures	TOL	Toluene
MGLY	Methylglyoxal	XO2	NO to NO_2 reaction
N2O5	Dinitrogen pentoxide	XO2N	NO to nitrate (NO_3^-) reaction
NO	Nitric oxide	XYL	Xylene
NO2	Nitrogen dioxide	MTHL	Methanol
NO3	Nitrogen trioxide	NONR	Nonreactive hydrocarbons
O	O^3P atom	TRAC	Tracer species

Note: MW = molecular weight

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

P02G generates ICON; this file contains the initial condition concentrations of the species shown in Table 2.8 for every grid cell and each model layer (or layer). ICON is used by the Core Model. There are up to 24 records written to the file prior to the concentration fields. These are described below.

a. Record 1

This record contains the parameters shown in Table 2.9. The parameters are first written to the buffer SEG1BF; subsequently subroutine WRCHAR writes the data (unformatted) to ICON. These steps are shown below.

```
      WRITE (SEG1BF,3001,IOSTAT=IOST)
     &      CDATIC, CTIMIC, SDATIC, STHRIC, TSTPIC, FRSTIC,
     &      GRDNIC, SWLNIC, SWLTIC, NELNIC, NELTIC, DLONIC,
     &      DLATIC, NCOLIC, NROWIC, NLEVIC, NSPCIC, (IDUM(J),J=1,8),
     &      ICNTIC
3001  FORMAT (6I8.8,A8,4F8.3,2F8.5,4I4.4,8I8.8,I4.4)

      CALL WRCHAR (IUNIT,SEG1BF,IOST)

      SUBROUTINE WRCHAR(IUNIT,CHBUF,IOST)
      IMPLICIT NONE
      INTEGER*4 IUNIT,IOST
      CHARACTER*(*) CHBUF
      WRITE(IUNIT,IOSTAT=IOST) CHBUF
      RETURN
      END
```

TABLE 2.9. ICON RECORD 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CDATIC		Integ*4	File creation date as *MMDDYY*
2	CTIMIC	EST	Integ*4	File creation time as *HHMMSS*
3	SDATIC		Integ*4	Julian scenario start date as *YYDDD*
4	STHRIC	EST	Integ*4	Scenario start time as *HH*
5	TSTPIC	s	Integ*4	Time step, simulation
6	FRSTIC	s	Integ*4	First time step in ICON
7	GRDNIC		Char*8	Grid name (see Appendix A)
8	SWLNIC	° W	Real*4	Longitude, SW corner of grid
9	SWLTIC	° N	Real*4	Latitude, SW corner of grid
10	NELNIC	° W	Real*4	Longitude, NE corner of grid
11	NELTIC	° N	Real*4	Latitude, NE corner of grid
12	DLONIC	° W	Real*4	Cell longitude increment
13	DLATIC	° N	Real*4	Cell latitude increment
14	NCOLIC		Integ*4	Number of columns in grid
15	NROWIC		Integ*4	Number of rows in grid
16	NLEVIC		Integ*4	Number of layers in simulation
17	NSPCIC		Integ*4	Number of species in ICON file
18	IDUM		Integ*4	Padding to fill buffer
19	ICNTIC		Integ*4	Number of text records

b. Record 2

This record contains the parameter shown in Table 2.10. The parameter is first written to the buffer SPNMBF; subsequently subroutine WRCHAR writes the data (unformatted) to ICON. The WRITE and FORMAT statements are listed below:

```
      WRITE (SPNMBF,3005,IOSTAT=IOST) (SPNMIC(I), I=1,NSPCIC)
 3005 FORMAT (<NSPECS>(A4))
      CALL WRCHAR (IUNIT,SPNMBF,IOST)
```

TABLE 2.10. ICON RECORD 2 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SPNMIC		Char*4	Chemical species name

c. Record 3

This record contains the parameter shown in Table 2.11. The parameter is first written to the buffer LEVNBF; subsequently subroutine WRCHAR writes the data (unformatted) to ICON. The WRITE and FORMAT statements are listed below:

```
      WRITE (LEVNBF,3007,IOSTAT=IOST)
     &      (LVNMIC(I), I=1,NLEVIC)
 3007 FORMAT (<NLEVS>(A4))
      CALL WRCHAR (IUNIT,LEVNBF,IOST)
```

TABLE 2.11. ICON RECORD 3 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	LVNMIC		Char*4	Layer number

d. Records 4 - (4 + ICNTIC)

These records contain the parameter shown in Table 2.12, which consists of the optional text that can be entered as part of P02G's input. The parameter is first written to the buffer TEXTBF from control card records 3 up to 22; subsequently, subroutine WRCHAR writes the 80-character strings to the output records in ICON. The WRITE statements are listed below:

```
C--WRITE HEADER - TEXT GROUP FOR 20 RECORDS
      DO 201, I = 1, ICNTIC
C--MOVE TEXT
      TEXTBF = TEXTIC(I)
C--WRITE
      CALL WRCHAR (IUNIT,TEXTBF,IOST)
201   CONTINUE
```

TABLE 2.12. ICON RECORDS 4 - (4 + ICNTIC) PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	TEXTIC		Char*80	Descriptive text

e. Record (5 + ICNTIC): the time step header record

There is only one time step header record because this initial concentration file contains only one time step, FRSTIC. This record contains the parameters shown in Table 2.13. The WRITE statements write the header from a COMMON declaration; these codes are listed below:

```
      REAL*4          DATIC, TIMIC, ELPIC, STPIC...
      COMMON /RTSHIC/ DATIC, TIMIC, ELPIC, STPIC...

C--WRITE TIME STEP HEADER
      CALL WRFILE (IUNIT,IWDLN2,DATIC,IOST)

      SUBROUTINE WRFILE (IUNIT, NWORDS, BUFFER, IOST)
      IMPLICIT NONE
      INTEGER*4    IUNIT, NWORDS, IOST
      REAL*4       BUFFER (NWORDS)
      WRITE (IUNIT, IOSTAT = IOST) BUFFER
      RETURN
      END
```

TABLE 2.13. ICON TIME STEP HEADER RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATIC		Real*4	Julian date on ICON file as *YYDDD*
2	TIMIC	EST	Real*4	Time on ICON file as *HHMMSS*
3	ELPIC	s	Real*4	Elapsed time since start
4	STPIC		Real*4	Step number (always = 1)

f. Records (6 + ICNTIC) - (NROWS x NSPECIES): species concentration data

These records contain the species concentrations, iterated over rows, that have the parameter shown in Table 2.14. The value of the IWDLN2 variable shown in the WRITE statements below is the product of the number of columns in the modeling domain and the number of model layers. IWDLN2 defines the buffer length.

```
C--WRITE ROW TO ICON
C--LOOP OVER SPECIES
      DO 301, ISPC = 1, NSPECS
      CALL WRFILE (IUNIT,IWDLN2,ICFILE(1,1,ISPC),IOST)
301   CONTINUE
```

TABLE 2.14. ICON SPECIES CONCENTRATIONS PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$ICFILE_{i,L,s}$	ppm	Real*4	Initial species concentrations

Note: i = column number; L = layer number; s = species, ordered as in Table 2.8.

III. CONTROL CARDS

Two control cards are used, in the format shown below. Table 2.15 describes the control card variables. Following these two control cards, optional text information can be entered in as many as 20 records. This text will be included in the ICON file's header.

SDATIC STHRIC
IDATIC ITIMIC
VAXFMT

TABLE 2.15. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Description
1	SDATIC		Julian scenario start date as *YYDDD*
2	STHRIC	EST	Scenario start hour as *HH*
3	IDATIC		Julian date of initial conditions as *YYDDD*
4	ITIMIC	EST	Time of initial conditions as *HHMMSS*
5	VAXFMT		Translate flag: .TRUE. ⇒ do not translate to IBM binary .FALSE. ⇒ translate to IBM binary

Example:

88236, 12

88236, 120000

.FALSE.

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	4	files	39	blocks
FORTRAN INCLUDE files:	11	files	36	blocks
Object files:	4	files	24	blocks
Executable file:	1	file	42	blocks
	20	files	141	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:00:14
Buffered I/O count:	227
Direct I/O count:	432
Peak working-set size:	506
Peak virtual size:	2514

C. Space Requirements: Log and Print Files

P02GX1.LOG:	15 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.16 shows the input file and output file space requirements.

TABLE 2.16. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	CLEANAIR	Nonstd	11	N/A
Output	ICON	Nonstd	2739	N/A

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P02GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P02GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P02GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMCOM.ROMNET]P02GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P02G.OBJ              A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]BLKIC.OBJ             A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]OPWRIC.OBJ            A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRIC.OBJ              A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ             A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ             A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CLOCK1.OBJ            A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATTIM.OBJ            A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ              A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ            A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE5.OBJ            A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ             A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ            A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRCHAR.OBJ            A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFILE.OBJ            A15
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMCOM.ROMNET]P02GV1A.EXE -
  /MAP = MET1:[S01S]P02GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A13, -
  A14, -
  A15
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P02GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P02GX1.COM follows; note that the file-naming conventions (for the ICON file) can be found in the Tutorial (Section 4) of Part 3 of this User's Guide.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! Run stream is in file DISK$VAXSET:[FOREMAN.FV1502A]P02GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ !
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                    PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]CLEANAIR.DAT           CLEANAIR
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET31:[FOREMAN.ICBCS]ICFV1502A.DAT          ICON
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARDS FILE
$ TYPE DISK$VAXSET:[FOREMAN.FV1502A]P02GX1.CPR
$ ASSIGN / USER_MODE DISK$VAXSET:[FOREMAN.FV1502A]P02GX1.CPR   SYS$INPUT
$ !
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P02GV1A.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P02GV1A

B. Subroutines

OPWRIC
WRIC

C. Functions

None

D. Block Data Files

BLKIC

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	PROGID
CELL0	WRCHAR
CLOCK1	WRFILE
DATTIM	

B. Functions

IOCL
JFILE2
JFILES
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

FLNAMS.EXT	LVNAME.EXT
GRDDEF.EXT	PARAM.EXT
HDFMTS.EXT	PRID.EXT
HEADIC.EXT	REGION.EXT
ICFILE.EXT	TSHDIC.EXT
LUNITS.EXT	

2.3 PROCESSOR P03G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 1 processor computes (1) the meteorological parameters used in later stages of the ROM from the observed surface data, and (2) the presence or absence of local nocturnal temperature inversions.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

PF195 from preprocessor MASK

2. Nonstandard Input Files

a. SURMET1

SURMET1, the final output from the meteorology preprocessors, contains hourly surface meteorology observations from the National Climatic Data Center and the Canadian Climate Centre raw data tapes. SURMET1 is read in two stages. The first stage reads header information. READ and FORMAT statements for SURMET1 are called by P03G from RDSRF1 in the UTILIO library. Table 2.17 shows the SURMET1 Stage 1 parameters.

TABLE 2.17. SURMET1 STAGE 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	INYR		Integer*4	Scenario year (*YY*)
2	INMON		Integer*4	Scenario month (*MM*)
3	INDAY		Integer*4	Scenario day (*DD*)
4	INHOUR	GMT	Integer*4	Scenario hour (*HH*)
5	NSTAT		Integer*4	No. of reporting stations at *HH*

The Stage 2 parameters that are used in P03G are described in Table 2.18:

TABLE 2.18. SURMET1 STAGE 2 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1a	$STAID_n$		Char*4	Station ID (non-WBAN format)
1b	$STAID_n$		Integer*4	Station ID (WBAN format)
2	$SLATS_n$	° N	Real*4	Latitude
3	$SLONS_n$	° W	Real*4	Longitude
4	$OPQCOV_n$	%	Real*4	Opaque sky cover
6	$COVER1_n$	%	Real*4	Coverage, lowest cloud layer
7	$BASE1_n$	m	Real*4	Lowest cloud base
8	$COVER2_n$	%	Real*4	Coverage, second-lowest cloud layer
9	$BASE2_n$	m	Real*4	Second-lowest cloud base
10	$COVER3_n$	%	Real*4	Coverage, third-lowest cloud layer
11	$BASE3_n$	m	Real*4	Third-lowest cloud base
12	$P0_n$	mb	Real*4	Sea level pressure
13	DIR_n	deg	Real*4	Wind direction
14	SPD_n	$m{\cdot}s^{-1}$	Real*4	Wind speed
15	T_n	°C	Real*4	Temperature
16	TD_n	°C	Real*4	Dew point
17	P_n	mb	Real*4	Station pressure

Note: n = surface station number (one or more digits).

b. SURELEV1

We manually created this file, which contains the elevation above mean sea level of all stations reporting surface meteorology.[1] The following READ and FORMAT statements pertain to non-WBAN and to WBAN station identification respectively. Table 2.19 shows the file's parameters.

```
        READ  ( IF1 , 1001 , IOSTAT  =  IOST ) ISTNBR, LAT, LON, IELE
        READ  ( IF1 , 1000 , IOSTAT  =  IOST ) STID, ISTNBR, LAT, LON, IELE
1000    FORMAT ( A4 , 1X , I5 , F6.2 , F7.2 , I5 )
1001    FORMAT ( 1X , I5.5 , 2X , F5.2 , 2X , F6.2 , 1X , I5 )
```

1. You can create a similar elevations list for other stations from a number of sources. We suggest that you refer to the WBAN document from the NCDC; for stations without WBAN numbers that are airports, the U.S. Federal Aviation Administration (FAA, Oklahoma City, OK) publishes regional guides to airports in the U.S. Each airport's altitude is listed. For airports in Canada or Mexico, elevation information can be obtained from FAA Sectional Charts.

TABLE 2.19. SURELEV1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1a	$STAID_n$		Char*4	Station ID (non-WBAN format)
1b	$STAID_n$		Integer*4	Station ID (WBAN format)
2	$SLAT_n$	° N	Real*4	Latitude
3	$SLON_n$	° W	Real*4	Longitude
4	$ELEV_n$	m above MSL	Real*4	Station elevation

Note: n = surface station number (one or more digits).

B. Output Files

1. Standard Output Files

a. PF106

This file contains hourly surface meteorology data for the modeling domain plus a 6°-border. It is required by P04G, P07G, P10G, P11G, P15G, P32G, and P33G. Table 2.20 shows its parameters.

TABLE 2.20. PF106 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$\hat{u}_n$	$m \cdot s^{-1}$	Real*4	West-to-east wind component
2	$\hat{v}_n$	$m \cdot s^{-1}$	Real*4	South-to-north wind component
3	q_n		Real*4	Mixing ratio of water vapor
4	RH_n		Real*4	Relative humidity
5	Td_n	°C	Real*4	Dew point temperature
6	Tv_n	°C	Real*4	Virtual Temperature
7	$PHIO_n$	$m^2 \cdot s^{-2}$	Real*4	1000 mb pressure surface geopotential
8	$.PHIO_n$	$m^2 \cdot s^{-3}$	Real*4	1000 mb geopotential time derivative
9	psl_n	mb	Real*4	Sea level pressure
10	RHO_n	$kg \cdot m^{-3}$	Real*4	Air density
11	H_n	K	Real*4	Potential temperature

Note: n = surface station number (one or more digits).

b. PF112

PF112 is the inversion-indicator file. Inversions are determined from an *ad hoc* method: the parameter shown in Table 2.21 below is calculated from gridded hourly nocturnal surface temperatures minus a weighted three-hour average centered at 1900 LST, plus an empirically-determined SPDFAC term (defined under Control Cards) that multiplies the gridded wind speed; finally, the term is normalized to zero. Note that in the ROM Version 2.1, the parameter DRHO1 is not actually used in any further calculations. Instead, if the HOURS logical in the file header is retained by the UTILIO function DATHR3 as "TRUE", then an inversion exists for the hour for which the logical is reporting. PF112 is used in P07G, P11G, and P12G.

TABLE 2.21. PF112 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$DRHO1_{i,j}$		Real*4	An inversion exists if $DRHO1_{i,j} < 0.0$

Note: i = column number, j = row number

2. Nonstandard Output Files

None

III. CONTROL CARDS

Six control cards are used to input control data in the format shown below. Table 2.22 defines the control card variables. Note that the ROM Version 2.1 does not use grid-based inversion parameters; the inversion either exists or it does not. However, the control cards for a subsequent version's grid-based inversion calculations have been included in P03G.

JD1 JD2
GDNAME
WL
SPDFAC
FRAC
INTFLAG

TABLE 2.22. CONTROL CARD VARIABLES

Variable Name	Unit	Description
JD1	EST	Julian scenario start date and time as *YYDDDHH*
JD2	EST	Julian scenario end date and time as *YYDDDHH*
GDNAME		Modeling domain name (see Appendix A)
WL		Barnes' inversion filtering wavelength[2]
SPDFAC		Weight of wind speed term
FRAC		Fraction of domain grid land area: ≥FRAC - inversion exists; <FRAC - no inversion
INTFLAG		Use of WBAN-format station identifiers: Y - WBAN format used; N - WBAN format not used

Note: WL and SPDFAC result in data that are not subsequently used in this version of the ROM.

Example:

8819612 8819911
ROMNET1
600.0
2.0
0.60
Y

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	3	files	69	blocks
FORTRAN INCLUDE files:	5	files	15	blocks
Object files:	3	files	21	blocks
Executable file:	1	file	57	blocks
	12	files	162	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:09:08
Buffered I/O count:	307
Direct I/O count:	1765
Peak working-set size:	1181
Peak virtual size:	4400

2. Barnes (1973)

C. Space Requirements: Log and Print Files

P03GX1.LOG:	46 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.23 shows the input file and output file space requirements.

TABLE 2.23. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF195	4	55	72 hours
	SURELEV1	Nonstd	33	N/A
	SURMET1	Nonstd	40251	1 month
			40339	
Output	PF106	3	2142	72 hours
	PF112	5	1899	72 hours
			4041	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P03GV1B.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P03GV1B.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P03GV1B.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P03GV1B.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P03G.OBJ              A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]HASH.OBJ              A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]STINFO.OBJ            A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ             A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]BGRID1.OBJ            A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELLO.OBJ             A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CHAIN2.OBJ            A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CLRHDR.OBJ            A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ            A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ            A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GETYN.OBJ             A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GMTLCL.OBJ            A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ             A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ              A14
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOERR.OBJ             A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JHOUR1.OBJ            A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ             A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ            A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE2.OBJ            A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JULIAN.OBJ            A20
```

```
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]MATCH.OBJ                      A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]NEXTHR.OBJ                     A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ                     A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ                      A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ                     A25
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ                     A26
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT4.OBJ                      A27
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ                      A28
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDSRF1.OBJ                     A29
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDWNDO.OBJ                     A30
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT3.OBJ                      A31
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT5.OBJ                      A32
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ                      A33
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P03GV1B.EXE -
  /MAP = MET1:[S01S]P03GV1B.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A31, -
  A32, -
  A33
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P03GV1B.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P03GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P03GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                                PFMFDIR
$ ASSIGN MET25:[FOREMAN.RAWDATA]SURELEV_WBAN.DAT                 SURELEV1
$ ASSIGN MET23:[FOREMAN.RAWDATA]SRFC_JUL85.DAT                   SURMET1
$ ASSIGN MET1:[ROMNET1.D00000]PF195A.DAT                         PF195
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF106A.DAT                        PF106
$ ASSIGN MET56:[ROMNET1.D85188]PF112A.DAT                        PF112
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARDS FILE
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P03GX1.CPR
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P03GX1.CPR     SYS$INPUT
$ !
$ RUN $2$DUA24:[ROMEXE.ROMNET]P03GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A        P03GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P03GV1B

B. Subroutines

HASH
STINFO

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	FSTAT1	OPEN2	RDWNDO
BGRID1	GMTLCL	PROGID	WRTF3
CELL0	IBLKR	RDDIR1	WRFT5
CHAIN2	IOERR	RDFT4	WRREC
CLRHDR	NEXTHR	RDREC	
FLSMRY	OPDIR2	RDSRF1	

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETYN	JULIAN
IOCL	JUNIT
JFILE2	MATCH
JHOUR1	

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

HASHT.EXT
PARM3.EXT
PARMS.EXT
SURMET.EXT

2.4 PROCESSOR P04G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 2 processor computes the gridded (1) micrometeorological and meteorological parameters that are used in later stages of the ROM from the observed surface, terrain, and land use data sets, (2) surface sensible heat flux, and (3) sea surface temperatures (which are used for quality control purposes only).

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

PF106 from P03G
PF117 from P19G
PF118 from P25G
PF119 from P06G

2. Nonstandard Input Files

a. BUOYDAT

BUOYDAT contains hourly buoy data, and is generated by meteorology preprocessor BUOY_RFMT. BUOYDAT, whose format mimics that of the SURMET1 data file, is read in two stages. The first stage reads header information. READ and FORMAT statements for BUOYDAT are called by P04G from RDSRF1 in the UTILIO library. Table 2.24 shows the BUOYDAT Stage 1 parameters.

TABLE 2.24. BUOYDAT STAGE 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	INYR		Integer*4	Scenario year (*YY*)
2	INMON		Integer*4	Scenario month (*MM*)
3	INDAY		Integer*4	Scenario day (*DD*)
4	INHOUR	GMT	Integer*4	Scenario hour (*HH*)
5	NBUOY		Integer*4	No. of reporting buoys at *HH*

The Stage 2 parameters that are used in P04G are described in Table 2.25:

TABLE 2.25. BUOYDAT STAGE 2 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$BUOYID_n$		Integer*4	Buoy ID
2	LAT_n	° N	Real*4	Latitude
3	LON_n	° W	Real*4	Longitude
4	none	none	Real*4	Missing
6	none	none	Real*4	Missing
7	none	none	Real*4	Missing
8	none	none	Real*4	Missing
9	none	none	Real*4	Missing
10	none	none	Real*4	Missing
11	none	none	Real*4	Missing
12	$P0_{n,h}$	mb	Real*4	Sea level pressure
13	$DIR_{n,h}$	deg	Real*4	Wind direction
14	$SPD_{n,h}$	$m{\cdot}s^{-1}$	Real*4	Wind speed
15	$T_{n,h}$	°C	Real*4	Air temperature (AT)
16	$SST_{n,h}$	°C	Real*4	Sea surface temperature (SST)
17	$DELT_{n,h}$	°C	Real*4	SST - AT

Note: n = buoy station number (one or more digits); h = hour; missing values are either -99. or -9999.

b. SURMETM

SURMETM contains monthly buoy data, and is generated by meteorology preprocessor BUOY_RFMT followed by human adjustment. Note that you must assign to this file the date and time that are the first hour, in GMT, of the first day of the month under consideration; e.g., 00 GMT on Julian date 182 corresponds with 19(00) EST on day 181. SURMETM, whose format mimics that of the SURMET1 data file, is read in two stages. The first stage reads header information. READ and FORMAT statements for SURMETM are called by P04G from RDSRF1 in the UTILIO library. Table 2.26 shows the SURMETM Stage 1 parameters.

TABLE 2.26. SURMETM STAGE 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	INYR		Integer*4	Scenario year (*YY*)
2	INMON		Integer*4	Scenario month (*MM*)
3	INDAY		Integer*4	Scenario day (= 01)
4	INHOUR	GMT	Integer*4	Scenario hour (= 00)
5	NSTAT		Integer*4	No. of reporting stations at hour 00

The Stage 2 parameters that are used in P04G are described in Table 2.27:

TABLE 2.27. SURMETM STAGE 2 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$BUOYID_n$		Integer*4	Buoy ID
2	LAT_n	° N	Real*4	Latitude
3	LON_n	° W	Real*4	Longitude
4	none	none	Real*4	Missing
6	none	none	Real*4	Missing
7	none	none	Real*4	Missing
8	none	none	Real*4	Missing
9	none	none	Real*4	Missing
10	none	none	Real*4	Missing
11	none	none	Real*4	Missing
12	none	none	Real*4	Missing
13	none	none	Real*4	Missing
14	none	none	Real*4	Missing
15	$AVGT_{n,MM}$	°C	Real*4	Monthly average air temperature (AT)
16	$AVGSST_{n,MM}$	°C	Real*4	Monthly average sea surface temperature (SST)
17	$ADELT_{n,MM}$	°C	Real*4	Monthly average difference, SST - AT

Note: *n* = buoy station number (one or more digits); *MM* = month; missing values are either -99. or -9999.

B. Output Files

1. Standard Output Files

a. PF103

This file contains hourly gridded surface air temperatures, relative humidity, and wind speed; it is used by P15G and P27G, and Table 2.28 shows its parameters.

TABLE 2.28. PF103 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$SURTMP_{i,j,h}$	K	Real*4	Surface air temperature
2	$SURFRH_{i,j,h}$	%	Real*4	Surface relative humidity
3	$SURWND_{i,j,h}$	$m \cdot s^{-1}$	Real*4	Surface wind speed

Note: *i* = column number, *j* = row number, *h* = hour

b. PF107

This file contains the hourly gridded Monin-Obukhov length, a measure of the relative importance of convective turbulence versus mechanical turbulence. Thus, the Monin-Obukhov length is an indicator of atmospheric stability. PF107 is used by P07G, P10G, P15G, P32G, and P33G, and Table 2.29 shows its parameter.

TABLE 2.29. PF107 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$L_{i,j,h}$	m	Real*4	Monin-Obukhov length

Note: i = column number, j = row number, h = hour

c. PF108

This file contains the gridded surface roughness length, an index of the mechanical turbulence that can be generated in air flowing over a rough surface. PF108 is used by P15G, and Table 2.30 shows its parameter.

TABLE 2.30. PF108 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$Z0_{i,j}$	m	Real*4	Surface roughness length

Note: i = column number, j = row number

d. PF109

This file contains the hourly gridded friction velocity. The friction velocity can be regarded as an index of surface momentum flux. PF109 is used by P15G and P32G, and Table 2.31 shows its parameter.

TABLE 2.31. PF109 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$u_{*i,j,h}$	$m \cdot s^{-1}$	Real*4	Friction velocity

Note: i = column number, j = row number, h = hour

e. PF110

This file contains the hourly gridded surface kinematic heat flux, which can be interpreted as $\overline{w'\theta'}$. PF110 is used by P07G and P32G, and Table 2.32 shows its parameter.

TABLE 2.32. PF110 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$Q_{i,j,h}$	$m \cdot K \cdot s^{-1}$	Real*4	Surface kinematic heat flux

Note: i = column number, j = row number, h = hour

f. PF193

This file contains the hourly gridded buoy weight influence function and sea surface temperature. It is used solely for quality control purposes, and Table 2.33 shows its parameters.

TABLE 2.33. PF193 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$BWGHT_{i,j,h}$	m	Real*4	Buoy weight influence function
2	$SST_{i,j,h}$	°C	Real*4	Sea surface temperature

Note: i = column number, j = row number, h = hour

2. Nonstandard Output Files

None

III. CONTROL CARDS

Six control cards are used to input control data in the format shown below. Note that (1) the fifth control card delimits the bands around the modeling domain that are to be included in the input data fields, and (2) the parameter in the sixth control card is repeated; this is because subroutine RDSRF1 requires an end time to be available in the files it reads. Table 2.34 defines the control card variables.

```
YYDD1 H1 C1
GDNAME
RDBOPT
```

XL
STHBORD, WSTBORD, NTHBORD, ESTBORD
JDAVG, JDAVG

TABLE 2.34. CONTROL CARD VARIABLES

Variable Name	Unit	Description
YYDD1		Julian scenario start date as *YYDDD*
H1	EST	Scenario start time as *HH*
C1	h	Number of hours to process
GDNAME		Modeling domain name (see Appendix A)
RDBOPT		Buoy data availability BUOY ⇒ buoy data are available NOBUOY ⇒ buoy data are unavailable
XL		Buoy influence if buoy data are available (see note)
STHBORD	° N	Southern boundary extension for input data
WSTBORD	° W	Western boundary extension for input data
NTHBORD	° N	Northern boundary extension for input data
ESTBORD	° W	Eastern boundary extension for input data
JDAVG	GMT	Date and hour of first hour of month being processed (Julian) as *YYDDDHH*

Note: Gridded fields of U_* and Q over water are computed from buoy data in a two-step process: (1) a field is determined from a Barnes interpolation of all buoy values; (2) a second field is determined by assigning grid values in the vicinity of buoys to be influenced by the buoy as a function of the negative exponent of the distance of the cell from the buoy. The parameter XL adjusts the radius of influence of the buoys in this second field; as XL increases, so do the buoys' radius of influence, while at XL = 0.01, a buoy's influence is effectively zero. XL interacts with the cell weighting parameter BWGHT in this way: assume that the cell containing a buoy has BWGHT = 1.0 ($0.0 \leq$ BWGHT ≤ 1.0); at a small XL, only this cell has a BWGHT = 1.0, while surrounding cells have BWGHT's assigned from the negative exponential function. As XL increases, more cells surrounding the buoy take on a BWGHT = 1.0. If you imagine a buoy's influence on surrounding cells to consist of choropleths of concentric annuli, then an increase in XL consistently increases the outer radius of each annulus. The final gridded fields of u_* and Q are derived from the sum of field (2), with the BWGHT of cells assigned according to the value of XL, and [(1.0 - BWGHT) · (field (1))]. Note that we have found that XL = 1.0 produces fields that appear meteorologically coherent.

Example:

88184 12 72
ROMNET1
BUOY
1.0
4.0 5.0 4.0 5.0
8818219 8818219

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	24	files	192	blocks
FORTRAN INCLUDE files:	15	files	54	blocks
Object files:	24	files	105	blocks
Executable file:	1	file	141	blocks
	64	files	492	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:39:46
Buffered I/O count:	373
Direct I/O count:	4723
Peak working-set size:	2236
Peak virtual size:	4844

C. Space Requirements: Log and Print Files

P04GX1.LOG:	177 blocks
Print Files:	21 blocks

D. Space Requirements: Input and Output Files

Table 2.35 shows the input file and output file space requirements.

TABLE 2.35. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF106	3	2142	72 hours
	PF117	5	1899	72 hours
	PF118	4	286	N/A
	PF119	4	55	N/A
	BUOYDAT	Nonstd	28266	4 months
	SURMETM	Nonstd	176	29 months
			32824	
Output	PF103	5	5694	72 hours
	PF107	5	1899	72 hours
	PF108	4	54	72 hours
	PF109	5	1899	72 hours
	PF110	5	1899	72 hours
	PF193	5	3798	72 hours
			15243	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P04GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P04GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P04GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P04GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P04G.OBJ          A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]AVGWAT.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]BUOYPR.OBJ        A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]CLDFN.OBJ         A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]CRTELV.OBJ        A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]FILLIN.OBJ        A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]GAMOVS.OBJ        A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]GRBRNS.OBJ        A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]GRBUOY.OBJ        A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]GRIDGN.OBJ        A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]ISLAND.OBJ        A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]PSIM.OBJ          A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]QFLUX.OBJ         A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RADLWV.OBJ        A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDSRFM.OBJ        A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDWNDM.OBJ        A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]SETUP.OBJ         A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]STABLE.OBJ        A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]STAWAT.OBJ        A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]TSCALE.OBJ        A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]UNSTBL.OBJ        A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]UNSWAT.OBJ        A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WEIGHT.OBJ        A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]ZONEIT.OBJ        A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ         A25
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]BGRID1.OBJ        A26
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]BGRID3.OBJ        A27
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL0.OBJ         A28
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL1.OBJ         A29
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL2.OBJ         A30
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ        A31
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ        A32
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GETYN.OBJ         A33
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GMTLCL.OBJ        A34
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GREG.OBJ          A35
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ         A36
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ          A37
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOERR.OBJ         A38
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ        A39
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ        A40
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE3.OBJ        A41
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ        A42
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JULIAN.OBJ        A43
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ         A44
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]NEXTHR.OBJ        A45
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ        A46
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ         A47
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ        A48
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ        A49
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT3.OBJ         A50
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT4.OBJ         A51
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ         A52
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ         A53
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDSRF1.OBJ        A54
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDWNDO.OBJ        A55
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]SOLELV.OBJ        A56
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT4.OBJ         A57
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT5.OBJ         A58
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ         A59
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]XATAN2.OBJ        A60
```

```
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P04GV1A.EXE -
  /MAP = MET1:[S01S]P04GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A58, -
  A59, -
  A60
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P04GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P04GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P04GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                       PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]BUOY_AVGALL.DAT           SURMETM
$ ASSIGN MET25:[FOREMAN.RAWDATA]BUOY_JULAUG85.DAT       SURMET1
$ ASSIGN MET56:[ROMNET1.D85188]PF106A.DAT               PF106
$ ASSIGN MET56:[ROMNET1.D85188]PF117A.DAT               PF117
$ ASSIGN MET1:[ROMNET1.D00000]PF118A.DAT                PF118
$ ASSIGN MET1:[ROMNET1.00000]PF119A.DAT                 PF119
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF103A.DAT               PF103
$ ASSIGN MET56:[ROMNET1.D85188]PF107A.DAT               PF107
$ ASSIGN MET56:[ROMNET1.D85188]PF108A.DAT               PF108
$ ASSIGN MET56:[ROMNET1.D85188]PF109A.DAT               PF109
$ ASSIGN MET56:[ROMNET1.D85188]PF110A.DAT               PF110
$ ASSIGN MET56:[ROMNET1.D85188]PF193A.DAT               PF193
$ ASSIGN MET56:[ROMNET1.D85188]P04PRTFLA.DAT            P04GPRF
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P04GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P04GX1.CPR    SYS$INPUT
$ !
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P04GV1A.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P04GV1A

B. Subroutines

AVGWAT	GRBUOY	RDWNDM	UNSTBL
BUOYPR	GRIDGN	SETUP	UNSWAT
CRTELV	ISLAND	STABLE	WEIGHT
FILLIN	QFLUX	STAWAT	ZONEIT
GRBRNS	RDSRFM	TSCALE	

C. Functions

CLDFN
GAMOVS
PSIM
RADLWV

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	FSTAT1	OPDIR2	RDREC
BGRID1	GMTLCL	OPEN2	RDSRF1
BGRID3	GREG	PROGID	RDWNDO
CELL0	IBLKR	RDDIR1	SOLELV
CELL1	IOERR	RDFT3	WRTF4
CELL2	JFILE1	RDFT4	WRFT5
FLSMRY	NEXTHR	RDFT5	WRREC

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETYN	JULIAN
IOCL	JUNIT
JFILE2	XATAN2
JFILE3	

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

BUODAT.EXT	FLXPRM.EXT	PF6.EXT	WATGRD.EXT
BUOGRD.EXT	LNDUSE.EXT	RADVAR.EXT	WGHTFN.EXT
BUOPRM.EXT	METVAR.EXT	STBVAR.EXT	Z0.EXT
CLDCVR.EXT	PARM4.EXT	STUNST.EXT	

2.5 PROCESSOR P05G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 2 processor computes (1) the grid cell fraction of the sky that is covered by cumulus clouds, and (2) the cumulus cloud top heights. For the sky cover fraction, P05G determines whether the clouds reported at surface stations are cumulus, and uses Barnes' analysis for the grid cell interpolation.[3] Cloud tops are calculated to be at a height where the temperature of a cloud falls below the temperature of the surrounding environment; upper-air data interpolation is used for determining the environmental temperature.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

PF111 from P16G
PF119 from P06G

2. Nonstandard Input Files

P05G uses one nonstandard input file, SURMET1, that contains hourly surface meteorology observations from the National Climatic Data Center and the Canadian Climate Centre raw data tapes. SURMET1 is read in two stages. The first stage reads header information, and the first READ and FORMAT statements for this file are as follows:

```
      READ (12,30) INYR,INMON,INDAY,INHOUR,NSTAT
30    FORMAT (4I2,5X,I3)
```

Table 2.36 shows the SURMET1 Stage 1 parameters.

TABLE 2.36. SURMET1 STAGE 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	INYR		Integer*4	Scenario year (*YY*)
2	INMON		Integer*4	Scenario month (*MM*)
3	INDAY		Integer*4	Scenario day (*DD*)
4	INHOUR	GMT	Integer*4	Scenario hour (*HH*)
5	NSTAT		Integer*4	No. of reporting stations at *HH*

3. Barnes (1973)

The Stage 2 READ and FORMAT statements are listed below, and the Stage 2 parameters that are used in P05G are described in Table 2.37:[4]

```
          READ (LU,93003,IOSTAT=IOST) PARMS
    93003 FORMAT (A4,2F7.2,F5.0,F4.0, 3(F5.0,F6.0,
         &        F8.1,4F7.1,F8.1)
```

The READ and FORMAT statements above are for non-WBAN format for the surface meteorology stations. For data in WBAN format, these statements read as follows:

```
          READ (LU,93004,IOSTAT=IOST) (PTEMP(N),N=2,NPARM)
    93004 FORMAT (I5.5,F6.2,F7.2,F5.0,F4.0, 3(F5.0,F6.0,
         &        F8.1,4F7.1,F8.1)
          DO 2110 N=1,NPARM
             PARMS(N)=PTEMP(N)
     2110 CONTINUE
```

TABLE 2.37. SURMET1 STAGE 2 PARAMETERS USED IN P05G

Parm No.	Parm Name	Unit	Data Type	Description
1a	SID_n		Char*4	Station ID (non-WBAN format)
1b	SID_n		Integer*4	Station ID (WBAN format)
2	YS_n	° N	Real*4	Latitude
3	XS_n	° W	Real*4	Longitude
4	$SIGC_n$	%	Real*4	Opaque sky cover
6	CA_{1n}	%	Real*4	Coverage, lowest cloud layer
7	CH_{1n}	m	Real*4	Lowest cloud base
8	CA_{2n}	%	Real*4	Coverage, second-lowest cloud layer
9	CH_{2n}	m	Real*4	Second-lowest cloud base
10	CA_{3n}	%	Real*4	Coverage, third-lowest cloud layer
11	CH_{3n}	m	Real*4	Third-lowest cloud base
15	T_n	°C	Real*4	Temperature
16	TD_n	°C	Real*4	Dew point
17	P_n	mb	Real*4	Station pressure

Note: n = surface station number (one or more digits).

B. Output Files

1. Standard Output Files

a. PF133

This file contains hourly data on the standard 64-cell by 52-cell grid; Table 2.38 shows its parameter. PF133 is not required by any other processor.

4. Note that not all 17 parameters in PARMS are used in P05G.

TABLE 2.38. PF133 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	Ztcu	m above MSL	Real*4	Height of cumulus cloud tops

b. PF157

This file contains hourly data on the standard 64-cell by 52-cell grid; Table 2.39 shows its parameter. PF157 is used by P12G and P32G.

TABLE 2.39. PF157 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SIGMAc		Real*4	Fraction of sky covered by cumulus clouds

2. Nonstandard Output Files

None

III. CONTROL CARDS

Two control cards are used to input control data in the format shown below. Table 2.40 defines the control card variables.

JD1 JD2 ICHK
INTFLAG

TABLE 2.40. CONTROL CARD VARIABLES

Variable Name	Unit	Description
JD1	EST	Julian scenario start date and time as *YYDDDHH*
JD2	EST	Julian scenario end date and time as *YYDDDHH*
ICHK		Cumulus top calculation flag: $= 0 \Rightarrow$ P05G determines cloud top height $\neq 0 \Rightarrow$ P05G does not determine cloud top height
INTFLAG		Use of WBAN-format station identifiers: Y - WBAN format used; N - WBAN format not used

Example:

8819612 8819911 0

Y

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	13	files	84	blocks
FORTRAN INCLUDE files:	11	files	33	blocks
Object files:	13	files	51	blocks
Executable file:	1	file	102	blocks
	38	files	270	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:05:33
Buffered I/O count:	283
Direct I/O count:	2802
Peak working-set size:	2236
Peak virtual size:	4844

C. Space Requirements: Log and Print Files

P05GX1.LOG:	90 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.41 shows the input file and output file space requirements.

TABLE 2.41. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF111	1	8904	72 hours
	PF119	4	55	
	SURMET1	Nonstd	40251	1 month
			49210	
Output	PF133	5	1899	72 hours
	PF157	5	1899	72 hours
			3798	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P05GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P05GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P05GV1A.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P05GV1A.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P05G.OBJ          A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]ABTERM5.OBJ       A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]BOYANT.OBJ        A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]LCLVAL.OBJ        A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]QFIELD.OBJ        A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]QFORM.OBJ         A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SFCMET.OBJ        A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]TDPGRD.OBJ        A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]TMPRAL.OBJ        A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]TOPSET.OBJ        A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]TSHIFT.OBJ        A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]TSMUTH.OBJ        A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]VPROFL.OBJ        A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ         A14
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]AGRID1.OBJ        A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]BGRID1.OBJ        A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELLO.OBJ         A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ        A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ        A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GETYN.OBJ         A20
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GMTLCL.OBJ        A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ         A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ          A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOERR.OBJ         A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ        A25
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE2.OBJ        A26
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JHOUR1.OBJ        A27
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JULIAN.OBJ        A28
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ         A29
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ        A30
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ         A31
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ        A32
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ        A33
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT1.OBJ         A34
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT4.OBJ         A35
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ         A36
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDSRF1.OBJ        A37
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDWNDO.OBJ        A38
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT5.OBJ         A39
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ         A40
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P05GV1A.EXE -
  /MAP = MET1:[S01S]P05GV1A.MAP -
  A1, -
  A2, -
   .
   .
   .
  A38, -
  A39, -
  A40
$ PRINT -
  /NOHEADER -
```

```
/NOFEED -
/NOTIFY -
MET1:[S01S]P05GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P05GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P05GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                              PFMFDIR
$ ASSIGN MET23:[FOREMAN.RAWDATA]SRFC_JUL85.DAT                 SURMET1
$ ASSIGN MET56:[ROMNET1.D85188]PF111A.DAT                      PF111
$ ASSIGN MET1:[ROMNET1.D00000]PF119A.DAT                       PF119
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF133A.DAT                      PF133
$ ASSIGN MET56:[ROMNET1.D85188]PF157A.DAT                      PF157
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! CONTROL CARD FILE FOLLOWS:
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P05GX1.CPR
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P05GX1.CPR    SYS$INPUT
$ !
$ RUN $2$DUA24:[ROMEXE.ROMNET]P05GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A                    P05GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P05GV1A

B. Subroutines

ABTERM5	QFORM	TOPSET
BOYANT	SFCMET	TSHIFT
LCLVAL	TMPGRD	TSMUTH
QFIELD	TMPRAL	VPROFL

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	FSTAT1	OPDIR2	RDFT4
AGRID1	GMTLCL	OPEN2	RDSRF1
BGRID1	IBLKR	PROGID	RDWNDO
CELL0	IOERR	RDDIR1	WRFT5
FLSMRY	JFILE1	RDFT1	WRREC

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETYN	JHOUR1
IOCL	JULIAN
JFILE2	JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

FILE11.EXT	LFCNLV.EXT	QMULUS.EXT
FILE19.EXT	OUTPT5.EXT	SFCMET.EXT
GRID1.EXT	PROFIL.EXT	UAGRID.EXT
GRIDG.EXT	QHIST.EXT	

2.6 PROCESSOR P06G (TERRAIN ELEVATION)

I. PROCESSOR FUNCTION

The following discussion requires us to define three different grids:

- GRDNM1 is the actual modeling domain, and is the reference grid for the other two grids. GRDNM1 is defined by the latitude and longitude coordinates of the northeast corner and southwest corner (45° N, 69° W and $36^1/_3$° N, 85° W, respectively) of the ROMNET1 domain. GRDNM1 has a cell size of 10′-latitude × 15′-longitude, and dimensions of 64 columns × 52 rows; it thus encompasses $8^2/_3$ × 16 degrees of arc.
- GRDNM5 is identical to GRDNM1, except that it extends outwards on all sides by a 3-cell band, i.e., it has dimensions of 70 columns × 58 rows. This is the grid used to obtain the 30′-latitude × 45′-longitude averaged terrain elevations.
- GRDNM2 has a cell size of 5′-latitude × 5′-longitude, and is centered over GRDNM1. GRDNM2 extends 7-GRDNM1-sized cells outward from GRDNM1, i.e., by 70′ on its northern and southern boundaries, and by 105′ on its eastern and western boundaries; thus, its dimensions are 234 columns × 132 rows.

This stage 0 processor computes (1) terrain elevations averaged over the 5′-latitude × 5′-longitude cells of GRDNM2; (2) the following moving averages over 30′-latitude × 45′-longitude rectangles within GRDNM2: (a) terrain for every cell of GRDNM5, (b) north-south and east-west slopes for every cell of GRDNM5, and (c) terrain for every cell of GRDNM1. Note that this processor needs to be run only once per modeling domain.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

TERRAIN*n* (n = 1 to a maximum of 45)

These files each contain altitude matrices of elevations for 30″×30″ cells in a $7^1/_2{}'$ × $7^1/_2{}'$ grid. The TERRAIN*n* files are produced by the Environmental Protection Agency's GRIDS database, and are derived from Digital Elevation Model data.[5] The TERRAIN*n* files are read using the READ and FORMAT statements listed below, and their parameters are shown in Table 2.42.

```
      READ (IUN,1001,IOSTAT=IOST) YDEG,YMIN,XDEG,XMIN
     &     ((ELFT(I,J),I=1,15),J=1,15)
 1001 FORMAT 2(I4,F4.1),225I5
```

TABLE 2.42. TERRAIN*n* PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	YDEG	° N	Integer*4	Latitude, truncated to degrees
2	YMIN	′N	Real*4	Latitude fraction
3	XDEG	° W	Integer*4	Longitude, truncated to degrees
4	XMIN	′W	Real*4	Longitude fraction
5	$ELFT_{ij}$	ft AMSL	Integer*4	Cell elevation

Note: i = column number, j = row number.

5. For access to the GRIDS database, contact the EPA's Office of Information Resources Management (OIRM), Program Support Division, 1505 Planning Research Drive, McLean, VA 22102; attn. Bob Pease, Program Director, mail drop TM228. The toll-free telephone number is (800)877-5800, extension 8843.

B. Output Files

1. Standard Output Files

a. PF119

This file contains the GRDNM1 10′-latitude × 15′-longitude altitude matrix. Each of the 10′-latitude × 15′-longitude cell elevations consist of 54 averaged elevations of the 5′-latitude × 5′-longitude GRDNM2 subgrid cells. PF119 is used by P04G, P05G, P08G, P09G, P10G, P12G, P17G, and P33G; Table 2.43 shows its parameter.

TABLE 2.43. PF119 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$Zt_{i,j}$	m above MSL	Real*4	GRDNM1 terrain elevations

Note: i = column number, j = row number

b. PF140

This file contains an altitude matrix of averaged terrain elevations for the 5′-latitude × 5′-longitude GRDNM2 cells. PF140 is used by P07G, and Table 2.44 shows its parameter.

TABLE 2.44. PF140 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$\hat{}Zt_{i,j}$	m above MSL	Real*4	GRDNM2 terrain elevations

Note: i = column number, j = row number

c. PF164

This file contains the GRDNM5 30′-latitude × 45′-longitude altitude matrix. Each of the 30′-latitude × 45′-longitude cell elevations consist of 54 averaged elevations of the 5′-latitude × 5′-longitude GRDNM2 subgrid cells. PF164 is used by P07G, and Table 2.45 shows its parameter.

TABLE 2.45. PF164 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$Zt_{i,j}$	m above MSL	Real*4	GRDNM5 terrain elevations

Note: i = column number, j = row number

d. PF165

This file contains the gridded east-west terrain gradients $\partial Z_t / \partial x$, where Z_t is from PF164. It is used by P07G, and Table 2.46 shows its parameter.

TABLE 2.46. PF165 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$DELXZT_{i,j}$	$m \cdot m^{-1}$	Real*4	East-west terrain gradient

Note: i = column number, j = row number

e. PF166

This file contains the gridded north-south terrain gradients $\partial Z_t / \partial y$, where Z_t is from PF164. It is used by P07G, and Table 2.47 shows its parameter.

TABLE 2.47. PF166 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$DELYZT_{i,j}$	$m \cdot m^{-1}$	Real*4	North-south terrain gradient

Note: i = column number, j = row number

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used in the format shown below. Table 2.48 defines the control card variable.

N

TABLE 2.48. CONTROL CARD VARIABLE

Variable Name	Unit	Description
N		Number of input TERRAINn files used

Example:

1

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	35	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	15	blocks
Executable file:	1	file	546	blocks
	3	files	596	blocks

B. Execution Time Requirements

	VAX 8650
Charged CPU time (hh:mm:ss):	00:03:06
Buffered I/O count:	364
Direct I/O count:	9575
Peak working-set size:	1535
Peak virtual size:	3885

C. Space Requirements: Log and Print Files

P06GX1.LOG:	7073 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.49 shows the input file and output file space requirements.

TABLE 2.49. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	TERRAIN1	Nonstd	34260	N/A
Output	PF119	4	40	N/A
	PF140	4	217	N/A
	PF164	4	50	N/A
	PF165	4	50	N/A
	PF166	4	50	N/A
			407	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P06GV1B follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P06GV1B.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P06GV1B.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P06GV1B.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P06G.OBJ              A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ             A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ             A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ            A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ            A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GETNUM.OBJ            A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ             A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ              A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ            A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILEX.OBJ            A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ             A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]LEN1.OBJ              A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]LEN2.OBJ              A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ            A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ             A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ            A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ            A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ             A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT4.OBJ             A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ             A20
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P06GV1B.EXE -
  /MAP = MET1:[S01S]P06GV1B.MAP -
  A1, -
  A2, -
  A3, -
  .
  .
  .
  A18, -
  A19, -
  A20
```

```
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P06GV1B.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P06GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1050A]P06GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]TOPOG1.DAT         TERRAIN1
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET1:[ROMNET.D00000]PF119A.DAT          PF119
$ ASSIGN MET1:[ROMNET.D00000]PF140A.DAT          PF140
$ ASSIGN MET1:[ROMNET.D00000]PF164A.DAT          PF164
$ ASSIGN MET1:[ROMNET.D00000]PF165A.DAT          PF165
$ ASSIGN MET1:[ROMNET.D00000]PF166A.DAT          PF166
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN USER2$DISK:[FOREMAN.FV1001A]P06GX1.CPR  SYS$INPUT
$ !
$ SHOW PROCESS / ACCOUNTING
$ !
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P06GV1B.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P06GV1B

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	GETNUM	OPDIR2	RDREC
CELL0	IBLKR	OPEN2	WRFT4
FLSMRY	JFILE1	PROGID	WRREC
FSTAT1	JFILEX	RDDIR1	

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JUNIT
LEN1
LEN2

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

2.7 PROCESSOR P07G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 3 processor computes the thickness of layers 0 and 1, and the fraction of terrain penetration through the tops of these layers. During inversion conditions, it also calculates (1) the growth of the "cold layer" (the inversion layer), and (2) the wind field below the top of the inversion (equivalent to the top of layer 1).

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

P07G uses 9 standard input files:

PF101 from P01G
PF106 from P03G
PF107 from P04G
PF110 from P04G
PF112 from P03G
PF140 from P06G
PF164 from P06G
PF165 from P06G
PF166 from P06G

2. Nonstandard Input Files

None

B. Output Files

1. Standard Output Files

a. PF125

This file contains the hourly gridded growth rate of the cold-layer depth. It is used by P12G, and its parameter is shown in Table 2.50.

TABLE 2.50. PF125 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	ETA_{vs}	$m{\cdot}s^{-1}$	Real*4	Hourly gridded growth rate of the cold-layer depth

Note: vs = virtual surface, i.e., the top of the inversion layer.

b. PF127

This file contains the hourly gridded sub-inversion wind components. PF127 is used by P11G, and its parameters are shown in Table 2.51.

TABLE 2.51. MF127 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Sub-inversion, hourly gridded:
1	<u>vl	$m{\cdot}s^{-1}$	Real*4	West-to-east wind component
2	<v>vl	$m{\cdot}s^{-1}$	Real*4	North-to-south wind component

c. PF132

This file contains the hourly gridded cell thicknesses of layer 0. PF132 is used by P08G, P09G, P10G, P12G, P17G, P32G, and P33G. Its parameter is shown in Table 2.52.

TABLE 2.52. PF132 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	h0	m	Real*4	Hourly gridded cell thicknesses of layer 0

d. PF148

This file contains the hourly gridded elevation of the top of layer 1. PF148 is used by P11G, and its parameter is shown in Table 2.53.

TABLE 2.53. PF148 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	Z1	m above MSL	Real*4	Hourly gridded elevation of the top of layer 1

e. MF109

This file contains the hourly gridded cell thicknesses of layer 1. MF109 is used by P08G, P09G, P10G, P11G, P12G, P17G, P29G, P32G, and P33G. Its parameter is shown in Table 2.54.

TABLE 2.54. MF109 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	h1	m	Real*4	Hourly gridded cell thicknesses of layer 1

f. MF110

This file contains the hourly gridded fraction of the top of layer 0 that is penetrated by terrain. MF110 is used by P10G, P12G, P33G, and P39G. Its parameter is shown in Table 2.55.

TABLE 2.55. MF110 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SIGMAt0	m above MSL	Real*4	Hourly gridded fraction of top of layer 0 penetrated by terrain

g. MF111

This file contains the hourly gridded fraction of the top of layer 1 that is penetrated by terrain. MF111 is used by P10G, P12G, P33G, and P39G. Its parameter is shown in Table 2.56.

TABLE 2.56. MF111 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SIGMAt1	m above MSL	Real*4	Hourly gridded fraction of top of layer 1 penetrated by terrain

2. Nonstandard Output Files

None

III. CONTROL CARDS

Three control cards are used to input control data in the format shown below. Table 2.57 defines the control card variables.

YYDDD HH TT
DT
H00

TABLE 2.57. CONTROL CARD VARIABLES

Variable Name	Unit	Description
YYDDD		Julian start date as *YYDDD*
HH	EST	Start hour as *HH*
TT	h	Scenario duration
DT	s	Time step for subroutine FLOMOD
H00	m	Initial layer 0 thickness

Example:

88194 12 24
120
30.0

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	25	files	390	blocks
FORTRAN INCLUDE files:	18	files	72	blocks
Object files:	25	files	144	blocks
Executable file:	1	file	159	blocks
	69	files	765	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

The charged CPU time can be highly variable, and depends on the number of simulated inversion hours. A rule-of-thumb is that almost 6 minutes of CPU time are required per modeled inversion hour. The run time with no inversion hours is ≃52 s per modeled hour.

	VAX 8650
Charged CPU time (hh:mm:ss):	01:38:13
Buffered I/O count:	347
Direct I/O count:	4442
Peak working-set size:	3202
Peak virtual size:	5655

C. Space Requirements: Log and Print Files

P07GX1.LOG:	234 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.58 shows the input file and output file space requirements.

TABLE 2.58. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF101	1	6465	72 hours
	PF106	3	2142	72 hours
	PF107	5	1899	72 hours
	PF110	5	1899	72 hours
	PF112	5	1899	72 hours
	PF140	4	286	
	PF164	4	66	
	PF165	4	66	
	PF166	4	66	
			14788	
Output	PF125	5	1794	72 hours
	PF127	5	3615	72 hours
	PF132	5	1899	72 hours
	PF148	5	1899	72 hours
	MF109	5	1899	72 hours
	MF110	5	1899	72 hours
	MF111	5	1899	72 hours
			14904	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P07GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P07GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P07GV1A.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P07GV1A.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P07G.OBJ            A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]BARS.OBJ            A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]BASEVA.OBJ          A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CLOCK.OBJ           A4
```

```
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]COLDGR.OBJ          A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CORFOR.OBJ          A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CPULOG.OBJ          A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]DPCD.OBJ            A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]EXTEND.OBJ          A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]FLOMOD.OBJ          A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]FRAME.OBJ           A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]GRADX6.OBJ          A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]H1HOZ.OBJ           A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]MINMAX.OBJ          A14
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]MINMX2.OBJ          A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]PRIMES.OBJ          A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]RADRAG.OBJ          A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SETBCS.OBJ          A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SETUP7.OBJ          A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SIGMAT.OBJ          A20
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SLPRES.OBJ          A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SMOOTH.OBJ          A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SOLVDH.OBJ          A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SOLVER.OBJ          A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SOLVUV.OBJ          A25
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ           A26
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELLO.OBJ           A27
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CLRHDR.OBJ          A28
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CPUTOT.OBJ          A29
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]DATHR3.OBJ          A30
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ          A31
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ          A32
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GETMSG.OBJ          A33
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ           A34
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ            A35
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ          A36
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JHOUR1.OBJ          A37
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ           A38
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]NEXTHR.OBJ          A39
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ          A40
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ           A41
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ          A42
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ          A43
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFHD1.OBJ          A44
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFHD2.OBJ          A45
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT1.OBJ           A46
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT3.OBJ           A47
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT4.OBJ           A48
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT5.OBJ           A49
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ           A50
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT5.OBJ           A51
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ           A52
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P07GV1A.EXE -
  /MAP = MET1:[S01S]P07GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A50, -
  A51, -
  A52
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P07GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P07GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P07GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                 PFMFDIR
$ ASSIGN MET56:[ROMNET1.D85188]PF101A.DAT         PF101
$ ASSIGN MET56:[ROMNET1.D85188]PF106A.DAT         PF106
$ ASSIGN MET56:[ROMNET1.D85188]PF107A.DAT         PF107
$ ASSIGN MET56:[ROMNET1.D85188]PF110A.DAT         PF110
$ ASSIGN MET56:[ROMNET1.D85188]PF112A.DAT         PF112
$ ASSIGN MET1:[ROMNET1.D00000]PF140A.DAT          PF140
$ ASSIGN MET1:[ROMNET1.D00000]PF164A.DAT          PF164
$ ASSIGN MET1:[ROMNET1.D00000]PF165A.DAT          PF165
$ ASSIGN MET1:[ROMNET1.D00000]PF166A.DAT          PF166
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF125A.DAT         PF125
$ ASSIGN MET56:[ROMNET1.D85188]PF127A.DAT         PF127
$ ASSIGN MET56:[ROMNET1.D85188]PF132A.DAT         PF132
$ ASSIGN MET56:[ROMNET1.D85188]PF148A.DAT         PF148
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT         MF109
$ ASSIGN MET56:[ROMNET1.D85188]MF110A.DAT         MF110
$ ASSIGN MET56:[ROMNET1.D85188]MF111A.DAT         MF111
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARDS FILE
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P07GX1.CPR
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P07GX1.CPR   SYS$INPUT
$ !
$ RUN $2$DUA24:[ROMEXE.ROMNET]P07GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A      P07GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P07GV1A

B. Subroutines

BARS	DPCD	MINMAX	SIGMAT
BASEVA	EXTEND	MINMX2	SLPRES
CLOCK	FLOMOD	PRIMES	SMOOTH
COLDGR	FRAME	RADRAG	SOLVDH
CORFOR	GRADX6	SETBCS	SOLVUV
CPULOG	H1H0Z	SETUP7	

C. Functions

SOLVER

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	NEXTHR	RDFT1
CELL0	OPDIR2	RDFT3
CLRHDR	OPEN2	RDFT4
CPUTOT	PROGID	RDFT5
FLSMRY	RDDIR1	RDREC
FSTAT1	RDFHD1	WRFT5
IBLKR	RDFHD2	WRREC

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

DATHR3	JFILE1
GETMSG	JHOUR1
IOCL	JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

BASEV.EXT	OBLEN.EXT	PF1.EXT	SOLGRP.EXT
BCON.EXT	PARM7.EXT	PRES.EXT	SURDAT.EXT
CD.EXT	PARMS.EXT	PROCES.EXT	ZT.EXT
CORF.EXT	PERTV.EXT	QFLUX.EXT	ZTHAT.EXT
ETAVS.EXT			

2.8 PROCESSOR P08G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 4 processor determines (1) the thickness of layer 2, (2) the updraft speed at the base of any cumulus clouds that it models to exist,[6] (3) the thickness of model layer 3, whose top is defined as the height at the top of the cumulus clouds.[7] P08G also produces an interim estimate of the fraction of the air that originates in layer 0 that is then updrafted into the cumulus cloud.

6. Cumulus cloud bases are defined to be located at the base of layer 3.
7. If cumulus clouds are not modeled to exist, the height of the upper boundary is defined as 100 m above layer 2.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

PF111 from P16G

PF119 from P06G

PF132 and MF109 from P07G

2. Nonstandard Input Files

a. SURMET1

SURMET1 contains hourly surface meteorology observations processed from the National Climatic Data Center and the Canadian Climate Centre raw data tapes. SURMET1 is read in two stages. The first stage reads header information, and the first READ and FORMAT statements for this file are as follows:

```
        READ (LU,93001,IOSTAT=IOST) INYR,INMON,INDAY,INHOUR,
     &     NSTAT
93001   FORMAT (4(I2),5X,I3)
```

Table 2.59 shows the SURMET 1 Stage 1 parameters.

TABLE 2.59. SURMET1 STAGE 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	INYR		Integer*4	Scenario year (*YY*)
2	INMON		Integer*4	Scenario month (*MM*)
3	INDAY		Integer*4	Scenario day (*DD*)
4	INHOUR	EST	Integer*4	Scenario hour (*HH*)
5	NSTAT		Integer*4	No. of reporting stations at *HH*

The Stage 2 READ and FORMAT statements are listed below, and the Stage 2 parameters are shown in Table 2.60:

```
        READ (LU,93003,IOSTAT=IOST) PARMS
93003   FORMAT (A4,2F7.2,F5.0,F4.0,3(F5.0,F6.0)
     &        F8.1,4(F7.1),F8.1)
```

The READ and FORMAT statements above are for non-WBAN format for the surface meteorology stations. For data in WBAN format, these statements read as follows:

```
        READ (LU,93004,IOSTAT=IOST) WBAN, (PTEMP(N), N=2,NPARM)
93004   FORMAT (I5.5,F6.2,F7.2,F5.0,F4.0,3(F5.0,F6.0)
     &         F8.1,4(F7.1),F8.1)
        DO 2110 N=1, NPARM
           PARMS(N) = PTEMP(N)
2110    CONTINUE
```

Note that the parameters shown in Table 3.61 are contained within a dimensioned array of 17 elements, PARMS, so that information can be easily passed from subroutine to subroutine. The parameters are not listed explicitly in the two READ statements above.

TABLE 2.60. SURMET1 STAGE 2 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1a	$STAID_n$		Char*4	Station ID (non-WBAN format)
1b	$STAID_n$		Integer*4	Station ID (WBAN format)
2	$SLAT_n$	° N	Real*4	Latitude
3	$SLON_n$	° W	Real*4	Longitude
4	$OPQCOV_n$	%	Real*4	Opaque sky cover
5	$TOTCOV_n$		Real*4	Total sky cover: 0 = clear 1 = partial obscuration 2 = thin scattered 3 = thin broken 4 = thin overcast 5 = scattered 6 = broken 7 = overcast 8 = obscured -99 = indeterminate
6	$COV1_n$	%	Real*4	Coverage, lowest cloud layer
7	$BASE1_n$	m	Real*4	Lowest cloud base
8	$COV2_n$	%	Real*4	Coverage, second-lowest cloud layer
9	$BASE2_n$	m	Real*4	Second-lowest cloud base
10	$COV3_n$	%	Real*4	Coverage, third-lowest cloud layer
11	$BASE3_n$	m	Real*4	Third-lowest cloud base
12	$P0_n$	mb	Real*4	Sea level pressure
13	DIR_n	deg	Real*4	Wind direction
14	SPD_n	$m{\cdot}s^{-1}$	Real*4	Wind speed
15	T_n	°C	Real*4	Temperature
16	TD_n	°C	Real*4	Dew point
17	P_n	mb	Real*4	Station pressure

Note: n = surface station number (one or more digits).

b. RAWSTA1

RAWSTA1 contains the land use classification of each rawind (upper-air) station in the domain; these data are used in the updraft calculation from layer 0. The READ statements for this file are as follows:

```
READ (STAT, *, IOSTAT = IOST) GRIDNM, YEARNM
READ (STAT, *, IOSTAT = IOST) IDRAD(N), TMP1, TMP2, TMP3,
                              (LNDUSE(JJ,n), JJ=1,10)
```

Table 2.61 shows the RAWSTA1 parameters.

TABLE 2.61. RAWSTA1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	GRIDNM		Char*8	Region name (see Appendix A)
2	YEARNM		Integer*4	Scenario year (*YY*)
3	$IDRAD_m$		Integer*4	Station WBAN identifier
4	$TMP1_m$	° N	Real*4	Station latitude
5	$TMP2_m$	° W	Real*4	Station longitude
6	$TMP3_m$	m	Real*4	Station elevation
7	$LNDUSE_m$		Real*4	Land use classification:
	URBAN			urban
	AGRI			agriculture
	RANGE			rangeland
	DF			deciduous forest
	CF			coniferous forest
	MF			mixed forests including wetlands
	WATER			water
	BARREN			barren land
	NFW			nonforested wetland
	MAR			mixed agriculture/rangeland

Note: m = upper-air station number (one or more digits).

B. Output Files

1. Standard Output Files

P08G produces 4 standard output files: PF142, PF156, MF112, and MF113.

PF142 is used by processor P12G.
PF156 is used by processors P12G and P32G.
MF112 is used by processors P09G, P10G, P11G, P12G, P17G, P29G, P32G, and P33G.
MF113 is used by processors P09G, P11G, P12G, P17G, P29G, and P32G.

a. PF142

This file contains hourly data on the standard 64-cell by 52-cell grid; Table 2.62 shows its parameter.

TABLE 2.62. PF142 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	PSI′		Real*4	Fraction of air entering a cumulus cloud that originates in Layer 0

b. PF156

This file contains hourly data on the standard 64-cell by 52-cell grid; Table 2.63 shows its parameter.

TABLE 2.63. PF156 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	barWc	$m{\cdot}s^{-1}$	Real*4	Cumulus cloud average updraft speed

c. MF112

This file contains hourly data on the standard 64-cell by 52-cell grid; Table 2.64 shows its parameter.

TABLE 2.64. MF112 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	h2	m	Real*4	Layer 2 cell thickness

d. MF113

This file contains hourly data on the standard 64-cell by 52-cell grid; Table 2.65 shows its parameter.

TABLE 2.65. MF113 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	h3	m	Real*4	Layer 3 cell thickness

2. Nonstandard Output Files

None

III. CONTROL CARDS

Two control cards are used to input control data in the format shown below. Table 2.66 defines the control card variables.

JD1 JD2
INTFLG

TABLE 2.66. CONTROL CARD VARIABLES

Variable Name	Unit	Description
JD1	EST	Julian scenario start date and time as *YYDDDHH*
JD2	EST	Julian scenario end date and time as *YYDDDHH*
INTFLG		Flag indicating whether surface stations are in WBAN format

Example:

8819612 8819911
Y

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	19	files	264	blocks
FORTRAN INCLUDE files:	1	files	4	blocks
Object files:	19	files	120	blocks
Executable file:	1	file	126	blocks
	40	files	514	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:07:28
Buffered I/O count:	287
Direct I/O count:	5762
Peak working-set size:	1738
Peak virtual size:	4411

C. Space Requirements: Log and Print Files

P08GX1.LOG:	411 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.67 shows the input file and output file space requirements.

TABLE 2.67. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF111	1	8904	72 hours
	PF119	4	55	72 hours
	PF132	5	1899	72 hours
	MF109	5	1899	72 hours
	SURMET1	Nonstd	40251	1 month
	RAWSTA1	Nonstd	24	N/A
			53032	
Output	PF142	5	1899	72 hours
	PF156	5	1899	72 hours
	MF112	5	1899	72 hours
	MF113	5	1899	72 hours
			7596	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P08GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P08GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P08GV1A.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P08GV1A.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P08G.OBJ        A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]ABTERM.OBJ      A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]ALBALF.OBJ      A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]ARCEST.OBJ      A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CLDFN8.OBJ      A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CLDMOD.OBJ      A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CLDPAR.OBJ      A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CLDRAD.OBJ      A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CRTEL8.OBJ      A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]DELTAT.OBJ      A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]FLXRAD.OBJ      A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]GMOVS8.OBJ      A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]PBLMIX.OBJ      A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]PROFIL.OBJ      A14
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]PTOZ.OBJ        A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]QFLUX8.OBJ      A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]RADLW8.OBJ      A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]RADPFL.OBJ      A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]RTIMOD.OBJ      A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ       A20
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELLO.OBJ       A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ      A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ      A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GETYN.OBJ       A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GMTLCL.OBJ      A25
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GREG.OBJ        A26
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ       A27
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ        A28
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOERR.OBJ       A29
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ      A30
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE2.OBJ      A31
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JHOUR1.OBJ      A32
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JULIAN.OBJ      A33
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ       A34
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ      A35
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ       A36
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ      A37
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ      A38
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT1.OBJ       A39
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT4.OBJ       A40
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT5.OBJ       A41
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ       A42
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDSRF1.OBJ      A43
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT5.OBJ       A44
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ       A45
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P08GV1A.EXE -
  /MAP = MET1:[S01S]P08GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A43, -
  A44, -
  A45
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P08GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P08GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P08GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]RAWSTA.DAT         RAWSTA1
$ ASSIGN MET23:[FOREMAN.RAWDATA]SRFC_JUL85.DAT   SURMET1
$ ASSIGN MET56:[ROMNET1.D85188]PF111A.DAT        PF111
$ ASSIGN MET1:[ROMNET1.D00000]PF119A.DAT         PF119
$ ASSIGN MET56:[ROMNET1.D85188]PF132A.DAT        PF132
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT        MF109
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF142A.DAT        PF142
$ ASSIGN MET56:[ROMNET1.D85188]PF156A.DAT        PF156
$ ASSIGN MET56:[ROMNET1.D85188]MF112A.DAT        MF112
$ ASSIGN MET56:[ROMNET1.D85188]MF113A.DAT        MF113
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! CONTROL CARDS FILE FOLLOWS:
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P08GX1.CPR
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P08GX1.CPR   SYS$INPUT
$ !
$ RUN $2$DUA24:[ROMEXE.ROMNET]P08GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A      P08GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P08GV1A

B. Subroutines

ABTERM	CLDPAR	FLXRAD	QFLUX8
ALBALF	CLDRAD	PBLMIX	RADPFL
ARCEST	CRTEL8	PROFIL	RTIMOD
CLDMOD	DELTAT	PTOZ	

C. Functions

RADLW8
CLDFN8
GMOVS8

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	GMTLCL	JFILE1	RDDIR1	RDSRF1
CELL0	GREG	OPDIR2	RDFT1	WRFT5
FLSMRY	IBLKR	OPEN2	RDFT4	WRREC
FSTAT1	IOERR	PROGID	RDFT5	

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETYN	JFILE2	JULIAN
IOCL	JHOUR1	JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

2.9 PROCESSOR P09G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 5 processor computes atmospheric density, temperature, water vapor concentration, cloud cover, and solar zenith angle. P09G computes these parameters in two subroutines: (1) subroutine DEN computes the hourly gridded layer-averaged air density, the air density at the top of each layer, the temperature, and the water vapor concentration;[8] (2) subroutine ZEN generates the hourly gridded zenith angle and the cloud-layer transmission factor. The computed parameters are used by the Core Model to calculate the adjustment factors for the chemical rate constants that are determined by preprocessor CVCHEM (see Section 7.2, Part 1), as well as being used in other processors.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

P09G uses 8 standard input files:

PF102 from P01G	PF131 from P01G	MF112 from P08G
PF104 from P01G	PF132 from P07G	MF113 from P08G
PF119 from P06G	MF109 from P07G	

8. We refer to water vapor concentration instead of to specific humidity; although the units for either parameter are identical, water vapor concentration is calculated on a volume basis ($10^{-6}L \cdot L^{-1}$), rather than on the mass basis ($10^{-6}kg \cdot kg^{-1}$) of specific humidity.

2. Nonstandard Input Files

P09G uses one nonstandard input file, SURMET1, that contains hourly surface meteorology observations from the National Climatic Data Center and the Canadian Climate Centre raw data tapes. SURMET1 is read in two stages. The first stage reads header information. READ and FORMAT statements for SURMET1 are called by P09G from RDSRF1 in the UTILIO library. Table 2.68 shows the SURMET1 Stage 1 parameters.

TABLE 2.68. SURMET1 STAGE 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	INYR		Integer*4	Scenario year (*YY*)
2	INMON		Integer*4	Scenario month (*MM*)
3	INDAY		Integer*4	Scenario day (*DD*)
4	INHOUR	EST	Integer*4	Scenario hour (*HH*)
5	NSTAT		Integer*4	No. of reporting stations at *HH*

The Stage 2 parameters that are used in P09G are described in Table 2.69:

TABLE 2.69. SURMET1 STAGE 2 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1a	$STAID_n$		Char*4	Station ID (non-WBAN format)
1b	$STAID_n$		Integer*4	Station ID (WBAN format)
2	$SLATS_n$	° N	Real*4	Latitude
3	$SLONS_n$	° W	Real*4	Longitude
4	$OPQCOV_n$	%	Real*4	Opaque sky cover
5	$TOTCOV_n$		Real*4	Total sky cover (see note below)
6	$COVER1_n$	%	Real*4	Coverage, lowest cloud layer
7	$BASE1_n$	m	Real*4	Lowest cloud base
8	$COVER2_n$	%	Real*4	Coverage, second-lowest cloud layer
9	$BASE2_n$	m	Real*4	Second-lowest cloud base
10	$COVER3_n$	%	Real*4	Coverage, third-lowest cloud layer
11	$BASE3_n$	m	Real*4	Third-lowest cloud base
12	$P0_n$	mb	Real*4	Sea level pressure
13	DIR_n	deg	Real*4	Wind direction
14	SPD_n	$m \cdot s^{-1}$	Real*4	Wind speed
15	T_n	°C	Real*4	Temperature
16	TD_n	°C	Real*4	Dew point
17	P_n	mb	Real*4	Station pressure

Note: n = surface station number (one or more digits). Total sky cover classes are as follows: 0 = clear; 1 = partial obscuration; 2 = thin scattered; 3 = thin broken; 4 = thin overcast; 5 = scattered; 6 = broken; 7 = overcast; 8 = obscured; -99 = indeterminate.

B. Output Files

1. Standard Output Files

a. MF132

This file contains the hourly, gridded, layer-averaged air density in layer 0. It is used by P10G, P33G, and P39G. Table 2.70 shows its parameter.

TABLE 2.70. MF132 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	<RHO>$0_{i,j,h}$	$mol{\cdot}ppm^{-1}{\cdot}m^{-3}$	Real*4	Layer-averaged air density in layer 0

Note: i = column number, j = row number, h = hour; units are mass per volume.

b. MF133

This file contains the hourly, gridded, layer-averaged air density in layer 1. It is used by P10G, P12G, P29G, P33G, and P39G. Table 2.71 shows its parameter.

TABLE 2.71. MF133 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	<RHO>$1_{i,j,h}$	$mol{\cdot}ppm^{-1}{\cdot}m^{-3}$	Real*4	Layer-averaged air density in layer 1

Note: i = column number, j = row number, h = hour; units are mass per volume.

c. MF134

This file contains the hourly, gridded, layer-averaged air density in layer 2. It is used by P10G, P12G, P29G, P33G, and P39G. Table 2.72 shows its parameter.

TABLE 2.72. MF134 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	<RHO>$2_{i,j,h}$	$mol{\cdot}ppm^{-1}{\cdot}m^{-3}$	Real*4	Layer-averaged air density in layer 2

Note: i = column number, j = row number, h = hour; units are mass per volume.

d. MF135

This file contains the hourly, gridded, layer-averaged air density in layer 3. It is used by P12G, P29G, and P39G. Table 2.73 shows its parameter.

TABLE 2.73. MF135 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$\langle \mathrm{RHO} \rangle 3_{i,j,h}$	$\mathrm{mol \cdot ppm^{-1} \cdot m^{-3}}$	Real*4	Layer-averaged air density in layer 3

Note: i = column number, j = row number, h = hour; units are mass per volume.

e. MF136

This file contains the hourly, gridded, layer-averaged air temperature in layer 1. It is used by P39G, and Table 2.74 shows its parameter.

TABLE 2.74. MF136 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$\langle T \rangle 1_{i,j,h}$	K	Real*4	Layer-averaged air temperature for layer 1

Note: i = column number, j = row number, h = hour

f. MF137

This file contains the hourly, gridded, layer-averaged air temperature in layer 2. It is used by P39G, and Table 2.75 shows its parameter.

TABLE 2.75. MF137 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$\langle T \rangle 2_{i,j,h}$	K	Real*4	Layer-averaged air temperature for layer 2

Note: i = column number, j = row number, h = hour

g. MF138

This file contains the hourly, gridded, layer-averaged air temperature in layer 3. It is used by P39G, and Table 2.76 shows its parameter.

TABLE 2.76. MF138 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$<T>3_{i,j,h}$	K	Real*4	Layer-averaged air temperature for layer 3

Note: i = column number, j = row number, h = hour

h. MF139

This file contains the hourly gridded solar zenith angle; grid cell latitude and longitude, and meridional variations in local standard time, are considered by REGPRM.EXT. It is used by P15G, P27G, and P39G. Table 2.77 shows its parameter.

TABLE 2.77. MF139 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$theta_{i,j,h}$	°	Real*4	Solar zenith angle

Note: i = column number, j = row number, h = hour

i. MF140

This file contains the hourly gridded cloud-layer transmission factor for solar energy. It is used by P15G and P39G. Table 2.78 shows its parameter.

TABLE 2.78. MF140 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$psi_{i,j,h}$		Real*4	Cloud-layer transmission factor: 1 = clear sky 0 = opaque sky

Note: i = column number, j = row number, h = hour

j. MF160

This file contains the hourly gridded air density at the top of layer 0. It is used by P12G and P39G. Table 2.79 shows its parameter.

TABLE 2.79. MF160 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$SRHO0_{i,j,h}$	$mol \cdot ppm^{-1} \cdot m^{-3}$	Real*4	Air density at the top surface of layer 0

Note: i = column number, j = row number, h = hour; units are mass per volume.

k. MF161

This file contains the hourly gridded air density at the top of layer 1. It is used by P12G and P39G. Table 2.80 shows its parameter.

TABLE 2.80. MF161 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$SRHO1_{i,j,h}$	$mol \cdot ppm^{-1} \cdot m^{-3}$	Real*4	Air density at the top surface of layer 1

Note: i = column number, j = row number, h = hour; units are mass per volume.

l. MF162

This file contains the hourly gridded air density at the top of layer 2. It is used by P12G and P39G. Table 2.81 shows its parameter.

TABLE 2.81. MF162 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$SRHO2_{i,j,h}$	$mol \cdot ppm^{-1} \cdot m^{-3}$	Real*4	Air density at the top surface of layer 2

Note: i = column number, j = row number, h = hour; units are mass per volume.

m. MF163

This file contains the hourly gridded air density at the top of layer 3. It is used by P12G and P39G. Table 2.82 shows its parameter.

TABLE 2.82. MF163 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$SRHO3_{i,j,h}$	$mol \cdot ppm^{-1} \cdot m^{-3}$	Real*4	Air density at the top surface of layer 3

Note: i = column number, j = row number, h = hour; units are mass per volume.

n. MF174

This file contains the hourly, gridded, layer-averaged water vapor concentration in layer 1. It is used by P39G, and Table 2.83 shows its parameter.

TABLE 2.83. MF174 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$<wvc>1_{i,j,h}$	ppm	Real*4	Layer-averaged water vapor conc. in layer 1

Note: i = column number, j = row number, h = hour

o. MF175

This file contains the hourly, gridded, layer-averaged water vapor concentration in layer 2. It is used by P39G, and Table 2.84 shows its parameter.

TABLE 2.84. MF175 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$<wvc>2_{i,j,h}$	ppm	Real*4	Layer-averaged water vapor conc. in layer 2

Note: i = column number, j = row number, h = hour

p. MF176

This file contains the hourly, gridded, layer-averaged water vapor concentration in layer 3. It is used by P39G, and Table 2.85 shows its parameter.

TABLE 2.85. MF176 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$<wvc>3_{i,j,h}$	ppm	Real*4	Layer-averaged water vapor conc. in layer 3

Note: i = column number, j = row number, h = hour

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 2.86 defines the control card variables.

ISDAT IENDAT INTFLAG

TABLE 2.86. CONTROL CARD VARIABLES

Variable Name	Unit	Description
ISDAT	EST	Julian scenario start date and time as *YYDDDHH*
IENDAT	EST	Julian scenario end date and time as *YYDDDHH*
INTFLAG		Use of WBAN-format station identifiers in SURMET1: .TRUE. ⇒ WBAN format used; .FALSE. ⇒ WBAN format not used

Example:

8819612 8819911 .TRUE.

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	3	files	65	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	3	files	22	blocks
Executable file:	1	file	98	blocks
	7	files	185	blocks

B. Execution Time Requitements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	01:19:20
Buffered I/O count:	434
Direct I/O count:	12823
Peak working-set size:	8192
Peak virtual size:	13068

C. Space Requirements: Log and Print Files

P09GX1.LOG:	47 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.87 shows the input file and output file space requirements.

TABLE 2.87. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF102	1	7128	72 hours
	PF104	1	294	72 hours
	PF119	4	4	72 hours
	PF131	1	5724	72 hours
	PF132	5	1438	72 hours
	MF109	5	1438	72 hours
	MF112	5	1438	72 hours
	MF113	5	1438	72 hours
	SURMET1	Nonstd	88714	1 month
			107616	
Output	MF132	5	1438	72 hours
	MF133	5	1438	72 hours
	MF134	5	1438	72 hours
	MF135	5	1438	72 hours
	MF136	5	1438	72 hours
	MF137	5	1438	72 hours
	MF138	5	1438	72 hours
	MF139	5	1438	72 hours
	MF140	5	1438	72 hours
	MF160	5	1438	72 hours
	MF161	5	1438	72 hours
	MF162	5	1438	72 hours
	MF163	5	1438	72 hours
	MF174	5	1438	72 hours
	MF175	5	1438	72 hours
	MF176	5	1438	72 hours
			23008	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P09GV1B.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P09GV1B.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P09GV1B.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P09GV1B.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P09G.OBJ           A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]DEN.OBJ            A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]ZEN.OBJ            A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ          A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]BGRID1.OBJ         A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]BGRID3.OBJ         A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ          A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ         A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ         A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GMTLCL.OBJ         A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GREG.OBJ           A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ          A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ           A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOERR.OBJ          A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ         A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ         A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ         A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JULIAN.OBJ         A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ          A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ         A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ          A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ         A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ         A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT1.OBJ          A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT4.OBJ          A25
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ          A26
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ          A27
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDSRF1.OBJ         A28
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDWNDO.OBJ         A29
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]SOLAR.OBJ          A30
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT5.OBJ          A31
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ          A32
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P09GV1B.EXE -
  /MAP = MET1:[S01S]P09GV1B.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A30, -
  A31, -
  A32
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P09GV1B.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P09GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P09GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                  PFMFDIR
$ ASSIGN MET23:[FOREMAN.RAWDATA]SRFC_JUL85.DAT     SURMET1
$ ASSIGN MET56:[ROMNET1.D85188]PF102A.DAT          PF102
$ ASSIGN MET56:[ROMNET1.D85188]PF104A.DAT          PF104
$ ASSIGN MET1:[ROMNET1.D00000]PF119A.DAT           PF119
$ ASSIGN MET56:[ROMNET1.D85188]PF131A.DAT          PF131
$ ASSIGN MET56:[ROMNET1.D85188]PF132A.DAT          PF132
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT          MF109
$ ASSIGN MET56:[ROMNET1.D85188]MF112A.DAT          MF112
$ ASSIGN MET56:[ROMNET1.D85188]MF113A.DAT          MF113
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]MF132A.DAT          MF132
$ ASSIGN MET56:[ROMNET1.D85188]MF133A.DAT          MF133
$ ASSIGN MET56:[ROMNET1.D85188]MF134A.DAT          MF134
$ ASSIGN MET56:[ROMNET1.D85188]MF135A.DAT          MF135
$ ASSIGN MET56:[ROMNET1.D85188]MF136A.DAT          MF136
$ ASSIGN MET56:[ROMNET1.D85188]MF137A.DAT          MF137
$ ASSIGN MET56:[ROMNET1.D85188]MF138A.DAT          MF138
$ ASSIGN MET56:[ROMNET1.D85188]MF139A.DAT          MF139
$ ASSIGN MET56:[ROMNET1.D85188]MF140A.DAT          MF140
$ ASSIGN MET56:[ROMNET1.D85188]MF160A.DAT          MF160
$ ASSIGN MET56:[ROMNET1.D85188]MF161A.DAT          MF161
$ ASSIGN MET56:[ROMNET1.D85188]MF162A.DAT          MF162
$ ASSIGN MET56:[ROMNET1.D85188]MF163A.DAT          MF163
$ ASSIGN MET56:[ROMNET1.D85188]MF174A.DAT          MF174
$ ASSIGN MET56:[ROMNET1.D85188]MF175A.DAT          MF175
$ ASSIGN MET56:[ROMNET1.D85188]MF176A.DAT          MF176
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P09GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P09GX1.CPR   SYS$INPUT
$ !
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P09GV1B.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P09GV1B

B. Subroutines

DEN
ZEN

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	GREG	PROGID	RDSRF1
BGRID1	IBLKR	RDDIR1	RDWNDO
BGRID3	IOERR	RDFT1	SOLAR
CELL0	JFILE1	RDFT4	WRFT5
FLSMRY	OPDIR2	RDFT5	WRREC
GMTLCL	OPEN2	RDREC	

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JHOUR1
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

2.10 PROCESSOR P10G (EMISSIONS)

I. PROCESSOR FUNCTION

This stage 7 processor transforms the emissions output from P14G and P27G into the hourly gridded area-flux emission rates (the *source strength functions*) *S*, S1, and S2. These three functions produce the emissions rate for layers 0, 1, and 2, respectively. The three basic tasks involved in computing the source strengths are: (1) estimating the effective heights of the major point sources, (2) partitioning the point- and area-source emissions among layers 0, 1, and 2, and (3) converting the results to units of parts per million (ppm). This processor also computes the plume volume fraction that is used in the layer 0 equations to parameterize subgrid-scale chemistry processes.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

P10G uses 17 standard input files:

PF106 from P03G	PF141 from P14G	MF111 from P07G
PF107 from P04G	PF144 from P27G	MF112 from P08G
PF115 from P11G	PF145 from P14G	MF132 from P09G
PF119 from P03G	PF155 from P33G	MF133 from P09G
PF132 from P07G	MF109 from P07G	MF134 from P09G
PF135 from P13G	MF110 from P07G	

2. Nonstandard Input Files

P10G uses one nonstandard input file, PTE3, that contains hourly emission rates for major point sources. PTE3 is an output file from P14G. This unformatted file is read into P10G with the following READ statement, and its parameters are shown in Table 2.88.

```
      READ (IUNIT,IOSTAT=IEOF) STKID,PSLON,PSLAT,EXFL,PLMT,
     &     DK,ZSK,TK,WK,(EK(L),L=1,NP)
```

TABLE 2.88. PTE3 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	STKID		Char*12	NEDS state/county/plant/point ID
2	PSLON	° W	Real*4	Longitude of stack
3	PSLAT	° N	Real*4	Latitude of stack
4	EXFL	$m^3 \cdot s^{-1}$	Real*4	Exhaust flow rate
5	PLMT	m	Real*4	Plume height
6	Dk	m	Real*4	Stack diameter
7	Zsk	m	Real*4	Stack height
8	Tk	°C	Real*4	Exhaust gas temperature
9	Wk	$m \cdot s^{-1}$	Real*4	Exhaust gas velocity
				Emission rate for:
10	EK(1)	$mol \cdot h^{-1}$	Real*4	Higher aldehydes
11	EK(2)	$mol \cdot h^{-1}$	Real*4	Carbon monoxide
12	EK(3)	$mol \cdot h^{-1}$	Real*4	Ethylene
13	EK(4)	$mol \cdot h^{-1}$	Real*4	Formaldehyde
14	EK(5)	$mol \cdot h^{-1}$	Real*4	Isoprene
15	EK(6)	$mol \cdot h^{-1}$	Real*4	Nitric oxide
16	EK(7)	$mol \cdot h^{-1}$	Real*4	Nitrogen dioxide
17	EK(8)	$mol \cdot h^{-1}$	Real*4	Olefins
18	EK(9)	$mol \cdot h^{-1}$	Real*4	Paraffins
19	EK(10)	$mol \cdot h^{-1}$	Real*4	Toluene
20	EK(11)	$mol \cdot h^{-1}$	Real*4	Xylene
21	EK(12)	$mol \cdot h^{-1}$	Real*4	Nonreactive hydrocarbons
22	EK(13)	$mol \cdot h^{-1}$	Real*4	Methane
23	EK(14)	$mol \cdot h^{-1}$	Real*4	Methanol
24	EK(15)	$kg \cdot h^{-1}$	Real*4	Volatile Organic Compounds
25	EK(16)		Real*4	NO_x/NMHC ratio

B. Output Files

1. Standard Output Files

a. MF123

This file contains the hourly gridded fractions of layer 0 that are filled by plumes from line sources and/or point sources. It is used by P40G, and its parameters are shown in Table 2.89.

TABLE 2.89. MF123 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ZETA		Real*4	Plume volume fraction of layer 0

b. MF143

This file contains hourly gridded area-flux emission rates ($\mathcal{S}$) for each pollutant species in layer 0. ($\mathcal{S}$ is known as the source strength function.) MF143 is used by P40G, and its parameters are shown in Table 2.90.

TABLE 2.90. MF143 PARAMETERS

Parm No.	Parm Name	Unit [a]	Data Type	Description
				Emission rate in layer 0 for:
1	ALD2	$ppm \cdot m \cdot s^{-1}$	Real*4	Higher aldehydes
2	CO	$ppm \cdot m \cdot s^{-1}$	Real*4	Carbon monoxide
3	ETH	$ppm \cdot m \cdot s^{-1}$	Real*4	Ethylene
4	FORM	$ppm \cdot m \cdot s^{-1}$	Real*4	Formaldehyde
5	ISOP	$ppm \cdot m \cdot s^{-1}$	Real*4	Isoprene
6	NO	$ppm \cdot m \cdot s^{-1}$	Real*4	Nitric oxide
7	NO2	$ppm \cdot m \cdot s^{-1}$	Real*4	Nitrogen dioxide
8	OLE	$ppm \cdot m \cdot s^{-1}$	Real*4	Olefins
9	PAR	$ppm \cdot m \cdot s^{-1}$	Real*4	Paraffins
10	TOL	$ppm \cdot m \cdot s^{-1}$	Real*4	Toluene
11	XYL	$ppm \cdot m \cdot s^{-1}$	Real*4	Xylene
12	NONR	$ppm \cdot m \cdot s^{-1}$	Real*4	Nonreactive hydrocarbons
13	TRAC	$ppm \cdot m \cdot s^{-1}$	Real*4	Tracer species
14	MTHL	$ppm \cdot m \cdot s^{-1}$	Real*4	Methanol

a. The unit of $ppm \cdot m \cdot s^{-1}$ is formed when the original emission rate data (units of $mol \cdot h^{-1}$) is converted to the source strength function (units of $mol \cdot m^{-2} \cdot h^{-1}$), which is expressed as an area-flux. The conversion to layer 0 mixing ratios results in the unit $ppm \cdot m \cdot s^{-1}$.

c. MF144

This file contains hourly gridded emission rates (S1) for each pollutant species in layer 1. (S1 is the layer 1 analog to S of layer 0.) MF144 is used by P40G, and its parameters are shown in Table 2.91.

TABLE 2.91. MF144 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Emission rate in layer 1 for:
1	ALD2	$ppm{\cdot}s^{-1}$	Real*4	Higher aldehydes
2	CO	$ppm{\cdot}s^{-1}$	Real*4	Carbon monoxide
3	ETH	$ppm{\cdot}s^{-1}$	Real*4	Ethylene
4	FORM	$ppm{\cdot}s^{-1}$	Real*4	Formaldehyde
5	ISOP	$ppm{\cdot}s^{-1}$	Real*4	Isoprene
6	NO	$ppm{\cdot}s^{-1}$	Real*4	Nitric oxide
7	NO2	$ppm{\cdot}s^{-1}$	Real*4	Nitrogen dioxide
8	OLE	$ppm{\cdot}s^{-1}$	Real*4	Olefins
9	PAR	$ppm{\cdot}s^{-1}$	Real*4	Paraffins
10	TOL	$ppm{\cdot}s^{-1}$	Real*4	Toluene
11	XYL	$ppm{\cdot}s^{-1}$	Real*4	Xylene
12	NONR	$ppm{\cdot}s^{-1}$	Real*4	Nonreactive hydrocarbons
13	TRAC	$ppm{\cdot}s^{-1}$	Real*4	Tracer species
14	MTHL	$ppm{\cdot}s^{-1}$	Real*4	Methanol

d. MF145

This file contains hourly gridded emission rates (S2) for each pollutant species in layer 2. It is used by P40G, and its parameters are shown in Table 2.92.

TABLE 2.92. MF145 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Emission rate in layer 2 for:
1	ALD2	$ppm{\cdot}s^{-1}$	Real*4	Higher aldehydes
2	CO	$ppm{\cdot}s^{-1}$	Real*4	Carbon monoxide
3	ETH	$ppm{\cdot}s^{-1}$	Real*4	Ethylene
4	FORM	$ppm{\cdot}s^{-1}$	Real*4	Formaldehyde
5	ISOP	$ppm{\cdot}s^{-1}$	Real*4	Isoprene
6	NO	$ppm{\cdot}s^{-1}$	Real*4	Nitric oxide
7	NO2	$ppm{\cdot}s^{-1}$	Real*4	Nitrogen dioxide
8	OLE	$ppm{\cdot}s^{-1}$	Real*4	Olefins
9	PAR	$ppm{\cdot}s^{-1}$	Real*4	Paraffins
10	TOL	$ppm{\cdot}s^{-1}$	Real*4	Toluene
11	XYL	$ppm{\cdot}s^{-1}$	Real*4	Xylene
12	NONR	$ppm{\cdot}s^{-1}$	Real*4	Nonreactive hydrocarbons
13	TRAC	$ppm{\cdot}s^{-1}$	Real*4	Tracer species
14	MTHL	$ppm{\cdot}s^{-1}$	Real*4	Methanol

2. Nonstandard Output Files

None

III. CONTROL CARDS

Two control cards are used to input control data in the format shown below. Table 2.93 defines the control card variables.

BIOG
SDATE SHR EDATE EHR

TABLE 2.93. CONTROL CARD VARIABLES

Variable Name	Unit	Description
BIOG		Biogenics flag .TRUE. = use biogenics data .FALSE. = do not use biogenics data
SDATE		Julian scenario start date as *YYDDD*
SHR	EST	Scenario start time as *HH*
EDATE		Julian scenario end date as *YYDDD*
EHR	EST	Scenario end time as *HH*

Example:

.TRUE.
80194 12 80197 11

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	5	files	54	blocks
FORTRAN INCLUDE files:	5	files	15	blocks
Object files:	5	files	42	blocks
Executable file:	1	file	84	blocks
	16	files	195	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:03:09
Buffered I/O count:	287
Direct I/O count:	17167
Peak working-set size:	4511
Peak virtual size:	8093

C. Space Requirements: Log and Print Files

P10GX1.LOG:	57 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.94 shows the input file and output file space requirements.

TABLE 2.94. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF106	3	2142	72 hours
	PF107	5	1899	72 hours
	PF115	5	3798	72 hours
	PF119	4	55	N/A
	PF132	5	1899	72 hours
	PF135	4	55	N/A
	PF141	4	54	N/A
	PF144	5	13287	72 hours
	PF145	5	30369	72 hours
	PF155	4	55	N/A
	MF109	5	1899	72 hours
	MF110	5	1899	72 hours
	MF111	5	1899	72 hours
	MF112	5	1899	72 hours
	MF132	5	1899	72 hours
	MF133	5	1899	72 hours
	MF134	5	1899	72 hours
	PTE3	Nonstd	13833	72 hours
			80739	
Output	MF123	4	1899	N/A
	MF143	5	26574	72 hours
	MF144	5	26574	72 hours
	MF145	5	26574	72 hours
			81621	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P10GV1B.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P10GV1B.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P10GV1B.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P10GV1B.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.PROCES]P10G.OBJ              A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.PROCES]ADPFMF.OBJ            A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.PROCES]RDSTAK.OBJ            A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.PROCES]SATEMP.OBJ            A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.PROCES]SETCLK.OBJ            A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]ADATE.OBJ             A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]CELL0.OBJ             A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]CELL1.OBJ             A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]CELL2.OBJ             A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]CELLAR.OBJ            A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]FLSMRY.OBJ            A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]FSTAT1.OBJ            A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]IBLKR.OBJ             A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]IOCL.OBJ              A14
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]JFILE1.OBJ            A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]JFILE2.OBJ            A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]JFILE3.OBJ            A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]JHOUR1.OBJ            A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]JUNIT.OBJ             A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]OPDIR2.OBJ            A20
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]OPEN2.OBJ             A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]PROGID.OBJ            A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]RDDIR1.OBJ            A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]RDFT3.OBJ             A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]RDFT4.OBJ             A25
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]RDFT5.OBJ             A26
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]RDFHD1.OBJ            A27
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]RDFHD2.OBJ            A28
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]RDREC.OBJ             A29
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]WRFT4.OBJ             A30
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]WRFT5.OBJ             A31
$ ASSIGN/USER_MODE $2$DUA24:[ROMCMS.UTILIO]WRREC.OBJ             A32
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P10GV1B.EXE -
  /MAP = MET1:[S01S]P10GV1B.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A30, -
  A31, -
  A32
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P10GV1B.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P10GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1503A]P10GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT;14                PFMFDIR
$ ASSIGN MET1:[ROMNET1.D00000]PF119A.DAT       PF119
```

```
$ ASSIGN MET1:[ROMNET1.D000000]PF135A.DAT          PF135
$ ASSIGN MET1:[ROMNET1.D000000]PF155A.DAT          PF155
$ ASSIGN MET58:[ROMNET1.D85188]PTE3A.DAT           PTE3
$ ASSIGN MET58:[ROMNET1.D85188]PF141A.DAT          PF141
$ ASSIGN MET58:[ROMNET1.D85188]PF145A.DAT          PF145
$ ASSIGN MET56:[ROMNET1.D85188]PF144A.DAT          PF144
$ ASSIGN MET56:[ROMNET1.D85188]PF106A.DAT          PF106
$ ASSIGN MET56:[ROMNET1.D85188]PF107A.DAT          PF107
$ ASSIGN MET56:[ROMNET1.D85188]PF132A.DAT          PF132
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT          MF109
$ ASSIGN MET56:[ROMNET1.D85188]MF110A.DAT          MF110
$ ASSIGN MET56:[ROMNET1.D85188]MF111A.DAT          MF111
$ ASSIGN MET56:[ROMNET1.D85188]PF115A.DAT          PF115
$ ASSIGN MET56:[ROMNET1.D85188]MF112A.DAT          MF112
$ ASSIGN MET56:[ROMNET1.D85188]MF132A.DAT          MF132
$ ASSIGN MET56:[ROMNET1.D85188]MF133A.DAT          MF133
$ ASSIGN MET56:[ROMNET1.D85188]MF134A.DAT          MF134
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET58:[ROMNET1.D85188]MF123A.DAT          MF123
$ ASSIGN MET58:[ROMNET1.D85188]MF143A.DAT          MF143
$ ASSIGN MET58:[ROMNET1.D85188]MF144A.DAT          MF144
$ ASSIGN MET58:[ROMNET1.D85188]MF145A.DAT          MF145
$ !
$ !SYNCHRONIZE/QUE=NORM_MOD1/ENTRY=895
$ !
$ ! ASSIGN CONTROL CARDS FILE
$ TYPE USER2$DISK:[FOREMAN.FV1503A]P10GX1.CPR
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1503A]P10GX1.CPR    SYS$INPUT
$ !
$ RUN $2$DUA24:[ROMEXE.ROMNET]P10GV1B.EXE
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P10GV1B

B. Subroutines

SETCLK RDSTAK
ADPFMF SATEMP

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	FSTAT1	RDDIR1	RDFHD2
CELL0	IBLKR	RDFT3	RDREC
CELL1	JFILE1	RDFT4	WRFT4
CELL2	OPDIR2	RDFT5	WRFT5
CELLAR	OPEN2	RDFHD1	WRREC
FLSMRY	PROGID		

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL	JHOUR1
JFILE2	JUNIT
JFILE3	

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

EMISP.EXT	STKDAT.EXT
FILEID.EXT	U1.EXT
METPRM.EXT	

2.11 PROCESSOR P11G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 5 processor computes the hourly gridded wind fields from Fourier transforms for layers 1, 2, and 3, except for layer 1 during nighttime inversion conditions.

When we write the latitudinal and longitudinal components of the wind field as Fourier series, each wind observation yields a pair of equations that relate (1) the Fourier coefficients of an observed wind, and (2) the Fourier basis functions at the observation site, to the u and v components of the observed wind. The set of equations can be collected into one matrix equation $C \cdot \vec{x} = \vec{b}$, where C is the matrix of values of the Fourier basis functions (calculated in trigonometric form as opposed to exponential form), $\vec{x}$ is the vector of the Fourier coefficients of the wind field, and $\vec{b}$ is the vector of wind observations. This equation has, in practice, infinitely many solutions. P11G determines a wind field from the set of infinite solutions by (1) determining the shape of the energy spectrum curve, and (2) the constant of proportionality that determines the amplitude of this curve from the estimated kinetic energy perturbation of the wind field. Given the curve shapes and the constants of proportionality, P11G computes an approximate solution to the -3 Power Law using a convergence algorithm (Lamb 1984a, Chapter 10).

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

P11G uses 7 standard input files:

PF101 from P01G	MF109 from P07G
PF106 from P03G	MF112 from P08G
PF127 from P07G	MF113 from P08G
PF148 from P07G	

2. Nonstandard Input Files

None

B. Output Files

1. Standard Output Files

a. PF113

This file contains the hourly gridded wind field for layer 3. It is used by P12G and P29G, and its parameters are shown in Table 2.95.

TABLE 2.95. PF113 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$<u>3_{i,j,h}$	$m \cdot s^{-1}$	Real*4	Layer 3 hourly gridded west-east wind component
2	$<v>3_{i,j,h}$	$m \cdot s^{-1}$	Real*4	Layer 3 hourly gridded north-south wind component

Note: i = column number, j = row number, h = hour.

b. PF114

This file contains the hourly gridded wind field for layer 2. It is used by P12G and P29G, and its parameters are shown in Table 2.96.

TABLE 2.96. PF114 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$<u>2_{i,j,h}$	$m{\cdot}s^{-1}$	Real*4	Layer 2 hourly gridded west-east wind component
2	$<v>2_{i,j,h}$	$m{\cdot}s^{-1}$	Real*4	Layer 2 hourly gridded north-south wind component

Note: i = column number, j = row number, h = hour.

c. PF115

This file contains the hourly gridded wind field for layer 1. It is used by P10G, P12G, P29G, and P33G; its parameters are shown in Table 2.97.

TABLE 2.97. PF115 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$<u>1_{i,j,h}$	$m{\cdot}s^{-1}$	Real*4	Layer 1 hourly gridded west-east wind component
2	$<v>1_{i,j,h}$	$m{\cdot}s^{-1}$	Real*4	Layer 1 hourly gridded north-south wind component

Note: i = column number, j = row number, h = hour.

2. Nonstandard Output Files

a. The WNDOB Files

The WNDOB1, WNDOB2, and WNDOB3 files are direct access (binary) files that contain records with lengths equal to (1 + 6 x NT), where NT is the maximum number of upper-air and surface observation stations in the modeling domain (for ROMNET, NT = 300). Each of the three WNDOB files contains 72 records, each with hourly layer-averaged u and v observed wind components at each observation site for layers 1, 2, and 3 respectively.[9] The contents of the files' records 1 - 72 are shown in Table 2.98; record 0 consists of a standard PF/MF file header.

9. A single scenario consists of 72 hours.

TABLE 2.98. WNDOB1, WNDOB2, AND WNDOB3 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$NSTA_{o,h}$		Integer*4	Number of stations reporting at this hour
2	$ID_{o,h}$		Integer*4	Station identifiers
3	$ELEV_{o,h}$	m	Real*4	Station elevation
4	$LON_{o,h}$	° W	Real*4	Station longitude
5	$LAT_{o,h}$	° N	Real*4	Station latitude
6	$U_{o,h}$	$m \cdot s^{-1}$	Real*4	Hourly observed west-east wind component
7	$V_{o,h}$	$m \cdot s^{-1}$	Real*4	Hourly observed north-south wind component

Note: Parameters 2 - 7 are the 6 variables for each station making up NT; these parameters are contained within *arrays* of the data type shown. Subscripts are: o = observed and h = hour. Surface stations are distinguished (from upper-air stations) by assigning them an elevation of 9999.0 m.

b. The FT Files

The FT1, FT2, and FT3 files are direct access (binary) files that contain records with lengths equal to (10 + NA + NT + 4 x M_0), where NA (= 100) is the number of annuli used in the processor's convergence procedure, NT is the maximum number of upper-air and surface observation stations in the modeling domain (for ROMNET, NT = 300), and M_0 (= 1037) is the number of wave numbers used for the Fourier series. Each of the three FT files contains 72 records, each with hourly Fourier coefficients, as well as layer-averaged u and v computed average wind components, for layers 1, 2, and 3 respectively. The contents of the files' records 1 - 72 are shown in Table 2.99; record 0 consists of a standard PF/MF file header.

TABLE 2.99. FT1, FT2, AND FT3 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	NOITER		Integer*4	No. of iterations used
2	MU	m	Real*4	Fundamental longitudinal wavelength
3	NU	m	Real*4	Fundamental latitudinal wavelength
4	KMAX1		Integer*4	Maximum longitudinal wave number used
5	LMAX2		Integer*4	Maximum latitudinal wave number used
6	MO		Integer*4	Maximum number of wavelengths
7	MSTAT		Integer*4	Number of stations used
8	UAVG	$m \cdot s^{-1}$	Real*4	Average layer u component of wind
9	VAVG	$m \cdot s^{-1}$	Real*4	Average layer v component of wind
10	NANULI		Integer*4	Number of annuli used
11	R(NA)		Real*4	Radii of the annuli
12	ISTANO		Integer*4	Station ID
13	UR		Real*4	Fourier coefficients: Real part of u
14	VR		Real*4	Imaginary part of u
15	UI		Real*4	Real part of v
16	VI		Real*4	Imaginary part of v

Note: Parameters 11 - 16 are contained within *arrays* of the data type shown.

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 2.100 defines the control card variables.

YYDDD HH TT

TABLE 2.100. CONTROL CARD VARIABLES

Variable Name	Unit	Description
YYDDD		Julian start date as *YYDDD*
HH	EST	Start hour as *HH*
TT	h	Scenario duration

Example:

88194 12 72

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	73	files	948	blocks
FORTRAN INCLUDE files:	20	files	144	blocks
Object files:	73	files	435	blocks
Executable file:	1	file	222	blocks
	167	files	1749	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	1:37:02
Buffered I/O count:	556
Direct I/O count:	9496
Peak working-set size:	3452
Peak virtual size:	16078

C. Space Requirements: Log and Print Files

P11GX1.LOG:	288 blocks
Print Files:	528 blocks

D. Space Requirements: Input and Output Files

Table 2.101 shows the input file and output file space requirements.

TABLE 2.101. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF101	1	6465	72 hours
	PF106	3	2142	72 hours
	PF127	5	3798	72 hours
	PF148	5	1899	72 hours
	MF109	5	1899	72 hours
	MF112	5	1899	72 hours
	MF113	5	1899	72 hours
			20001	
Output	PF113	5	3798	72 hours
	PF114	5	3798	72 hours
	PF115	5	3798	72 hours
	WNDOB1	nonstd	960	72 hours
	WNDOB2	nonstd	960	72 hours
	WNDOB3	nonstd	960	72 hours
	FT1	nonstd	2574	72 hours
	FT2	nonstd	2574	72 hours
	FT3	nonstd	2574	72 hours
			21996	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P11GV1B.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P11GV1B.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P11GV1B.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMCOM.ROMNET]P11GV1B.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P11G.OBJ         A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]ALLOC.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]ALTSOL.OBJ       A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]CMPAUX.OBJ       A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]CONVRG.OBJ       A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]CPCONF.OBJ       A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]CPGRAD.OBJ       A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]CUBMIN.OBJ       A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]DIVRG.OBJ        A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]DSLMAT.OBJ       A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]DVGCLC.OBJ       A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]DVGGRD.OBJ       A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]DVMCLC.OBJ       A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]EXPCLC.OBJ       A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]FILES.OBJ        A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]FINLPT.OBJ       A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]GCALC.OBJ        A17
```

```
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]GENCLC.OBJ          A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]HESS.OBJ            A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]INITPT.OBJ          A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]LAVRG.OBJ           A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]LINMIN.OBJ          A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]MEAN.OBJ            A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P18.OBJ             A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P20.OBJ             A25
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]POW.OBJ             A26
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]PROJEC.OBJ          A27
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDHTS.OBJ           A28
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDPLOT.OBJ          A29
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]READIN11.OBJ        A30
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]REFRSH.OBJ          A31
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]SETUP11.OBJ         A32
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]SLMAT.OBJ           A33
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]TSTFN.OBJ           A34
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]UV11.OBJ            A35
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WEIGHT11.OBJ        A36
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRPLOT.OBJ          A37
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRTOBS.OBJ          A38
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]XVCCLC.OBJ          A39
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ           A40
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]BGRID2.OBJ          A41
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]BGRID3.OBJ          A42
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL0.OBJ           A43
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL1.OBJ           A44
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL2.OBJ           A45
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR3.OBJ          A46
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR6.OBJ          A47
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ          A48
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ          A49
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GETMEM.OBJ          A50
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GETYN.OBJ           A51
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ           A52
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ            A53
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ          A54
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ          A55
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ          A56
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE3.OBJ          A57
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ           A58
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]NEXTHR.OBJ          A59
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ          A60
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ           A61
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ          A62
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ          A63
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD1.OBJ          A64
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD2.OBJ          A65
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT1.OBJ           A66
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT3.OBJ           A67
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ           A68
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ           A69
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT5.OBJ           A70
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ           A71
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMCOM.ROMNET]P11GV1B.EXE -
  /MAP = MET1:[S01S]P11GV1B.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A69, -
  A70, -
  A71
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P11GV1B.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P11GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P11GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                     PFMFDIR
$ ASSIGN MET56:[ROMNET1.D85188]PF101A.DAT             PF101
$ ASSIGN MET56:[ROMNET1.D85188]PF106A.DAT             PF106
$ ASSIGN MET56:[ROMNET1.D85188]PF127A.DAT             PF127
$ ASSIGN MET56:[ROMNET1.D85188]PF148A.DAT             PF148
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT             MF109
$ ASSIGN MET56:[ROMNET1.D85188]MF112A.DAT             MF112
$ ASSIGN MET56:[ROMNET1.D85188]MF113A.DAT             MF113
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF113A.DAT             PF113
$ ASSIGN MET56:[ROMNET1.D85188]PF114A.DAT             PF114
$ ASSIGN MET56:[ROMNET1.D85188]PF115A.DAT             PF115
$ ASSIGN MET56:[ROMNET1.D85188]FT1A.DAT               FT1
$ ASSIGN MET56:[ROMNET1.D85188]FT2A.DAT               FT2
$ ASSIGN MET56:[ROMNET1.D85188]FT3A.DAT               FT3
$ ASSIGN MET56:[ROMNET1.D85188]WNDOB1A.DAT            WNDOB1
$ ASSIGN MET56:[ROMNET1.D85188]WNDOB2A.DAT            WNDOB2
$ ASSIGN MET56:[ROMNET1.D85188]WNDOB3A.DAT            WNDOB3
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARD(S) FILE
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P11GX1.CPR
$ !
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P11GX1.CPR   SYS$INPUT
$ RUN $2$DUA24:[ROMEXE.ROMNET]P11GV1B
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A        P11GX1
$ PRINT/BURST/NOFEED/HEADER/COPIES=1 MET56:[ROMNET1.D85188]P11PRTFLA.APF
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P11GV1B

B. Subroutines

ALLOC	EXPCLC	RDHTS
ALTSOL	FILES	RDPLOT
CPGRAD	FINLPT	READIN11
CUBMIN	GCALC	REFRSH
CMPAUX	GENCLC	SETUP11
CONVRG	INITPT	SLMAT
CPCONF	LAVRG	UV11
DIVRG	MEAN	WEIGHT11
DSLMAT	P18	WRPLOT
DVGCLC	P20	WRTOBS
DVGGRD	PROJEC	XVCCLC
DVMCLC		

C. Functions

HESS
LINMIN
POW
TSTFN

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	FSTAT1	RDFHD1
BGRID2	IBLKR	RDFHD2
BGRID3	JFILE1	RDFT1
CELL0	NEXTHR	RDFT3
CELL1	OPDIR2	RDFT5
CELL2	OPEN2	RDREC
DATHR6	PROGID	WRFT5
FLSMRY	RDDIR1	WRREC

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

DATHR3	JFILE2	JHOUR1
GETMEM	JFILE3	JUNIT
IOCL		

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

DESCENT.EXT	FTREC.EXT	RADII.EXT
DIVMAT.EXT	GENTRY.EXT	RAW.EXT
DVGPARMS.EXT	MATRIX.EXT	STATNS.EXT
FHEAD2.EXT	PARM11.EXT	UNITNOS.EXT

2.12 PROCESSOR P12G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 6 processor calculates the upward and downward volume fluxes between the ROM model layers. The fluxes are calculated at the layer interfaces. Parameterizations of the effects of cumulus clouds (when they are present) are included. These volume fluxes are used by the ROM in calculating the interlayer transport of pollutants.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

P12G uses 28 standard input files:

PF112 from P03G	PF152 from P17G	MF112 from P08G
PF113 from P11G	PF153 from P17G	MF113 from P08G
PF114 from P11G	PF156 from P08G	MF133 from P09G
PF115 from P11G	PF157 from P05G	MF134 from P09G
PF119 from P06G	PF160 from P32G	MF135 from P09G
PF125 from P07G	PF161 from P32G	MF160 from P09G
PF132 from P07G	MF109 from P07G	MF161 from P09G
PF142 from P08G	MF110 from P07G	MF162 from P09G
PF150 from P17G	MF111 from P07G	MF163 from P09G
PF151 from P17G		

2. Nonstandard Input Files

None

B. Output Files

1. Standard Output Files

a. MF129

This file contains the hourly gridded volume fluxes through the top surface of layer 3. MF129 is used by P40G, and its parameter is shown in Table 2.102.

TABLE 2.102. MF129 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	.H3	$m \cdot s^{-1}$	Real*4	Volume flux through top of layer 3

b. MF151

This file contains hourly gridded volume fluxes in the cumulus updraft at the base of the clouds. MF151 is used by P04G, and its parameter is shown in Table 2.103.

TABLE 2.103. MF151 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	V3	$m \cdot s^{-1}$	Real*4	Cumulus updraft volume flux at cloud base

c. MF152

This file contains hourly gridded downward volume fluxes of air (whose source lies outside the cumulus roots) through the top surface of layer 2. MF152 is used by P40G, and its parameter is shown in Table 2.104.

TABLE 2.104. MF152 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	W2MNS	$m \cdot s^{-1}$	Real*4	Downward volume flux through top of layer 2

d. MF153

This file contains hourly gridded upward volume fluxes of air (whose source lies outside the cumulus roots) through the top of layer 2. MF153 is used by P40G, and its parameter is shown in Table 2.105.

TABLE 2.105. MF153 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	W2PLS	$m \cdot s^{-1}$	Real*4	Upward volume flux through top of layer 2

e. MF154

This file contains hourly gridded downward volume fluxes of air (whose source lies outside the cumulus roots) through the top of layer 1. MF154 is used by P40G, and its parameter is shown in Table 2.106.

TABLE 2.106. MF154 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	W1MNS	$m \cdot s^{-1}$	Real*4	Downward volume flux through top of layer 1

f. MF155

This file contains hourly gridded upward volume fluxes of air (whose source lies outside the cumulus roots) through the top of layer 1. MF153 is used by P40G, and its parameter is shown in Table 2.107.

TABLE 2.107. MF155 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	W1PLS	$m \cdot s^{-1}$	Real*4	Upward volume flux through top of layer 1

g. MF156

This file contains hourly gridded downward volume fluxes of air (whose source lies outside the cumulus roots) through the top of layer 0. MF156 is used by P40G, and its parameter is shown in Table 2.108.

TABLE 2.108. MF156 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	W0MNS	$m \cdot s^{-1}$	Real*4	Downward volume flux through top of layer 0

h. MF157

This file contains hourly gridded upward volume fluxes of air (whose source lies outside the cumulus roots) through the top of layer 0. MF157 is used by P40G, and its parameter is shown in Table 2.109.

TABLE 2.109. MF157 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	W0PLS	$m \cdot s^{-1}$	Real*4	Upward volume flux through top of layer 0

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 2.110 defines the control card variables.

IDATE IHRBGN NUMHRS

TABLE 2.110. CONTROL CARD VARIABLES

Variable Name	Unit	Description
IDATE		Julian scenario start date as *YYDDD*
IHRBGN	EST	Scenario start time as *HH*
NUMHRS	h	Number of hours to process

Example:

80194 12 72

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	6	files	100	blocks
FORTRAN INCLUDE files:	4	files	29	blocks
Object files:	6	files	40	blocks
Executable file:	1	file	73	blocks
	17	files	242	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:40:31
Buffered I/O count:	455
Direct I/O count:	13369
Peak working-set size:	3783
Peak virtual size:	6802

C. Space Requirements: Log and Print Files

P12GX1.LOG:	±204 blocks (varies)
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.111 shows the input file and output file space requirements.

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P12GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P12GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P12GV1A.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P12GV1A.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P12G.OBJ          A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]ALLFLX.OBJ        A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CUFLUX.OBJ        A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CUSET.OBJ         A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]ERF.OBJ           A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]VOLFLX.OBJ        A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ         A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELLO.OBJ         A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]DATHR3.OBJ        A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ        A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ        A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ         A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ          A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ        A14
```

TABLE 2.111. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF112	5	1898	72 hours
	PF113	5	3770	72 hours
	PF114	5	3770	72 hours
	PF115	5	3770	72 hours
	PF119	5	52	72 hours
	PF125	5	1898	72 hours
	PF132	5	1898	72 hours
	PF142	5	1898	72 hours
	PF150	5	1898	72 hours
	PF151	5	1898	72 hours
	PF152	5	1898	72 hours
	PF153	5	1898	72 hours
	PF156	5	1898	72 hours
	PF157	5	1898	72 hours
	PF160	5	1898	72 hours
	PF161	5	1898	72 hours
	MF109	5	1898	72 hours
	MF110	5	1898	72 hours
	MF111	5	1898	72 hours
	MF112	5	1898	72 hours
	MF113	5	1898	72 hours
	MF133	5	1898	72 hours
	MF134	5	1898	72 hours
	MF135	5	1898	72 hours
	MF160	5	1898	72 hours
	MF161	5	1898	72 hours
	MF162	5	1898	72 hours
	MF163	5	1898	72 hours
			56914	
Output	MF129	5	1898	72 hours
	MF151	5	1898	72 hours
	MF152	5	1898	72 hours
	MF153	5	1898	72 hours
	MF154	5	1898	72 hours
	MF155	5	1898	72 hours
	MF156	5	1898	72 hours
	MF157	5	1898	72 hours
			15184	

```
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JHOUR1.OBJ                  A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ                   A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]NEXTHR.OBJ                  A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ                  A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ                   A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ                  A20
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ                  A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFHD1.OBJ                  A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ                   A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT4.OBJ                   A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT5.OBJ                   A25
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT5.OBJ                   A26
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ                   A27
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P12GV1A.EXE -
  /MAP = MET1:[S01S]P12GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A25, -
  A26, -
  A27
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P12GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P12GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P12GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                   PFMFDIR
$ ASSIGN MET56:[ROMNET1.D85188]PF112A.DAT           PF112
$ ASSIGN MET56:[ROMNET1.D85188]PF113A.DAT           PF113
$ ASSIGN MET56:[ROMNET1.D85188]PF114A.DAT           PF114
$ ASSIGN MET56:[ROMNET1.D85188]PF115A.DAT           PF115
$ ASSIGN MET1:[ROMNET1.D00000]PF119A.DAT            PF119
$ ASSIGN MET56:[ROMNET1.D85188]PF125A.DAT           PF125
$ ASSIGN MET56:[ROMNET1.D85188]PF132A.DAT           PF132
$ ASSIGN MET56:[ROMNET1.D85188]PF142A.DAT           PF142
$ ASSIGN MET56:[ROMNET1.D85188]PF150A.DAT           PF150
$ ASSIGN MET56:[ROMNET1.D85188]PF151A.DAT           PF151
$ ASSIGN MET56:[ROMNET1.D85188]PF152A.DAT           PF152
$ ASSIGN MET56:[ROMNET1.D85188]PF153A.DAT           PF153
$ ASSIGN MET56:[ROMNET1.D85188]PF156A.DAT           PF156
$ ASSIGN MET56:[ROMNET1.D85188]PF157A.DAT           PF157
$ ASSIGN MET56:[ROMNET1.D85188]PF160A.DAT           PF160
$ ASSIGN MET56:[ROMNET1.D85188]PF161A.DAT           PF161
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT           MF109
$ ASSIGN MET56:[ROMNET1.D85188]MF110A.DAT           MF110
$ ASSIGN MET56:[ROMNET1.D85188]MF111A.DAT           MF111
$ ASSIGN MET56:[ROMNET1.D85188]MF112A.DAT           MF112
$ ASSIGN MET56:[ROMNET1.D85188]MF113A.DAT           MF113
$ ASSIGN MET56:[ROMNET1.D85188]MF133A.DAT           MF133
$ ASSIGN MET56:[ROMNET1.D85188]MF134A.DAT           MF134
$ ASSIGN MET56:[ROMNET1.D85188]MF135A.DAT           MF135
$ ASSIGN MET56:[ROMNET1.D85188]MF160A.DAT           MF160
$ ASSIGN MET56:[ROMNET1.D85188]MF161A.DAT           MF161
$ ASSIGN MET56:[ROMNET1.D85188]MF162A.DAT           MF162
```

```
$ ASSIGN MET56:[ROMNET1.D85188]MF163A.DAT          MF163
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]MF129A.DAT         MF129
$ ASSIGN MET56:[ROMNET1.D85188]MF151A.DAT         MF151
$ ASSIGN MET56:[ROMNET1.D85188]MF152A.DAT         MF152
$ ASSIGN MET56:[ROMNET1.D85188]MF153A.DAT         MF153
$ ASSIGN MET56:[ROMNET1.D85188]MF154A.DAT         MF154
$ ASSIGN MET56:[ROMNET1.D85188]MF155A.DAT         MF155
$ ASSIGN MET56:[ROMNET1.D85188]MF156A.DAT         MF156
$ ASSIGN MET56:[ROMNET1.D85188]MF157A.DAT         MF157
$ SHOW PROCESS / ACCOUNTING
$ !
$ !SYNCHRONIZE/ENTRY=578
$ ! ASSIGN CONTROL CARDS FILE
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P12GX1.CPR
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P12GX1.CPR     SYS$INPUT
$ !
$ RUN $2$DUA24:[ROMEXE.ROMNET]P12GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A                 P12GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P12GV1A

B. Subroutines

ALLFLX
CUFLUX
CUSET
VOLFLX

C. Functions

ERF

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	JFILE1	PROGID	RDFT5
CELL0	NEXTHR	RDDIR1	RDREC
FLSMRY	OPDIR2	RDFHD1	WRFT5
FSTAT1	OPEN2	RDFT4	WRREC
IBLKR			

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

DATHR3
IOCL
JHOUR1
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

GRIDD.EXT
INTRNL.EXT
MFOUT.EXT
PFMFIN.EXT

2.13 PROCESSOR P13G (EMISSIONS)

I. PROCESSOR FUNCTION

This stage 0 processor uses the data output from the LINE_C and LINE_CR preprocessors to produce a gridded total line-source length for each grid cell in the modeling domain; note that P13G needs to be run only one time per modeling domain. P13G selects and can weight the importance of the roads from among the 81 total possibilities contained in the original U.S. GeoData set (and the input file) with respect to their importance as line sources. The selection and weighting code is shown below; the real numbers represent the weighting factors.

```
DATA TFAC /4*1.0, 8*0.0, 19*1.0, 49*0.0, 1.0/
```

The line sources contained in the output file are as follows: the first four sources are U.S. Interstates or other major highways; roads 13 - 31 are minor U.S. and state roads; the final entry is the main-line railroads. Note that this version of the ROM does not account for Canadian line sources. However, you may decide that you want to include these data; in this case, we suggest that you use a digitizer to capture Canadian roads, and include the data in the input file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

P13G uses one nonstandard file, LINES1.DAT, that is generated by LINE_CR. This file is read by the READ and FORMAT statements listed below, and its parameters are shown in Table 2.112.

```
     READ (NRD1,100,END=40) LID,NPTS,IC,IR
     READ (NRD1,200,END=40) XLT1,XLN1
     READ (NRD1,200) XLT1,XLN1
100  FORMAT (I7,I6,I5,I2)
200  FORMAT (2F9.4)
```

TABLE 2.112. LINES1.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	LID		Integer*4	Sequential line number
2	$NPTS_l$		Integer*4	Number of nodes in this line
3	IC_l		Integer*4	Line attribute major code
4	IR_l		Integer*4	Line attribute minor code (rank)
5	$XLT1_1$	° N	Real*4	Latitude of first node of line segment
6	$XLN1_1$	° W	Real*4	Longitude of first node of line segment
.	.			.
.	.			.
.	.			.
N	$XLT1_N$	° N	Real*4	Latitude of N^{th} node of line segment
N	$XLN1_N$	° W	Real*4	Longitude of N^{th} node of line segment

Note: l = line identifier number; major codes are all 29050; minor codes are two-digit codes that correspond to the final two numbers in the minor codes listed in Appendix A, Domaratz *et al.*, (1983).

B. Output Files

1. Standard Output Files

P13G generates one standard file, PF135, that contains gridded line-source lengths; the file is used by P10G, and its parameter are shown in Table 2.113.

TABLE 2.113. P135 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$L_{i,j}$	m	Real*4	Total cell line-source length

Note: i = column number, j = row number

2. Nonstandard Output Files

None

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	3	files	13	blocks
FORTRAN INCLUDE files:	1	files	3	blocks
Object files:	3	files	12	blocks
Executable file:	1	file	43	blocks
	8	files	71	blocks

B. Execution Time Requirements

	VAX 8650
Charged CPU time (hh:mm:ss):	00:02:46
Buffered I/O count:	133
Direct I/O count:	921
Peak working-set size:	618
Peak virtual size:	965

C. Space Requirements: Log and Print Files

P13GX1.LOG:	9 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.114 shows the input file and output file space requirements.

TABLE 2.114. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	LINES1.DAT	Nonstd	11469	N/A
Output	PF135	4	40	N/A

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P13GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P13GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P13GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P13GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P13G.OBJ          A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RBCLIP.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]SEGSUB.OBJ        A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ         A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL0.OBJ         A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL1.OBJ         A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ        A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ        A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ         A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ          A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ        A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ        A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ         A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ        A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ         A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ        A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ        A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ         A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT4.OBJ         A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ         A20
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P13GV1A.EXE -
  /MAP = MET1:[S01S]P13GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A18, -
  A19, -
  A20
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P13GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P13GX1.COM follows:

(end of run stream)

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1050A]P13GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]LINES1.DAT         LINES1
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET1:[ROMNET.D00000]PF135A.DAT          PF135
$ SHOW PROCESS / ACCOUNTING
$ !
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P13GV1A.EXE
$ SHOW PROCESS / ACCOUNTING
```

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P13GV1A

B. Subroutines

RBCLIP
SEGSUB

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	IBLKR	OPDIR2	RDDIR1
CELL0	IOCL	OPEN2	RDREC
CELL1	JFILE1	PROGID	WRFT4
FLSMRY	JUNIT	RBCLIP	WRREC
FSTAT1			

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JFILE2

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

2.14 PROCESSOR P14G (EMISSIONS)

I. PROCESSOR FUNCTION

This stage 2 processor generates a three-day emissions scenario (72 hours) from point-source and area-source emissions data, and, optionally, from mobile-source emissions data. P14G determines the day type (weekday, Saturday, or Sunday) of the start of the scenario, and uses the appropriate hourly/daily/seasonal emissions data to create the output files. These files are produced using the following steps:

- Major point sources (as designated by preprocessor PTEXTR) are written to a sequential file;
- Minor point sources (also designated by PTEXTR) are merged with the area-source emissions on a gridded, hourly basis;
- Scenario mobile-source emissions (from P26G) are optionally merged (via a control card) with the minor point-source and area-source emissions;
- All emissions units, if not already standardized, are converted to $mol \cdot h^{-1}$, except for volatile organic compounds (VOC), which have units of $kg \cdot h^{-1}$;
- Two scenario-specific files are written, one containing the major point-source emissions and stack parameters, the other the merged point-, area-, and mobile-source emissions.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

PF167 from P26G

2. Nonstandard Input Files

P14G uses seven nonstandard input files; there are three day-specific files for both point-source and area-source emissions, and one file that contains the stack parameters of the major point sources.

a. AREASA.CS0, AREASU.CS0, and AREAWK.CS0

These are the day-specific area-source emissions data files, generated by P34G, for Saturdays, Sundays, and weekdays respectively. The READ and FORMAT statements for these files are listed below, and their parameters are shown in Table 2.115.

```
      READ(UNIT1,90,IOSTAT=IOST)COL, ROW, JTIMEA,
                                (STEA(I),I=1,NPOLIN)
90    FORMAT(2I4,I2,17F10.0)
```

TABLE 2.115. AREASA.CS0, AREASU.CS0, AND AREAWK.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Integer*4	Column number
2	ROW		Integer*4	Row number
3	JTIMEA	LST	Integer*4	Hour as *HH*
4	$STEA(1)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO_x
5	$STEA(2)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for VOC
6	$STEA(3)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for THC
7	$STEA(4)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for CO
8	$STEA(5)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for olefins
9	$STEA(6)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for paraffins
10	$STEA(7)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for toluene
11	$STEA(8)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for xylene
12	$STEA(9)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for formaldehyde
13	$STEA(10)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for aldehydes
14	$STEA(11)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for ethene
15	$STEA(12)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for isoprenes
16	$STEA(13)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for nonr.
17	$STEA(14)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for methane
18	$STEA(15)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO
19	$STEA(16)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO_2
20	$STEA(17)_{i,j,h}$	$mol \cdot kg^{-1}$	Real*4	Emission rate for methanol

Note: i = column number; j = row number; h = hour; VOC = volatile organic compounds; THC = total hydrocarbons; nonr. = nonreactive hydrocarbons.

b. PNTSA.CS0, PNTSU.CS0, and PNTWK.CS0

These are the day-specific point-source emissions binary data files, generated by P34G, for Saturdays, Sundays, and weekdays respectively. The READ statement for these files are listed below, and their parameters are shown in Table 2.116.

```
READ(UNIT1,IOSTAT=IOST)STACK, JTIME, (SEK(I),I=1,PNTPOL)
```

TABLE 2.116. PNTSA.CS0, PNTSU.CS0, AND PNTWK.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	STACK		Char*12	AEROS state/county/plant/point ID
2	JTIME	LST	Integer*4	Hour as *HH*
3	$SEK(1)_{p,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for VOC
4	$SEK(2)_{p,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for CO
5	$SEK(3)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for olefins
6	$SEK(4)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for paraffins
7	$SEK(5)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for toluene
8	$SEK(6)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for xylene
9	$SEK(7)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for formaldehyde
10	$SEK(8)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for aldehydes
11	$SEK(9)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for ethene
12	$SEK(10)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for nonr.
13	$SEK(11)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for isoprenes
14	$SEK(12)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for methane
15	$SEK(13)_{p,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for NO
16	$SEK(14)_{p,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for NO_2
17	$SEK(15)_{p,h}$	$mol{\cdot}kg^{-1}$	Real*4	Emission rate for methanol

Note: p = point source; h = hour; VOC = volatile organic compounds; nonr. = nonreactive hydrocarbons.

c. STACKS.CS0

STACKS.CS0 contains the point-source stack and locational parameters, and is generated by preprocessor PTEXTR. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 2.117.

```
      READ(IDEV1,170,IOSTAT=IOST)SHFLG(I), STKID(I), PSLON(I),
     $     PSLAT(I), EXFL(I), PLMT(I), DK(I), TK(I), WK(I), ZSK(I)
  170 FORMAT(I1, T4, A12, T18, 8F10.0)
```

TABLE 2.117. STACKS.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$SHFLG_p$		Integer*4	Major/minor point source flag: 0 = major; 1 = minor
2	$STKID_p$		Char*12	AEROS state/county/plant/point ID
3	$PSLON_p$	° W	Real*4	Longitude (fractional)
4	$PSLAT_p$	° N	Real*4	Latitude (fractional)
5	$EXFL_p$	$m^3 \cdot s^{-1}$	Real*4	Exhaust flow rate
6	$PLMT_p$	m	Real*4	Plume height
7	DK_p	m	Real*4	Stack diameter
8	TK_p	°C	Real*4	Exhaust gas temperature
9	WK_p	$m \cdot s^{-1}$	Real*4	Exhaust gas exit velocity
10	ZSK_p	m	Real*4	Stack height

Note: p = point source.

B. Output Files

1. Standard Output Files

a. PF141

PF141 contains the number of minor point sources in each grid cell; it is used by P10G, and its parameter is shown in Table 2.118.

TABLE 2.118. PF141 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$NMP_{i,j}$		Real*4	No. of minor point sources in cell

Note: i = column number, j = row number.

b. PF145

PF145 contains hourly gridded minor point-source, area-source, and, optionally, mobile-source emissions data; it is used by P10G, and its parameters are shown in Table 2.119.

TABLE 2.119. PF145 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$AEMIS(1)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for aldehydes
2	$AEMIS(2)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for CO
3	$AEMIS(3)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for ethene
4	$AEMIS(4)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for formaldehyde
5	$AEMIS(5)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for isoprenes
6	$AEMIS(6)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for NO
7	$AEMIS(7)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for NO_2
8	$AEMIS(8)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for olefins
9	$AEMIS(9)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for paraffins
10	$AEMIS(10)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for toluene
11	$AEMIS(11)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for xylene
12	$AEMIS(12)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for nonr.
13	$AEMIS(13)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methane
14	$AEMIS(14)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methanol
15	$AEMIS(15)_{i,j,h}$	$kg \cdot h^{-1}$	Real*4	Emission rate for VOC
16	$AEMIS(16)_{i,j,h}$		Real*4	$NMHC/NO_x$ ratio

Note: i = column number, j = row number, h = hour; nonr. = nonreactive hydrocarbons, VOC = volatile organic compounds, NMHC = nonmethane hydrocarbons.

2. Nonstandard Output Files

P14G generates one binary output file, PTE3.DAT, that contains the major point-source emissions data. The WRITE statement for this file is listed below, and its parameters are shown in Table 2.120.

```
      WRITE(IDEV1) STACK, SLON, SLAT, SXFL, SPLM, SDK, SZSK,
     &             STK, SWK, (SEK(I), I=1,NPOL)
```

III. CONTROL CARDS

Three control cards are used to input control data in the format shown below. Table 2.121 defines the control card variables.

NHR
ISDATE, ISTIME
MOBLFLG

TABLE 2.120. PTE3.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$STACK_p$		Char*12	AEROS state/county/plant/point ID
2	$SLON_p$	° W	Real*4	Longitude (fractional)
3	$SLAT_p$	° N	Real*4	Latitude (fractional)
4	$SXFL_p$	$m^3 \cdot s^{-1}$	Real*4	Exhaust flow rate
5	$SPLM_p$	m	Real*4	Plume height
6	SDK_p	m	Real*4	Stack diameter
7	$SZSK_p$	m	Real*4	Stack height
8	STK_p	°C	Real*4	Exhaust gas temperature
9	SWK_p	$m \cdot s^{-1}$	Real*4	Exhaust gas exit velocity
10	$SEK(1)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for aldehydes
11	$SEK(2)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for CO
12	$SEK(3)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for ethene
13	$SEK(4)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for formaldehyde
14	$SEK(5)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for isoprenes
15	$SEK(6)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for NO
16	$SEK(7)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for NO_2
17	$SEK(8)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for olefins
18	$SEK(9)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for paraffins
19	$SEK(10)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for toluene
20	$SEK(11)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for xylene
21	$SEK(12)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for nonr.
22	$SEK(13)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methane
23	$SEK(14)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methanol
24	$SEK(15)_{p,h}$	$kg \cdot h^{-1}$	Real*4	Emission rate for VOC
25	$SEK(16)_{p,h}$		Real*4	NMHC/NO_x ratio

Note: p = point source, h = hour; nonr. = nonreactive hydrocarbons, VOC = volatile organic compounds, NMHC = nonmethane hydrocarbons.

TABLE 2.121. CONTROL CARD VARIABLES

Variable Name	Unit	Description
NHR		Number of hours in the scenario
ISDATE		Julian scenario start date as *YYDDD*
ISTIME	EST	Scenario start time as *HH*
MOBLFLG		Flag for mobile-source emissions: .TRUE. ⇒ include .FALSE. ⇒ exclude

Example:

72
88184,12
.TRUE.

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	8	files	81	blocks
FORTRAN INCLUDE files:	8	files	34	blocks
Object files:	8	files	45	blocks
Executable file:	1	file	975	blocks
	25	files	1135	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:24:24
Buffered I/O count:	458
Direct I/O count:	25876
Peak working-set size:	4483
Peak virtual size:	6708

C. Space Requirements: Log and Print Files

P14G.LOG:	48 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.122 shows the input file and output file space requirements.

TABLE 2.122. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF167	5	30369	72 hours
	AREASA.DAT	Nonstd	28392	daily
	AREASU.DAT	Nonstd	28392	daily
	AREAWK.DAT	Nonstd	28392	daily
	PNTSA.DAT	Nonstd	57960	daily
	PNTSU.DAT	Nonstd	57960	daily
	PNTWK.DAT	Nonstd	57960	daily
	STACKS	Nonstd	3021	
			292446	
Output	PF141	4	54	
	PF145	5	26574	72 hours
	PTE3.DAT	Nonstd	14571	72 hours
			41199	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P14GV1B.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P14GV1B.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P14GV1B.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P14GV1B.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]P14G.OBJ          A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]ARESEL.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]PONSEL.OBJ        A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]RDPFMF.OBJ        A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]STKHSH.OBJ        A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]STKMRG.OBJ        A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]SWAPEM.OBJ        A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]SWAPSTK.OBJ       A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ         A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL0.OBJ         A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL1.OBJ         A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CHAIN2.OBJ        A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATEHR.OBJ        A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR4.OBJ        A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ        A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSKIP1.OBJ        A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ        A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ         A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]INDEX1.OBJ        A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ          A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ        A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ        A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE4.OBJ        A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ        A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ         A25
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]KDAY.OBJ          A26
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]MATCH.OBJ         A27
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ        A28
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ         A29
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ        A30
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ        A31
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD1.OBJ        A32
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD2.OBJ        A33
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ         A34
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ         A35
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT5.OBJ         A36
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ         A37
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT4.OBJ         A38
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P14GV1B.EXE -
  /MAP = MET1:[S01S]P14GV1B.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A36, -
  A37, -
  A38
$ PRINT -
  /NOHEADER -
```

```
/NOFEED -
/NOTIFY -
MET1:[S01S]P14GV1B.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P14GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1873A]P14GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! Assign Data Base:
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                PFMFDIR
$ ! Assign point source input files:
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]STACKS.CSO    STACK
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]PNTWK.CSO     PNTWK
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]PNTSA.CSO     PNTSA
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]PNTSU.CSO     PNTSU
$ ! Assign area source input files:
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]AREAWK.CSO    AREAWK
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]AREASA.CSO    AREASA
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]AREASU.CSO    AREASU
$ ! Assign mobile source input files:
$ ASSIGN MET57:[ROMNET1.D88184]PF167A.DAT        PF167
$ ! Assign output files:
$ ASSIGN MET57:[ROMNET1.D88184]PTE3A.DAT         PTE3
$ ASSIGN MET57:[ROMNET1.D88184]PF145A.DAT        PF145
$ ASSIGN MET57:[ROMNET1.D88184]PF141A.DAT        PF141
$ !
$ ! ASSIGN CONTROL CARD(S) FILE
$ TYPE USER2$DISK:[FOREMAN.FV1873A]P14GX1.CPR
$ !
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1873A]P14GX1.CPR   SYS$INPUT
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P14GV1B.EXE
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P14GV1B

B. Subroutines

ARESEL	STKHSH	SWAPEM
PONSEL	STKMRG	SWAPSTK
RDPFMF		

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	FSKIP1	KDAY	RDFHD2
CELL0	IBLKR	OPDIR2	RDFT5
CELL1	IOCL	OPEN2	RDREC
CHAIN2	JFILE1	PROGID	WRFT4
DATEHR	JFILE4	RDDIR1	WRFT5
DATHR4	JHOUR1	RDFHD1	WRREC
FLSMRY	JUNIT		

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

INDEX1
JFILE2
KDAY

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

EMISAR.EXT	PRID.EXT
EMISPAR.EXT	STKDATA.EXT
HASH1.EXT	STKDAT2.EXT
ISDATE.EXT	

2.15 PROCESSOR P15G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 6 processor produces the hourly gridded values of effective deposition velocities for ten chemical species. The hourly effective bulk deposition resistances are first calculated over the grid, and are used in conjunction with the calculated aerodynamic and quasilaminar sublayer resistances to estimate the hourly gridded effective deposition velocities. The effective deposition velocities for these ten species represent the effective deposition velocities for the ROM's full complement of 35 chemical species.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

PF103 from P04G	PF108 from P04G	PF118 from P25G
PF106 from P03G	PF109 from P04G	MF139 from P09G
PF107 from P04G	PF110 from P04G	MF140 from P09G

2. Nonstandard Input Files

None

B. Output Files

1. Standard Output Files

P15G writes MF146, which contains hourly gridded effective deposition velocities for ten chemical species. MF146 is used by P40G, the B-matrix compiler, and Table 2.123 shows its parameters.

TABLE 2.123. MF146 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Hourly gridded effective deposition velocity for:
1	ALD2	$m \cdot s^{-1}$	Real*4	Higher aldehydes
2	CO	$m \cdot s^{-1}$	Real*4	Carbon monoxide
3	FORM	$m \cdot s^{-1}$	Real*4	Formaldehyde
4	H2O2	$m \cdot s^{-1}$	Real*4	Hydrogen peroxide
5	HNO2	$m \cdot s^{-1}$	Real*4	Nitrous acid
6	HNO3	$m \cdot s^{-1}$	Real*4	Nitric acid
7	NO	$m \cdot s^{-1}$	Real*4	Nitric oxide
8	NO2	$m \cdot s^{-1}$	Real*4	Nitrogen dioxide
9	O3	$m \cdot s^{-1}$	Real*4	Ozone
10	PAN	$m \cdot s^{-1}$	Real*4	Peroxyacetyl nitrate

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 2.124 defines the control card variables.

IDATE0 IHR0 NUMHRS ISESN

TABLE 2.124. CONTROL CARD VARIABLES

Variable Name	Unit	Description
IDATE0		Julian scenario start date as *YYDDD*
IHR0	EST	Scenario start time as *HH*
NUMHRS		Number of hours to process
ISESN		Seasonal category flag: 1 Midsummer with lush vegetation 2 Fall with unharvested cropland 3 Late fall after frost, no snow 4 Winter with snow on the ground 5 Transitional spring

Example:

88194 12 72 1

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	58	blocks
FORTRAN INCLUDE files:	6	files	40	blocks
Object files:	2	files	20	blocks
Executable file:	1	file	77	blocks
	11	files	195	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:13:58
Buffered I/O count:	284
Direct I/O count:	4929
Peak working-set size:	2066
Peak virtual size:	4425

C. Space Requirements: Log and Print Files

P15GX1.LOG:	41 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.125 shows the input file and output file space requirements.

TABLE 2.125. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF103	5	5642	72 hours
	PF106	3	2140	72 hours
	PF107	5	1898	72 hours
	PF108	4	52	72 hours
	PF109	5	1898	72 hours
	PF110	5	1898	72 hours
	PF118	4	286	72 hours
	MF139	5	1898	72 hours
	MF140	5	1898	72 hours
			17610	
Output	MF146	5	18746	72 hours

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P15GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P15GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P15GV1A.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P15GV1A.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P15G.OBJ                A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]RST.OBJ                 A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ               A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELLO.OBJ               A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ              A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ              A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ               A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ                A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ              A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE2.OBJ              A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JHOUR1.OBJ              A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ               A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]NEXTHR.OBJ              A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ              A14
```

```
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ                  A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ                 A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ                 A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFHD1.OBJ                 A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFHD2.OBJ                 A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT3.OBJ                  A20
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT4.OBJ                  A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT5.OBJ                  A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ                  A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT5.OBJ                  A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ                  A25
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P15GV1A.EXE -
  /MAP = MET1:[S01S]P15GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A23, -
  A24, -
  A25
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P15GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P15GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P15GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                   PFMFDIR
$ ASSIGN MET56:[ROMNET1.D85188]PF103A.DAT           PF103
$ ASSIGN MET56:[ROMNET1.D85188]PF106A.DAT           PF106
$ ASSIGN MET56:[ROMNET1.D85188]PF107A.DAT           PF107
$ ASSIGN MET56:[ROMNET1.D85188]PF108A.DAT           PF108
$ ASSIGN MET56:[ROMNET1.D85188]PF109A.DAT           PF109
$ ASSIGN MET56:[ROMNET1.D85188]PF110A.DAT           PF110
$ ASSIGN MET1:[ROMNET1.D00000]PF118A.DAT            PF118
$ ASSIGN MET56:[ROMNET1.D85188]MF139A.DAT           MF139
$ ASSIGN MET56:[ROMNET1.D85188]MF140A.DAT           MF140
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]MF146A.DAT           MF146
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARD(S) FILE
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P15GX1.CPR
$ !
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P15GX1.CPR   SYS$INPUT
$ RUN $2$DUA24:[ROMEXE.ROMNET]P15GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A      P15GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P15GV1A

B. Subroutines

RST

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	JFILE1	RDDIR1	RDFT5
CELL0	NEXTHR	RDFHD1	RDREC
FLSMRY	OPDIR2	RDFHD2	WRFT5
FSTAT1	OPEN2	RDFT3	WRREC
IBLKR	PROGID	RDFT4	

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JHOUR1
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

GRIDEF.EXT
INPRES.EXT
INPUTS.EXT
PFILE6.EXT
RESIST.EXT

2.16 PROCESSOR P16G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 1 processor produces hourly vertical profiles from observed rawinsonde soundings and surface meteorology observations. This is accomplished by first interpolating sounding data to a 25-mb resolution, horizontally filtering these data using the Barnes objective analysis scheme (Barnes, 1973), and vertically filtering the result. Linear interpolation is then performed on the filtered soundings to produce hourly profiles. The hourly surface observations, after smoothing by the Barnes scheme, are incorporated into the interpolated profiles. P16G then checks the profiles to ensure that they violate no physical principles.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. RAWIN1

RAWIN1 contains rawinsonde upper-air soundings at 12-hour intervals. The file is produced from raw data tapes that undergo extensive processing and quality control (refer to Meteorology Preprocessors, Section 3, Part 1). RAWIN1 is read using the READ and FORMAT statements listed below, and its parameters are shown in Table 2.126.

```
      READ(RAWIN1,200) IYR,IMO,IDA,IHR,IDST,LEV,LEVS,PRES,
     &                 HTP,TP,DP,RH,DIR,SPD,XLAT,XLON
200   FORMAT (4I3,I6,2X,2I2,9F8.0)
```

b. RAWSTA1

RAWSTA1 contains the land use classification of each rawind (upper-air) station in the domain. The READ statements for this file are as follows:

```
      READ (STAT, *, IOSTAT = IOST) GRIDNM, YEARNM
      READ (STAT, *, IOSTAT = IOST) IDRAD(N), TMP1, TMP2, TMP3,
                                    (LNDUSE(JJ,n), JJ=1,10)
```

Table 2.127 shows the RAWSTA1 parameters.

TABLE 2.126. RAWIN1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IYR		Integer*4	Scenario year (*YY*)
2	IMO		Integer*4	Scenario month (*MM*)
3	IDA		Integer*4	Scenario day (*DD*)
4	IHR	EST	Integer*4	Scenario hour (*HH*)
5	IDST		Integer*4	Station ID
6	LEV		Integer*4	Level of the data
7	LEVS		Integer*4	No. levels in sounding
8	$PRES_{m,i}$	mb	Real*4	Pressure
9	$HTP_{m,i}$	m	Real*4	Geopotential height
10	$TP_{m,i}$	°C	Real*4	Temperature
11	$DP_{m,i}$	°C	Real*4	Dew point
12	$RH_{m,i}$	%	Real*4	Relative humidity
13	$DIR_{m,i}$	deg	Real*4	Wind direction
14	$SPD_{m,i}$	$m \cdot s^{-1}$	Real*4	Wind speed
15	$XLAT_m$	° N	Real*4	Latitude
16	$XLON_m$	° W	Real*4	Longitude

Note: m = upper-air station number (one or more digits); i = level number in 25 mb intervals (one or more digits).

TABLE 2.127. RAWSTA1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	GRIDNM		Char*8	Region name (see Appendix A)
2	YEARNM		Integer*4	Scenario year (*YY*)
3	$IDRAD_m$		Integer*4	Station WBAN identifier
4	$TMP1_m$	° N	Real*4	Station latitude
5	$TMP2_m$	° W	Real*4	Station longitude
6	$TMP3_m$	m	Real*4	Station elevation
7	$LNDUSE_m$		Real*4	Land use classification:
	URBAN			urban
	AGRI			agriculture
	RANGE			rangeland
	DF			deciduous forest
	CF			coniferous forest
	MF			mixed forests including wetlands
	WATER			water
	BARREN			barren land
	NFW			nonforested wetland
	MAR			mixed agriculture/rangeland

Note: m = upper-air station number (one or more digits).

c. SURMET1

SURMET1 contains hourly surface meteorology observations processed from the National Climatic Data Center and the Canadian Climate Centre raw data tapes. SURMET1 is read in two stages. The first stage reads header information, and the first READ and FORMAT statements for this file are as follows:

```
      READ (NSFC,200) IYR,IMON,IDA,IHR,NUMSTA
200   FORMAT (4I2,I8)
```

Table 2.128 shows the SURMET1 Stage 1 parameters.

TABLE 2.128. SURMET1 STAGE 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IYR		Integer*4	Scenario year *(YY)*
2	IMON		Integer*4	Scenario month *(MM)*
3	IDA		Integer*4	Scenario day *(DD)*
4	IHR	EST	Integer*4	Scenario hour *(HH)*
5	NUMSTA		Integer*4	No. of reporting stations at *HH*

The Stage 2 READ and FORMAT statements for data in non-WBAN format are listed below.

```
      READ (NSFC,203) CALL,Y(N),X(N),Z(S,N),DR,SP,Z(3,N),Z(4,N),PSTA
203   FORMAT (A4,2F7.2,42X,F8.1,4F7.1,F8.1)
```

For data in WBAN format, these statements read as follows:

```
      READ (NSFC,202) WBAN,Y(N),X(N),Z(S,N),DR,SP,Z(3,N),Z(4,N),PSTA
202   FORMAT (I5.5,2F7.2,42X,F8.1,4F7.1,F8.1)
```

The Stage 2 parameters are described in Table 2.129.

B. Output Files

1. Standard Output Files

P16G writes PF111, which contains hourly-interpolated upper-air profiles at 25-mb intervals. PF111 is used by P01G, P05G, and P08G. Table 2.130 shows the parameters of PF111.

TABLE 2.129. SURMET1 STAGE 2 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1a	$CALL_n$		Char*4	Station ID (non-WBAN format)
1b	$WBAN_n$		Integer*4	Station ID (WBAN format)
2	Y_n	° N	Real*4	Latitude
3	X_n	° W	Real*4	Longitude
4	ZS_n	mb	Real*4	Sea level pressure
5	DR_n	deg	Real*4	Wind direction
6	SP_n	$m{\cdot}s^{-1}$	Real*4	Wind speed
7	$Z3_n$	°C	Real*4	Temperature
8	$Z4_n$	°C	Real*4	Dew point
9	$PSTA_n$	mb	Real*4	Station pressure

Note: n = surface station number (one or more digits).

TABLE 2.130. PF111 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$\hat{u}_{m,i}$	$m{\cdot}s^{-1}$	Real*4	West-to-east wind component
2	$\hat{v}_{m,i}$	$m{\cdot}s^{-1}$	Real*4	South-to-north wind component
3	$T_{m,i}$	°C	Real*4	Dry bulb temperature
4	$TD_{m,i}$	°C	Real*4	Dew point temperature
5	$P_{m,i}$	mb	Real*4	Atmospheric pressure level

Note: m = upper-air station number (one or more digits); i = level number at 25 mb intervals (one or more digits).

2. Nonstandard Output Files

None

III. CONTROL CARDS

Two control cards are used to input control data in the format shown below. Table 2.131 defines the control card variables.

JDT1 JDT2
INTFLG

Example:

8819612 8819911
Y

TABLE 2.131. CONTROL CARD VARIABLES

Variable Name	Unit	Description
JDT1	EST	Julian scenario start date and time as *YYDDDHH*
JDT2	EST	Julian scenario end date and time as *YYDDDHH*
INTFLG		Flag for surface stations in WBAN format (Y/N)

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	14	files	147	blocks
FORTRAN INCLUDE files:	9	files	30	blocks
Object files:	14	files	72	blocks
Executable file:	1	file	87	blocks
	38	files	336	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:02:13
Buffered I/O count:	263
Direct I/O count:	2620
Peak working-set size:	887
Peak virtual size:	4400

C. Space Requirements: Log and Print Files

P16GX1.LOG:	30 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.132 shows the input file and output file space requirements.

TABLE 2.132. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	RAWIN1	Nonstd	21585	1 month
	RAWSTA1	Nonstd	24	N/A
	SURMET1	Nonstd	40251	1 month
			61860	
Output	PF011	1	1899	72 hours

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P16GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P16GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P16GV1A.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P16GV1A.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P16G.OBJ          A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]APPROX.OBJ        A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]COUPLE.OBJ        A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]ENDPTS.OBJ        A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]FILLER.OBJ        A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]GM2MOD.OBJ        A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]MBAR25.OBJ        A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]RADTIM.OBJ        A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]RWND25.OBJ        A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SFCHRS.OBJ        A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SFCLEV.OBJ        A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]SMUTHR.OBJ        A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]WHICH1.OBJ        A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]WNDFIX.OBJ        A14
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ         A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ        A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ        A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GETYN.OBJ         A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ         A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ          A20
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ        A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE2.OBJ        A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JHOUR1.OBJ        A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ         A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ        A25
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ         A26
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]POINTS.OBJ        A27
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ        A28
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ        A29
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ         A30
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT1.OBJ         A31
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ         A32
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P16GV1A.EXE -
  /MAP = MET1:[S01S]P16GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A30, -
  A31, -
  A32
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P16GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P16GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P16GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                          PFMFDIR
$ ASSIGN MET25:[FOREMAN.RAWDATA]RWND_JUL85.DAT             RAWIN1
$ ASSIGN MET1:[ROMNET.RAWDATA]RAWSTA.DAT                   RAWSTA1
$ ASSIGN MET23:[FOREMAN.RAWDATA]SRFC_JUL85.DAT             SURMET1
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF111A.DAT                  PF111
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P16GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P16GX1.CPR    SYS$INPUT
$ !
$ RUN $2$DUA24:[ROMEXE.ROMNET]P16GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A                P16GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P16GV1A

B. Subroutines

APPROX	FILLER	RADTIM	SFCLEV
COUPLE	GM2MOD	RWND25	SMUTHR
ENDPTS	MBAR25	SFCHRS	WNDFIX

C. Functions

WHICH1

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	IBLKR	OPEN2	RDDIR1
FLSMRY	JFILE1	POINTS	WRFT1
FSTAT1	OPDIR2	PROGID	WRREC

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETYN
IOCL
JFILE2
JHOUR1
JULIAN

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

ENDRAD.EXT	GMTTIM.EXT	MODTIM.EXT
ENDSFC.EXT	HOURLY.EXT	OUTPUT.EXT
FLOGIC.EXT	INTPRM.EXT	RAWSND.EXT

2.17 PROCESSOR P17G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 5 processor computes the heights of layers 0, 1, 2, and 3, and determines the hourly gridded local time derivatives ($\partial / \partial t$) of the heights of each layer. Figure 2.1 shows an example of the model layer heights and thicknesses along a cross section of row 41 of a hypothetical domain.

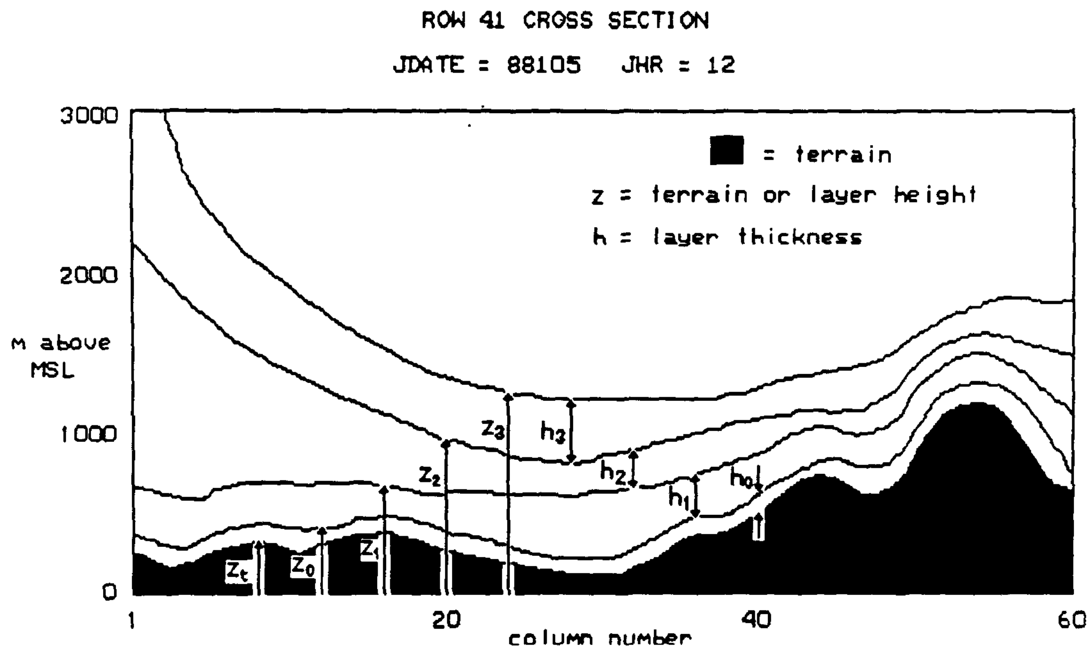


Figure 2.1. Sample model layer heights and thicknesses along row 41.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

P17G uses 5 standard input files:

PF119 from P06G
PF132 from P07G
MF109 from P07G
MF112 from P08G
MF113 from P08G

2. Nonstandard Input Files

None

B. Output Files

1. Standard Output Files

a. PF150

This file contains the hourly gridded local time derivative of the height of layer 0. It is used by P12G. The PF150 parameter is shown in Table 2.133.

TABLE 2.133. PF150 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$.Z0_{i,j}$	$m{\cdot}s^{-1}$	Real*4	Local time derivative of height Z_0

Note: i = column number, j = row number.

b. PF151

This file contains the hourly gridded local time derivative of the height of layer 1. It is used by P12G. The PF151 parameter is shown in Table 2.134.

TABLE 2.134. PF151 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$.Z1_{i,j}$	$m \cdot s^{-1}$	Real*4	Local time derivative of height Z_1

Note: i = column number, j = row number.

c. PF152

This file contains the hourly gridded local time derivative of the height of layer 2. It is used by P12G. The PF152 parameter is shown in Table 2.135.

TABLE 2.135. PF152 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$.Z2_{i,j}$	$m \cdot s^{-1}$	Real*4	Local time derivative of height Z_2

Note: i = column number, j = row number.

d. PF153

This file contains the hourly gridded local time derivative of the height of layer 3. It is used by P12G. The PF153 parameter is shown in Table 2.136.

TABLE 2.136. PF153 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$.Z3_{i,j}$	$m \cdot s^{-1}$	Real*4	Local time derivative of elevation Z_3

Note: i = column number, j = row number.

e. MF164

This file contains the hourly gridded height of the top of layer 0. It is used by P12G. The MF164 parameter is shown in Table 2.137.

TABLE 2.137. MF164 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$Z0_{i,j}$	m above MSL	Real*4	Height of the top of layer 0

Note: i = column number, j = row number.

f. MF165

This file contains the hourly gridded height of the top of layer 1. It is used by P12G. The MF165 parameter is shown in Table 2.138.

TABLE 2.138. MF165 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$Z1_{i,j}$	m above MSL	Real*4	Height of the top of layer 1

Note: i = column number, j = row number.

g. MF166

This file contains the hourly gridded height of the top of layer 2. It is used by P12G. The MF166 parameter is shown in Table 2.139.

TABLE 2.139. MF166 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$Z2_{i,j}$	m above MSL	Real*4	Height of the top of layer 2

Note: i = column number, j = row number.

h. MF167

This file contains the hourly gridded height of the top of layer 3. It is used by P12G. The MF167 parameter is shown in Table 2.140.

TABLE 2.140. MF167 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$Z3_{i,j}$	m above MSL	Real*4	Height of the top of layer 3

Note: i = column number, j = row number.

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 2.141 defines the control card variable.

YYDD1 H1 C1

TABLE 2.141. CONTROL CARD VARIABLES

Variable Name	Unit	Description
YYDD1		Julian start date as *YYDDD*
H1	EST	Start hour as *HH*
C1	h	Scenario duration

Example:

88194 12 72

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	15	blocks
FORTRAN INCLUDE files:	1	file	1	block
Object files:	2	files	9	blocks
Executable file:	1	file	52	blocks
	6	files	77	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:00:17
Buffered I/O count:	232
Direct I/O count:	5740
Peak working-set size:	1800
Peak virtual size:	4034

C. Space Requirements: Log and Print Files

P17GX1.LOG:	91 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.142 shows the input file and output file space requirements.

TABLE 2.142. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF119	4	40	72 hours
	PF132	5	1440	72 hours
	MF109	5	1440	72 hours
	MF112	5	1440	72 hours
	MF113	5	1440	72 hours
			5800	
Output	PF150	5	1440	72 hours
	PF151	5	1440	72 hours
	PF152	5	1440	72 hours
	PF153	5	1440	72 hours
	MF164	5	1440	72 hours
	MF165	5	1440	72 hours
	MF166	5	1440	72 hours
	MF167	5	1440	72 hours
			11520	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P17GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P17GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P17GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMCOM.ROMNET]P17GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P17G.OBJ              A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDPRM2.OBJ            A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ             A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR3.OBJ            A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ             A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ            A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ            A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ             A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ              A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ            A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ            A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ             A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ            A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ             A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ            A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ            A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD1.OBJ            A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ             A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT4.OBJ             A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ             A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT5.OBJ             A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ             A22
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMCOM.ROMNET]P17GV1A.EXE -
  /MAP = MET1:[S01S]P17GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A20, -
  A21, -
  A22
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P17GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P17GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P17GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                   PFMFDIR
$ ASSIGN MET1:[ROMNET1.D00000]PF119A.DAT            PF119
$ ASSIGN MET56:[ROMNET1.D85188]PF132A.DAT           PF132
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT           MF109
$ ASSIGN MET56:[ROMNET1.D85188]MF112A.DAT           MF112
$ ASSIGN MET56:[ROMNET1.D85188]MF113A.DAT           MF113
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF150A.DAT           PF150
$ ASSIGN MET56:[ROMNET1.D85188]PF151A.DAT           PF151
$ ASSIGN MET56:[ROMNET1.D85188]PF152A.DAT           PF152
$ ASSIGN MET56:[ROMNET1.D85188]PF153A.DAT           PF153
$ ASSIGN MET56:[ROMNET1.D85188]MF164A.DAT           MF164
```

```
$ ASSIGN MET56:[ROMNET1.D85188]MF165A.DAT         MF165
$ ASSIGN MET56:[ROMNET1.D85188]MF166A.DAT         MF166
$ ASSIGN MET56:[ROMNET1.D85188]MF167A.DAT         MF167
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P17GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P17GX1.CPR   SYS$INPUT
$ !
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P17GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A        P17GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P17GV1A

B. Subroutines

RDPRM2

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	IBLKR	OPEN2	RDREC
CELL0	IDATE	PROGID	TIME
FSTAT1	JFILE1	RDDIR1	WRFT5
FLSMRY	OPDIR2	RDFT4	WRREC

B. Functions

None

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

2.18 PROCESSOR P19G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 1 processor computes the hourly gridded fractional sky coverage by all cloud types combined. Interpolation of valid surface observations of sky cover (between 0% and 100%) is achieved by the Barnes gridding subroutine BGRID1. Each grid cell value is then checked to ensure that all values lie between 0 (negative sky coverage) and 1 (sky coverage >100%); values are reset if necessary.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

P19G uses one nonstandard input file, SURMET1, that contains hourly surface meteorology observations from the National Climatic Data Center and the Canadian Climate Centre raw data tapes. SURMET1 is read in two stages. The first stage reads header information. READ and FORMAT statements for SURMET1 are called by P19G from RDSRF1 in the UTILIO library. Table 2.143 shows the SURMET1 Stage 1 parameters.

TABLE 2.143. SURMET1 STAGE 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	INYR		Integer*4	Scenario year (*YY*)
2	INMON		Integer*4	Scenario month (*MM*)
3	INDAY		Integer*4	Scenario day (*DD*)
4	INHOUR	EST	Integer*4	Scenario hour (*HH*)
5	NSTAT		Integer*4	No. of reporting stations at *HH*

The Stage 2 parameters that are used in P19G are described in Table 2.144:

TABLE 2.144. SURMET1 STAGE 2 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1a	$STAID_n$		Char*4	Station ID (non-WBAN format)
1b	$STAID_n$		Integer*4	Station ID (WBAN format)
2	$SLATS_n$	° N	Real*4	Latitude
3	$SLONS_n$	° W	Real*4	Longitude
4	$OPQCOV_n$	%	Real*4	Opaque sky cover
5	$TOTCOV_n$		Real*4	Total sky cover (see note below)
6	$COVER1_n$	%	Real*4	Coverage, lowest cloud layer
7	$BASE1_n$	m	Real*4	Lowest cloud base
8	$COVER2_n$	%	Real*4	Coverage, second-lowest cloud layer
9	$BASE2_n$	m	Real*4	Second-lowest cloud base
10	$COVER3_n$	%	Real*4	Coverage, third-lowest cloud layer
11	$BASE3_n$	m	Real*4	Third-lowest cloud base
12	$P0_n$	mb	Real*4	Sea level pressure
13	DIR_n	deg	Real*4	Wind direction
14	SPD_n	$m{\cdot}s^{-1}$	Real*4	Wind speed
15	T_n	°C	Real*4	Temperature
16.	TD_n	°C	Real*4	Dew point
17	P_n	mb	Real*4	Station pressure

Note: n = surface station number (one or more digits). Total sky cover classes are as follows: 0 = clear; 1 = partial obscuration; 2 = thin scattered; 3 = thin broken; 4 = thin overcast; 5 = scattered; 6 = broken; 7 = overcast; 8 = obscured; -99 = indeterminate.

B. Output Files

1. Standard Output Files

P19G produces one standard file, PF117, that contains the hourly gridded fraction of sky coverage by all cloud types combined. It is used by P17G, and Table 2.145 shows its parameter.

TABLE 2.145. PF117 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SIGMAct		Real*4	Fractional sky coverage by all cloud types

Note: ct = cloud total

2. Nonstandard Output Files

None

III. CONTROL CARDS

Two control cards are used to input control data in the format shown below. Table 2.146 defines the control card variables.

ISTDAT ENDAT
INTFLAG

TABLE 2.146. CONTROL CARD VARIABLES

Variable Name	Unit	Description
ISTDAT	EST	Julian scenario start date and time as *YYDDDHH*
ENDAT	EST	Julian scenario end date and time as *YYDDDHH*
INTFLAG		Use of WBAN-format station identifiers in SURMET1: Y ⇒ WBAN format used; N ⇒ WBAN format not used

Example:

8819612 8819911
Y

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	files	9	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	files	5	blocks
Executable file:	1	file	63	blocks
	3	files	77	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:08:06
Buffered I/O count:	287
Direct I/O count:	1692
Peak working-set size:	735
Peak virtual size:	4400

C. Space Requirements: Log and Print Files

P19GX1.LOG:	27 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.147 shows the input file and output file space requirements.

TABLE 2.147. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SURMET1	Nonstd	107616	1 month
Output	PF117	5	1438	72 hours

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P19GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P19GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P19GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P19GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P19G.OBJ          A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]BGRID1.OBJ       A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ        A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ       A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ       A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GETYN.OBJ        A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GMTLCL.OBJ       A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ        A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ         A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOERR.OBJ        A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ       A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ       A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ       A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JULIAN.OBJ       A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ        A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ       A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ        A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ       A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ       A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ        A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDSRF1.OBJ       A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDWNDO.OBJ       A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT5.OBJ        A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ        A25
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P19GV1A.EXE -
  /MAP = MET1:[S01S]P19GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A23, -
```

```
  A24, -
  A25
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P19GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P19GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P19GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                   PFMFDIR
$ ASSIGN MET23:[FOREMAN.RAWDATA]SRFC_JUL85.DAT      SURMET1
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF117A.DAT           PF117
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN USER2$DISK:[FOREMAN.FV1501A]P19GX1.CPR
$ !
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P19GX1.CPR    SYS$INPUT
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P19GV1A.EXE
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P19GV1A

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	GMTLCL	OPEN2	RDSRF1
BGRID1	IBLKR	PROGID	RDWNDO
CELL0	IOERR	RDDIR1	WRFT5
FLSMRY	JFILE1	RDREC	WRREC
FSTAT1	OPDIR2		

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETYN	JHOUR1
IOCL	JULIAN
JFILE2	JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

2.19 PROCESSOR P21G (CHEMISTRY)

I. PROCESSOR FUNCTION

This stage 0 processor uses the initial mean tropospheric background concentrations listed in Table 2.148 (Killus and Whitten, 1984) and computes, using the temperature-dependent rate constants calculated by preprocessor CVCHEM, all the carbon-bond mechanism 4.2 chemical reactions (Gery *et al.*, 1988) for "clean air." The processor equilibriates gases using constant daylight photolysis and reaction rates for 720 minutes (12 hours); calculated background concentrations are output after 600 minutes. These calculations are followed by equilibriation at constant nighttime reaction rates for the remaining 720 minutes of a 24-hour day; nighttime background concentrations are output after 1320 minutes.

TABLE 2.148. INITIAL MEAN TROPOSPHERIC BACKGROUND CONCENTRATIONS, PRECURSOR SPECIES

Species	Concentration (ppm)	Species	Concentration (ppm)
CO	0.1	Aldehydes	1.12×10^{-3}
NO_2	1.0×10^{-3}	Formaldehyde	1.4×10^{-3}
NO	1.0×10^{-3}	Toluene	1.4×10^{-4}
Ethanol	3.5×10^{-4}	Xylene	1.05×10^{-4}
Olefins	2.1×10^{-4}	Paraffins	7.42×10^{-3}
		All others	1.0×10^{-16}

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

P21G uses one nonstandard input file, RATECONS, that contains the rate constants output by CVCHEM. The read and format statements for this file are listed below. Note that the file contains two header records followed by the data records; Table 2.149 shows the file's parameters.

```
      READ (JRATE,1017) TEXT
 1017 FORMAT (A100)
      READ (JRATE,1017) TEXT
      DO 103 IDORN=1,2
         READ (JRATE,1023) NORD(IDORN)
 1023    FORMAT (A8)
         DO 103 ILAY=1,NLAY
            READ (JRATE,1023) INLAY(ILAY)
            READ (JRATE,1027) (RK(I,LAY,IDORN),IF1,NREACT)
 1027       FORMAT (5E15.0)
  103 CONTINUE
```

TABLE 2.149. RATECONS PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	TEXT		Char*100	Header record text
2	NORD		Char*8	NIGHT or DAY indicator
3	INLAY		Char*8	Layer number
4	$RK_{r,L,i}$	$ppm^{-1} \cdot s^{-1}$	Real*4	Rate constant for 2-species chemical reactions
4	$RK_{r,L,i}$	s^{-1}		Rate constant for monospecific photolytic reactions

Note: r = chemical reaction number; L = layer number; i = 1 (day) or 2 (night).

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

P21G generates CLEANAIR. This file contains the "clean air" (background) concentrations for all the chemical species (listed in Gery *et al.*, 1988) used in the ROM. The first record of this file contains the first and second parameters shown in Table 2.150; the next record is a P21G-generated header record; this is followed by the remaining data records, shown in the same Table. CLEANAIR is used in P02G, P23G, and P24G. The 35 chemical species contained in the file are shown in Table 3.155. The file is produced by the WRITE and FORMAT statements listed below.

```
      WRITE (OUTUNT,1001) CDATE,CTIME
 1001 FORMAT (1X,'FILE CREATED ON DATE: ',A8,'AT TIME: ',A8)
      WRITE (OUTUNT,1021)
 1021 ('THIS FILE OF CLEAN-AIR BACKGROUND CONCENTRATIONS IS
     &  VALID FOR SUMMERTIME SCENARIOS.')
      DO 205 IDORN=1,2
         WRITE (OUTUNT,1023) NORD(IDORN)
 1023    FORMAT (A8)
         DO 205 ILAY=1,NLAY
            WRITE (OUTUNT,1023) INLAY(ILAY)
            WRITE (OUTUBT,1011) (SPNAME(I),NITRO(I),C(I),I=1,NSPECS)
 1011       FORMAT (A4,2X,L1,3X,E10.4)
  205 CONTINUE
```

TABLE 2.150. CLEANAIR PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CDATE		Char*8	File creation date as *MMDDYY*
2	CTIME	EST	Char*8	File creation time as *HHMMSS*
3	NORD		Char*8	NIGHT or DAY indicator
4	INLAY		Char*8	Level number
5	SPNAME		Char*4	Chemical species name / abbreviation
6	NITRO		Logical*4	Flag: = .TRUE. for all nitrogenous species
7	$C_{L,i}$	ppm	Real*4	Species concentration

Note: L = level number; i = 1 (day) or 2 (night).

III. CONTROL CARDS

One control card is used, in the format shown below. Table 2.151 describes the control card variables.

DELTAT OUTIME TSP

TABLE 2.151. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Data Type	Description
1	DELTAT	min	Real*4	Total simulation time for each set of rate constants
2	OUTIME	min	Real*4	Simulation time at which output files are written
3	TSP	min	Real*4	Simulation time interval for printing concentrations to the log file for quality assurance checks

Example:

720.0 600.0 60.0

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	23	blocks
FORTRAN INCLUDE files:	7	files	21	blocks
Object files:	2	files	17	blocks
Executable file:	1	file	15	blocks
	12	files	76	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:01:35
Buffered I/O count:	149
Direct I/O count:	328
Peak working-set size:	571
Peak virtual size:	937

C. Space Requirements: Log and Print Files

P21GX1.LOG:	198 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.152 shows the input file and output file space requirements.

TABLE 2.152. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	RATECON	Nonstd	22	N/A
Output	CLEANAIR	Nonstd	11	set by control card

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P21GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P21GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P21GV1A.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P21GV1A.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P21G.OBJ              A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]BLKCLN.OBJ            A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ             A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]GPRIME.OBJ            A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ              A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE2.OBJ            A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ             A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PQ1CB42.OBJ           A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PQCOEFCB42.OBJ        A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ            A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RATEDCB42.OBJ         A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]TOTAL.OBJ             A12
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P21GV1A.EXE -
  /MAP = MET1:[S01S]P21GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A10, -
  A11, -
  A12
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P21GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P21GX1.COM for U.S. data follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1002A]P21GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ !
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[NEROS]DIRECT2.DAT                    PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]RATECB4-2.DAT         RATECONS
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET1:[ROMNET.RAWDATA]CLEANAIR_ROMNET.DAT   CLEANAIR
$ !
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARD FILE
$ TYPE USER2$DISK:[FOREMAN.FV1002A]P21GX1.CPR
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1002A]P21GX1.CPR   SYS$INPUT
$ !
$ RUN USER2$DISK:[ROMEXE.NEROS]P21G.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P21GV1A

B. Subroutines

None

C. Functions

None

D. Block Data Files

BLKCLN

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	PQCOEFCB42	RATEDCB42
GPRIME	PROGID	TOTAL
PQ1CB42		

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

COMBS.EXT	IOUNIT.EXT	PRID.EXT
CONCS.EXT	PARAM.EXT	TRCONC.EXT
FRACT.EXT		

2.20 PROCESSOR P22G (CHEMISTRY)

I. PROCESSOR FUNCTION

This stage 2 processor generates BCON, the boundary conditions file, which contains the species concentrations at each of the modeling domain's four boundaries (north, south, east, and west) for every time step.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

P22G uses one nonstandard input file output from P24G, EQUILBCS, which contains the steady state species concentrations (for 35 species) at each boundary. Note that in this version of the ROM, EQUILBCS.DAT contains data only for the two hours of each day of an episode at which P24G calculates boundary conditions. These hours are assumed to be representative of daytime and nighttime boundary conditions respectively. The READ and FORMAT statements for EQUILBCS.DAT are listed below, and its parameters are shown in Table 2.153.

```
      READ (JCONC,*) BCSTDT,BCSTHR
      DO 109 ILEV=1,NLEVBC
      DO 109 ISPC=1,NSPCBC
         READ (JCONC,1003) SPNMBC(ISPC),
     &        (CONCS(ILEV,IBND,ISPC),IBND=1,NBOUND)
1003     FORMAT (A4, <NBOUND>(2X,E10.4))
 109  CONTINUE
```

TABLE 2.153. EQUILBCS PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	BCSTDT		Integer*4	Boundary condition Julian date as *YYDDD*
2	BCSTHR	EST	Integer*4	Boundary condition time as *HH*
3	$SPNMBC_{L,b}$		Char*4	Species name (see Table 2.154)
4	$CONCS_{L,b}$	ppm	Real*4	Species concentration

Note: L = layer number; b = boundary.

The 35 species used in the ROM are shown in Table 2.154.

TABLE 2.154. ROM SPECIES AND REACTIONS

Symbol	Description	Symbol	Description
ALD2	High MW aldehydes	O1D	O^1D atom
C203	Peroxyacetyl radical	O3	Ozone
CO	Carbon monoxide	OH	Hydroxyl radical
CRES	Cresol and high MW phenols	OLE	Olefinic carbon bond
CRO	Methylphenoxy radical	OPEN	High MW aromatic oxidation ring fragment
ETH	Ethene		
FORM	Formaldehyde	PAN	Peroxyacetyl nitrate
H2O2	Hydrogen peroxide	PAR	Paraffinic carbon bond
HNO2	Nitrous acid	PNA	Peroxynitric acid
HNO3	Nitric acid	ROR	Secondary organic oxy radical
HO2	Hydroperoxy radical	TO2	Toluene-hydroxyl radical adduct
ISOP	Isoprene structures	TOL	Toluene
MGLY	Methylglyoxal	XO2	NO to NO_2 reaction
N2O5	Dinitrogen pentoxide	XO2N	NO to nitrate (NO_3^-) reaction
NO	Nitric oxide	XYL	Xylene
NO2	Nitrogen dioxide	MTHL	Methanol
NO3	Nitrogen trioxide	NONR	Nonreactive hydrocarbons
O	O^3P atom	TRAC	Tracer species

Note: MW = molecular weight

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

P22G generates BCON; this file contains the boundary condition concentrations of the species shown in Table 2.154 for each model level (or layer), for a one-cell depth at each of the boundaries. The BCON file contains the same data as the input file, but in the standard format of ROM concentration files. However, P22G writes a boundary concentration for every model time step; these data are copies of information contained in EQUILBCS, and change only when new data is encountered in the input file. (Recall that EQUILBCS contains only two times for each day at which boundary conditions are calculated.) BCON is used by the Core Model. There are up to 24 records written to the file prior to the concentration field. These are described below.

a. Record 1

This record contains the parameters shown in Table 2.155. The parameters are first written to the buffer SEG1BF; subsequently subroutine WRCHAR writes the data (unformatted) to BCON. These steps are listed below.

```
      WRITE (SEG1BF,1001,IOSTAT=IOST)
     &      CDATBC, CTIMBC, SDATBC, STHRBC, TSTPBC, FRSTBC,
     &      GRDNBC, SWLNBC, SWLTBC, NELNBC, NELTBC, DLONBC,
     &      DLATBC, NCOLBC, NROWBC, NLEVBC, NSPCBC, ICNTBC
 1001 FORMAT (6I8.8,A8,4F8.3,2F8.5,5I4.4)

      CALL WRCHAR (IUNIT,SEG1BF,IOST)

      SUBROUTINE WRCHAR(IUNIT,CHBUF,IOST)
      IMPLICIT NONE
      INTEGER*4 IUNIT,IOST
      CHARACTER*(*) CHBUF
      WRITE(IUNIT,IOSTAT=IOST) CHBUF
      RETURN
      END
```

b. Record 2

This record contains the parameter shown in Table 2.156. The parameter is first written to the buffer SPNMBF; subsequently subroutine WRCHAR writes the data (unformatted) to BCON. The WRITE and FORMAT statements are listed below:

```
      WRITE (SPNMBF,1003,IOSTAT=IOST) (SPNMBC(I), I=1,NSPCBC)
 1003 FORMAT (6(10(A4))/)
      CALL WRCHAR (IUNIT,SPNMBF,IOST)
```

TABLE 2.155. BCON RECORD 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CDATBC		Integer*4	File creation date as *MMDDYY*
2	CTIMBC	EST	Integer*4	File creation time as *HHMMSS*
3	SDATBC		Integer*4	Julian scenario start date as *YYDDD*
4	STHRBC	EST	Integer*4	Scenario start time as *HH*
5	TSTPBC	s	Integer*4	Time step, simulation
6	FRSTBC	s	Integer*4	First time step in BCON
7	GRDNBC		Char*8	Grid name (see Appendix A)
8	SWLNBC	° W	Real*4	Longitude, SW corner of grid
9	SWLTBC	° N	Real*4	Latitude, SW corner of grid
10	NELNBC	° W	Real*4	Longitude, NE corner of grid
11	NELTBC	° N	Real*4	Latitude, NE corner of grid
12	DLONBC	° W	Real*4	Cell longitude increment
13	DLATBC	° N	Real*4	Cell latitude increment
14	NCOLBC		Integer*4	Number of columns in grid
15	NROWBC		Integer*4	Number of rows in grid
16	NLEVBC		Integer*4	Number of levels in simulation
17	NSPCBC		Integer*4	Number of species in BCON file
18	ICNTBC		Integer*4	Number of text records

TABLE 2.156. BCON RECORD 2 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SPNMBC		Char*4	Chemical species name

c. Record 3

This record contains the parameter shown in Table 2.157. The parameter is first written to the buffer LEVNBF; subsequently subroutine WRCHAR writes the data (unformatted) to BCON. The WRITE and FORMAT statements are listed below:

```
      WRITE (LEVNBF,1005,IOSTAT=IOST)
     &      (LVNMBC(I), I=1,NLEVBC)
 1005 FORMAT (2(10(A4))/)
      CALL WRCHAR (IUNIT,LEVNBF,IOST)
```

TABLE 2.157. BCON RECORD 3 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	LVNMBC		Char*4	Layer number

d. Records 4 - (4 + ICNTBC)

These records contain the parameter shown in Table 2.158, which consists of the optional text that can be entered as part of P22G's input. The parameter is first written to the buffer TEXTBF from control card records 3 up to 22; subsequently, subroutine WRCHAR writes the 80-character strings to the output records in BCON. The WRITE statements are listed below:

```
      DO 200, I = 1, ICNTBC
      TEXTBF = TEXTBC(I)
      CALL WRCHAR (IUNIT,TEXTBF,IOST)
200   CONTINUE
```

TABLE 2.158. BCON RECORDS 4 - (4 + ICNTBC) PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	TEXTBC		Char*80	Descriptive text

e. Record (5 + ICNTBC): the time step header

There is one time step header record for each time step. This record contains the parameters shown in Table 2.159. The WRITE statements write the header from a COMMON declaration; these codes are listed below:

```
      REAL*4          DATBC, TIMBC, ELPBC, STPBC
      COMMON /RTSHBC/ DATBC, TIMBC, ELPBC, STPBC

C--WRITE TIME STEP HEADER
      CALL WRFILE (IUNIT,IWLTSH,DATBC,IOST)

      SUBROUTINE WRFILE (IUNIT, NWORDS, BUFFER, IOST)
      IMPLICIT NONE
      INTEGER*4    IUNIT, NWORDS, IOST
      REAL*4       BUFFER (NWORDS)
      WRITE (IUNIT, IOSTAT = IOST) BUFFER
      RETURN
      END
```

TABLE 2.159. BCON TIME STEP HEADER RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATBC		Real*4	Julian date on BCON file as *YYDDD*
2	TIMBC	EST	Real*4	Time on BCON file as *HHMMSS*
3	ELPBC	s	Real*4	Elapsed time since start
4	STPBC		Real*4	Step number on BCON file

f. Records (6 + ICNTBC) - (4 x NSPECIES): species concentration data

These records are written for each time step, and contain the boundary species concentrations by layer (determined by the subroutine WRBC), iterated over either columns (northern and southern boundaries), or rows (western and eastern boundaries). The records' parameters are shown in Table 2.160. The value of the IWDLN2 variable shown in the WRITE statements below is the product of the number of columns in the modeling domain and the number of model layers; the value of IWDLN3 is the product of the number of rows in the modeling domain and the number of model layers. IWDLN2 and IWDLN3 define the buffer length.

```
      DO 301, ISPC = 1, NSPECS
      CALL WRFILE (IUNIT,IWDLN3,WEST(1,1,ISPC),IOST)
301   CONTINUE
      DO 303, ISPC = 1, NSPECS
      CALL WRFILE (IUNIT,IWDLN3,EAST(1,1,ISPC),IOST)
303   CONTINUE
      DO 305, ISPC = 1, NSPECS
      CALL WRFILE (IUNIT,IWDLN2,NORTH(1,1,ISPC),IOST)
305   CONTINUE
      DO 307, ISPC = 1, NSPECS
      CALL WRFILE (IUNIT,IWDLN2,SOUTH(1,1,ISPC),IOST)
307   CONTINUE
```

TABLE 2.160. BCON SPECIES CONCENTRATION RECORDS PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Species concentrations at the:
1	$WEST_{j,L,s}$	ppm	Real*4	Western boundary
2	$EAST_{j,L,s}$	ppm	Real*4	Eastern boundary
3	$NORTH_{i,L,s}$	ppm	Real*4	Northern boundary
4	$SOUTH_{i,L,s}$	ppm	Real*4	Southern boundary

Note: i = column number; j = row number; L = layer number; s = species.

III. CONTROL CARDS

Two control cards are used, in the format shown below. Table 2.161 describes the control card variables. Following these two control cards, optional text information can be entered in as many as 20 records. This text will be included in the BCON file's header.

REGION SDATBC STHRBC
IDATBC ITIMBC NSTEPS
VAXFMT

TABLE 2.161. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Description
1	REGION		Modeling domain name
2	SDATBC		Julian scenario start date as *YYDDD*
3	STHRBC	EST	Scenario start hour as *HH*
4	IDATBC		Julian date of boundary conditions as *YYDDD*
5	ITIMBC	EST	Time of initial conditions as *HHMMSS*
6	NSTEPS		Number of time steps
7	VAXFMT		Translate flag: .TRUE. ⇒ do not translate to IBM binary .FALSE. ⇒ translate to IBM binary

Example:

'ROMNET' 88236, 12
88236, 120000, 145
BOUNDARY CONDITIONS FILE GENERATED FOR 145 TIME STEPS.
THIS BC FILE IS PRODUCED BY DISK$VAXSET:[ROMEXE.ROMNET]P22GV1C.EXE.
THERE ARE THIRTY-FIVE SPECIES USED IN THE CURRENT VERSION OF THE ROM.
.FALSE.

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	4	files	53	blocks
FORTRAN INCLUDE files:	11	files	39	blocks
Object files:	4	files	25	blocks
Executable file:	1	file	45	blocks
	20	files	162	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:00:52
Buffered I/O count:	574
Direct I/O count:	2998
Peak working-set size:	570
Peak virtual size:	2518

C. Space Requirements: Log and Print Files

P22GX1.LOG:	54 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.162 shows the input file and output file space requirements.

TABLE 2.162. I/O FILE SPACE REQUIREMENTS

File Group	Logical File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	EQUILBCS	Nonstd	330	20 days
Output	BCON	Nonstd	30677	72 hours

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P22GV1C.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P22GV1C.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P22GV1C.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMCOM.ROMNET]P22GV1C.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P22G.OBJ              A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]BLKBC.OBJ             A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]OPWRBC.OBJ            A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRBC.OBJ              A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ             A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ             A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CLOCK1.OBJ            A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CLOCK2.OBJ            A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATTIM.OBJ            A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ              A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ            A11
```

```
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE5.OBJ          A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ           A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ          A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRCHAR.OBJ          A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFILE.OBJ          A16
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMCOM.ROMNET]P22GV1C.EXE -
  /MAP = MET1:[S01S]P22GV1C.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A14, -
  A15, -
  A16
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P22GV1C.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P22GX1.COM follows; note that file-naming conventions (for the BCON file) is described in Part 3 of this User's Guide (EPA 600/8-90-083c).

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[FOREMAN.FV1502A]P22GX1.COM
$ !
$ ASSIGN MLAB: SYS$PRINT
$ !
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                    PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]EQBC_RNETJUL85.DAT     EQUILBCS
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET31:[FOREMAN.ICBCS]BCFV1502A.DAT          BCON
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE DISK$VAXSET:[FOREMAN.FV1502A]P22GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ !$ ASSIGN / USER_MODE DISK$VAXSET:[FOREMAN.FV1502A]P22GX1.CPR   SYS$INPUT
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P22GV1C.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P22GV1C

B. Subroutines

OPWRBC
WRBC

C. Functions

None

D. Block Data Files

BLKBC

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	CLOCK2	WRCHAR
CELL0	DATTIM	WRFILE
CLOCK1	PROGID	

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JFILE5
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

BCFILE.EXT	HEADBC.EXT	PRID.EXT
FLNAMS.EXT	LUNITS.EXT	REGION.EXT
GRDDEF.EXT	LVNAME.EXT	TSHDBC.EXT
HDFMTS.EXT	PARM22.EXT	

2.21 PROCESSOR P23G (CHEMISTRY)

I. PROCESSOR FUNCTION

This stage 1 processor reads the 35 species background concentrations from CLEANAIR, and outputs the upper boundary (the top of layer 3) concentrations for a subset of 12 "species." These "species" are, in fact, aggregations of species by magnitude of their concentrations, as shown by example in Table 2.163. P23G assigns a discrete value, calculated as the mean group concentration, to each group. It chooses a representative species from each group to denote the name of the group in the MF147 output file. The purpose of these aggregations is (1) to reduce CPU time in later stages of the ROM, and

(2) to conform to the structure of PF/MF files, which limit the number of species that can be written to them. The ROM can reconstruct the individual species concentrations as required via an indexing array that holds the relevant mapping information.

TABLE 2.163. EXAMPLE OF SPECIES AGGREGATIONS IN P23G

Magnitude	Species	Mean group concentration (ppm)
10^{-1}	carbon monoxide	0.988
10^{-2}	ozone	2.91
10^{-3}	formaldehyde hydrogen peroxide nitric acid paraffinic C-bond	1.81
10^{-4}	aldehydes nitric oxide nitrogen dioxide peroxyacetyl nitrate	1.75
10^{-5}	ethene hydroperoxy radical toluene $NO \rightarrow NO_2$ reaction $NO \rightarrow NO_3^-$ reaction	1.18
10^{-6}	peroxyacetyl radical heavy phenols nitrous acid peroxynitric acid	1.57
10^{-7}	hydroxyl radical heavy ring fragments	3.98
10^{-8}	methylphenoxy radical methylglyoxal nitrogen trioxide	0.708
10^{-9}	dinitrogen pentoxide olefins xylene	1.60
10^{-10}	O^3P atom O^1D atom toluene-hydroxyl radical adduct	1.57
10^{-11}	2° organic oxy radical	0.836
10^{-12}	none	
10^{-13}-10^{-14}	none	
10^{-15}	none	
10^{-16}	isoprenes methanol nonreactive HC's tracer species	1.00

We equilibriate the layer 3 ozone concentrations with ozone levels in the free atmosphere by initializing the layer 3 background concentration with free atmosphere concentrations. The chemical reactions are then modeled until the system reaches a steady state.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. CLEANAIR

CLEANAIR, created by P21G, contains the "clean air" (background) steady state daytime and nighttime concentrations for the 35 chemical species used in the ROM. It is used to generate the steady state boundary conditions. The file is read by the WRITE and FORMAT statements listed below, and its parameters are shown in Table 2.164.

```
        READ (JCONC,1011) HDTEXT
        READ (JCONC,1011) HDTEXT
        DO 101 IDORN=1,2
           READ (JCONC,1015) NORD
           DO 101 ILAY=1,NLAY
              READ (JCONC,1015) INLAY
              READ (JCONC,1019) (SPNAME(ISPC),NITRO(ISPC),LAYERC(I,ILAY,IDORN),
     &                   ISPC=1,NSPECS)
 101 CONTINUE
1011 FORMAT (A100)
1015 FORMAT (A8)
1019 FORMAT (A4,2X,L1,3X,E10.4)
```

TABLE 2.164. CLEANAIR PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	HDTEXT		Char*100	Header text record
2	NORD		Char*8	NIGHT or DAY indicator
3	INLAY		Char*8	Layer number
4	$SPNAME_{L,i}$		Char*4	Chemical species name / abbreviation
5	$NITRO_{L,i}$		Logical*4	Flag: = .TRUE. for all nitrogenous species
6	$LAYERC_{L,i}$	ppm	Real*4	Species concentration

Note: L = layer number, i = 1 (day) or 2 (night)

b. RATECONS

RATECONS, created by preprocessor CVCHEM, contains the rate constants for the reactions of the Carbon-bond 4.2 mechanism. The rates in this file have been adjusted to the meteorological parameters in CVCHEM (see Section 7.2.1, Part 1). The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 2.165.

```
      READ (JRATE,1011) HDTEXT
      READ (JRATE,1011) HDTEXT
1011  FORMAT (A100)
      DO 103 IDORN=1,2
        READ (JRATE,1015) NORD
          DO 103 ILAY=1,INLAY
          READ (JRATE,1015) INLAY
          READ (JRATE,1027) (RK(I,ILAY,IDORN),I=1,NREACT)
103   CONTINUE
1015  FORMAT (A8)
1027  FORMAT (5E15.0)
```

TABLE 2.165. RATECONS PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	HDTEXT		Char*100	Header text record
2	NORD		Char*8	NIGHT or DAY indicator
3	INLAY		Char*8	Layer name
4	$RK(1)_{i,L}$	min^{-1}	Real*4	Rate constant, species 1
.	.		.	.
.	.	or	.	.
.	.		.	.
86	$RK(83)_{i,L}$	$ppm^{-1} \cdot min^{-1}$	Real*4	Rate constant, species 83

Note: i = day (1) or night (2) indicator; L = layer. Units are dependent on the number of reacting gases.

B. Output Files

1. Standard Output Files

P23G creates MF147, which contains the hourly gridded upper boundary condition species concentrations ($c_{\cdot}$) for 12 species. MF147 is used by P39G, and its parameters are shown in Table 2.166.

2. Nonstandard Output Files

None

TABLE 2.166. MF147 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ALD2	ppm	Real*4	c_∞, aldehydes
2	C2O3	ppm	Real*4	c_∞, acetate
3	CO	ppm	Real*4	c_∞, carbon monoxide
4	CRO	ppm	Real*4	c_∞, methylphenoxy radical
5	ETH	ppm	Real*4	c_∞, ethene
6	FORM	ppm	Real*4	c_∞, formaldehyde
7	ISOP	ppm	Real*4	c_∞, isoprenes
8	N2O5	ppm	Real*4	c_∞, nitrogen pentoxide
9	O	ppm	Real*4	c_∞, atomic oxygen
10	O3	ppm	Real*4	c_∞, ozone
11	OH	ppm	Real*4	c_∞, hydroxyl radical
12	ROR	ppm	Real*4	c_∞, secondary organic oxy radicals

Note: c_∞ = background concentration at top of layer 3

III. CONTROL CARDS

One control card is used, in the format shown below. Table 2.167 describes the control card variables.

SDATIN STHRIN NHOURS

TABLE 2.167. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Description
1	SDATIN		Julian scenario start date as *YYDDD*
2	STHRIN	EST	Scenario start hour as *HH*
3	NHOURS	h	Number of hours to process

Example:

88105 120000 72

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	24	blocks
FORTRAN INCLUDE files:	13	files	78	blocks
Object files:	1	file	15	blocks
Executable file:	1	file	75	blocks
	16	files	192	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:00:37
Buffered I/O count:	253
Direct I/O count:	2778
Peak working-set size:	88
Peak virtual size:	2992

C. Space Requirements: Log and Print Files

P23GX1.LOG:	60 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.168 shows the input file and output file space requirements.

TABLE 2.168. I/O FILE SPACE REQUIREMENTS

File Group	Logical File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	CLEANAIR	Nonstd	11	N/A
	RATECONS	Nonstd	22	N/A
			33	
Output	MF147	5	22776	72 hours

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P23GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P23GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P23GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P23GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P23G.OBJ              A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ            A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ            A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ           A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ           A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GPRIME.OBJ           A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ            A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]INDEX1.OBJ           A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ             A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ           A10
```

```
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ              A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ              A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ               A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PQ1CB42.OBJ             A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PQCOEFCB42.OBJ          A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ              A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ              A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ               A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RATEDCB42.OBJ           A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ              A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ               A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]TOTAL.OBJ               A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ               A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT5.OBJ               A24
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P23GV1A.EXE -
  /MAP = MET1:[S01S]P23GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A22, -
  A23, -
  A24
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P23GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P23GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P23GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                  PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]CLEANBCS.DAT         CLEANAIR
$ ASSIGN MET1:[ROMNET.RAWDATA]RATECB4-2.DAT        RATECONS
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]MF147A.DAT          MF147
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARD(S) FILE
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P23GX1.CPR
$ !
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P23GX1.CPR    SYS$INPUT
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P23GV1A.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P23GV1A

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	IBLKR	OPDIR2	RDREC
CELL0	JFILE1	OPEN2	TOTAL
FLSMRY	PQ1CB42	RATEDCB42	WRREC
FSTAT1	PQCOEFCB42	RDDIR1	WRFT5
GPRIME	PROGID		

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

INDEX1
IOCL
JFILE2
JHOUR1
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

COMBS.EXT	FDIR2.EXT	FRACT.EXT	PARAM.EXT
CONCS.EXT	FHEAD1.EXT	GRDDEF.EXT	PARMS.EXT
DIRREC.EXT	FHEAD2.EXT	IOUNIT.EXT	PRID.EXT
FDIR1.EXT			

2.22 PROCESSOR P24G (CHEMISTRY)

I. PROCESSOR FUNCTION

This stage 1 processor generates the twice daily (daytime and nighttime) gridded equilibriated concentrations for the 35 species used in the ROM, for the north, south, east, and west boundaries of each model layer. Each boundary is assigned a single value for ozone, which all grid cells in that boundary assume; so that no systematic bias is introduced, this value roughly agrees with the observed ozone concentrations along the boundary. The remaining 34 species are then equilibriated to this ozone concentration, generating the set of concentration values for that boundary. The purpose of the equilibriation calculations performed by P24G is to reduce CPU time in later stages of the ROM.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. CLEANAIR

CLEANAIR, created by P21G, contains the "clean air" (background) steady state daytime and nighttime concentrations for the 35 chemical species used in the ROM. It is used to generate the steady state boundary conditions. The file is read by the following READ and FORMAT statements, and its parameters are shown in Table 2.169.

```
      READ (JICS,1009) HDTEXT
      READ (JICS,1009) HDTEXT
      DO 101 IDORN=1,2
         READ (JICS,1013) NORD
         DO 101 ILAY=1,NLAY
            READ (JICS,1013) INLAY
            READ (JICS,1017) (SPNAME(I),NITRO(I),
     &                       LAYERC(I,ILAY,IDORN),I=1,NSPECS)
  101 CONTINUE
 1009 FORMAT (A100)
 1013 FORMAT (A8)
 1017 FORMAT (A4,2X,L1,3X,E10.4)
```

TABLE 2.169. CLEANAIR PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	HDTEXT		Char*100	Header text record
2	NORD		Char*8	NIGHT or DAY indicator
3	INLAY		Char*8	Layer number
4	$SPNAME_{L,i}$		Char*4	Chemical species name / abbreviation
5	$NITRO_{L,i}$		Logical*4	Flag: = .TRUE. for all nitrogenous species
6	$LAYERC_{L,i}$	ppm	Real*4	Species concentration

Note: L = layer number, i = 1 (day) or 2 (night)

b. RATECONS

RATECONS, created by preprocessor CVCHEM, contains the rate constants for the reactions of the Carbon-bond 4.2 mechanism. The rates have been adjusted to meteorological parameters by CVCHEM (Section 7.2, Part 1). The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 2.170.

```
      READ (JRATE,1009) HDTEXT
      READ (JRATE,1009) HDTEXT
1009  FORMAT (A100)
      DO 103 IDORN=1,2
        READ (JRATE,1013) NORD
          DO 101 ILAY=1,INLAY
          READ (JRATE,1013) INLAY
          READ (JRATE,1027) (RK(I,ILAY,IDORN),I=1,NREACT)
101   CONTINUE
1013  FORMAT (A8)
1027  FORMAT (5E15.0)
```

TABLE 2.170. RATECONS PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	HDTEXT		Char*100	Header text record
2	NORD		Char*8	NIGHT or DAY indicator
3	INLAY		Char*8	Layer name
4	$RK(1)_{i,L}$	min^{-1}	Real*4	Rate constant, species 1
.	.		.	.
.	.	or	.	.
.	.		.	.
86	$RK(83)_{i,L}$	$ppm^{-1} \cdot min^{-1}$	Real*4	Rate constant, species 83

Note: i = 1 (day) or 2 (night) indicator; L = layer. Units are dependent on the number of reacting gases.

c. OZONE

OZONE, created by preprocessing AIRS data (Section 7.1, Part 1) contains the observed surface ozone concentrations for each of the four model boundaries. The READ and FORMAT statements for our version of this file are listed below, and its parameters are shown in Table 2.171. Note that you may decide on a different format for OZONE, in which case you will need to reflect your changes in the code shown below.

```
      READ (JRATE,1009) HDTEXT
      READ (JRATE,1009) HDTEXT
1009  FORMAT (A100)
      READ (JO3,1031) JDATE, TIME, (O3DAY1(IBND), IBND=1,NBOUND)
1031  FORMAT (I5,2X,I2,2X,4I5)
```

TABLE 2.171. OZONE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	HDTEXT		Char*100	Header text record
2	JDATE		Integer*4	Julian date as *YYDDD*
3	TIME	EST	Integer*4	Time as *HH*
4	$O3DAY1_b$	ppm	Real*4	Ozone concentration

Note: b = boundary

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

P24G creates one nonstandard file, EQUILBCS, that contains the twice daily (daytime and nighttime) steady-state species concentrations at all four boundaries for the 35 species used in the ROM. Note that there is only a single concentration for each of the species at each boundary. EQUILBCS is used by P22G. The WRITE and FORMAT statements for this file are listed below, and its parameters are shown in Table 2.172.

```
      WRITE (OUTUNT, 1037) JDATE, TIME
1037  FORMAT (T1,I5.5,2X,I2.2)
      DO 213 ILAY=1,NLAY
         DO 211 ISPC=1,NSPC
         WRITE (OUTUNT,1043) SPNAME(ISPC), (BCONC(IBND,ISPC),IBND=1,NBOUND)
1043  FORMAT (A4,<NBOUND>(2X,E10.4))
 211     CONTINUE
 213  CONTINUE
```

TABLE 2.172. EQUILBCS PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	JDATE		Integer*4	Julian date as *YYDDD*
2	TIME	EST	Integer*4	Time as *HH*
3	$SPNAME_L$		Char*4	Species name
4	$BCONC_{i,s,L,b}$	ppm	Real*4	Equilibriated boundary concentration

Note: L = layer, i = 1 (day) or 2 (night) indicator, s = species, b = boundary

III. CONTROL CARDS

One control card is used, in the format shown below. Table 2.173 describes the control card variables.

DELTAT TSP

TABLE 2.173. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Description
1	DELTAT	min	Maximum chemistry equilibriation time
2	TSP	min	Time between reported concentrations to the log

Example:

90.0 90.0

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	29	blocks
FORTRAN INCLUDE files:	7	files	21	blocks
Object files:	1	file	13	blocks
Executable file:	1	file	45	blocks
	10	files	108	blocks

B. Execution Time Requirements (Representative Values for a 20-Day Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:09:38
Buffered I/O count:	293
Direct I/O count:	2676
Peak working-set size:	552
Peak virtual size:	2528

C. Space Requirements: Log and Print Files

P24GX1.LOG:	2292 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.174 shows the input file and output file space requirements.

TABLE 2.174. I/O FILE SPACE REQUIREMENTS

File Group	Logical File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	CLEANAIR	Nonstd	11	N/A
	RATECONS	Nonstd	22	N/A
	OZONE	Nonstd	11	20 days
			44	
Output	EQUILBCS	Nonstd	462	20 days

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P24GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P24GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P24GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P24GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P24G.OBJ             A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ            A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GPRIME.OBJ           A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ             A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ           A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ            A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PQ1CB42.OBJ          A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PQCOEFCB42.OBJ       A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ           A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RATEDCB42.OBJ        A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]TOTAL.OBJ            A11
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P24GV1A.EXE -
  /MAP = MET1:[S01S]P24GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
```

```
  A9, -
  A10, -
  A11
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P24GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P24GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1502A]P24GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                  PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]CLEANAIR.DAT         CLEANAIR
$ ASSIGN MET1:[ROMNET.RAWDATA]OZONEJUL85.DAT       OZONE
$ ASSIGN MET1:[ROMNET.RAWDATA]RATECB4-2.DAT        RATECONS
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET1:[ROMNET.RAWDATA]EQBC_RNETJUL85.DAT   EQUILBCS
$ SHOW PROCESS / ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARD(S) FILE
$ TYPE USER2$DISK:[FOREMAN.FV1502A]P24GX1.CPR
$ !
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1502A]P24GX1.CPR   SYS$INPUT
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P24GV1A.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P24GV1A

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

GPRIME
PQ1CB42
PQCOEFCB42
RATEDCB42
TOTAL

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

PARM24.EXT
PRID.EXT

2.23 PROCESSOR P25G (LAND USE)

I. . PROCESSOR FUNCTION

In 1980, the EPA laboratory at Las Vegas, Nevada, delivered the National Land Use and Land Cover Inventory to the Regional Oxidant Modeling program. The inventory includes all of the United States and Canada to 55 °N latitude. Land use data were assigned on the basis of 20-km grid cells covering the inventory area as a percentage of 10, later 11, land use and land cover categories in each cell. Note that actual grid cells are a constant $^1/_4$ degree longitude by $^1/_6$ degree latitude. This is nominally 20 kilometers on a side, but in fact the longitudinal distance varies from 16 km to 25 km between 25 °N and 55 °N respectively. The latitudinal distance is a constant 18.5 km over the whole domain. The data consist of a tape record of the latitude and longitude of the center point of each cell, together with the cell's land cover categories. Land use and land cover data were derived from the U.S. Geological Survey's Land Use and Land Cover (LULC) 1:250000 scale map series (circa 1972), from Landsat regional mosaic images of the same date for areas with no LULC data, and from Landsat spot coverages over major metropolitan areas where needed.

The set of 11 land use categories are: (1) urban land, (2) agricultural land, (3) rangeland, (4) deciduous forest, (5) coniferous forest, (6) mixed forest (including forested wetland), (7) water, (8) barren land, (9) nonforested wetland, (10) mixed agricultural land and rangeland, and (11) rocky open places occupied by low shrubs and lichens.

This stage 0 processor extracts land use data from LUSE1, the land use master-file tape, and computes, for every grid cell, the fraction of each land use category in the cell. Note that you need to run the processor only once for each modeling domain.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

P25G uses one nonstandard input file, LUSE1, that contains land use percentages for each land use category by latitude and longitude. LUSE1 is read in using the READ and FORMAT statements below, and Table 2.175 shows its parameters.

```
      READ (LUSEUN, 93001, IOSTAT=IOST) LAT, LATFRC, LONG,
     &      LONFRC, (LNDUSE(I), I = 1,NPARM)
93001 FORMAT (4I4,11I5)
```

TABLE 2.175. LUSE1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	LAT	° N	Integer*4	Latitude
2	LATFRC	1/12 ° N	Integer*4	Latitude in twelfths
3	LONG	° W	Integer*4	Longitude
4	LONFRC	1/8 ° W	Integer*4	Longitude in eighths
				Percent of land in a cell that is:
5	LNDUSE(I)	%	Integer*4	urban
6	LNDUSE(I)	%	Integer*4	agricultural
7	LNDUSE(I)	%	Integer*4	rangeland
8	LNDUSE(I)	%	Integer*4	deciduous forest
9	LNDUSE(I)	%	Integer*4	coniferous forest
10	LNDUSE(I)	%	Integer*4	mixed forest (includes wetlands)
11	LNDUSE(I)	%	Integer*4	water
12	LNDUSE(I)	%	Integer*4	outside study area
13	LNDUSE(I)	%	Integer*4	nonforested wetland
14	LNDUSE(I)	%	Integer*4	mixed agricultural and rangeland
15	LNDUSE(I)	%	Integer*4	rocky open areas occupied by low-growing shrubs and lichens

B. Output Files

1. Standard Output Files

a. PF118

This file contains the land use fractions for all land use categories in each grid cell; it is used by processors P04G and P15G. Table 2.176 shows its parameters.

TABLE 2.176. PF118 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Fraction of land in a cell that is:
1	URBAN		Real*4	urban
2	AGRI		Real*4	agricultural
3	RANGE		Real*4	rangeland
4	DF		Real*4	deciduous forest
5	EV		Real*4	evergreen forest
6	MF		Real*4	mixed forest (includes wetlands)
7	WATER		Real*4	water
8	BARREN		Real*4	barren
9	NFW		Real*4	nonforested wetland
10	MAR		Real*4	mixed agriculture/rangeland
11	ROCKY		Real*4	rocky, open

2. Nonstandard Output Files

None

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	12	blocks
FORTRAN INCLUDE files:	2	files	26	blocks
Object files:	1	file	7	blocks
Executable file:	1	file	46	blocks
	5	files	91	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:00:32
Buffered I/O count:	173
Direct I/O count:	462
Peak working-set size:	692
Peak virtual size:	1613

C. Space Requirements: Log and Print Files

P25GV1A.LOG:	11 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.177 shows the input file and output file space requirements.

TABLE 2.177. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	LUSE1	Nonstd	3774	N/A
Output	PF118	4	286	N/A

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P25GV1A follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P25GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P25GV1A.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P25GV1A.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P25G.OBJ              A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ             A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELL0.OBJ             A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELL1.OBJ             A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ            A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ            A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ             A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ              A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOERR.OBJ             A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ            A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE2.OBJ            A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ             A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ            A13
```

```
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ                 A14
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ                A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ                A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ                 A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT4.OBJ                 A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ                 A19
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P25GV1A.EXE -
  /MAP = MET1:[S01S]P25GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A17, -
  A18, -
  A19
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P25GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P25GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1050A]P25GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                  PFMFDIR
$ ASSIGN MET1:[ROMNET.RAWDATA]LNDUSE1.DAT          LUSE1
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET1:[ROMNET.D00000]PF118A.DAT            PF118
$ SHOW PROCESS/ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1050A]P25GX1.CPR
$ RUN USER2$DISK:[ROMNET.P25G]P25G
$ SHOW PROCESS/ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1000A        P25GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P25GV1A

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	FSTAT1	OPEN2	RDREC
CELL0	IBLKR	PROGID	WRFT4
CELL1	JFILE1	RDDIR1	WRREC
FLSMRY	OPDIR2		

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

PARMS.EXT

2.24 PROCESSOR P26G (MOBILE-SOURCE EMISSIONS)

I. PROCESSOR FUNCTION

This stage 1 processor prepares mobile-source emissions data for use in the emissions processors. The original mobile-source data are transformed by control efficiency adjustment factors[10] and by temperature adjustments. The adjustment factors are time-dependent because they are a function of the daily average temperature in each grid cell. The output file contains gridded, hourly, adjusted, mobile-source emissions.

10. The magnitude of hydrocarbon emissions contained in the raw data sets represent "uncontrolled" emission levels. An effective emissions reduction has been realized at certain locations within the NAPAP inventory domain as a result of legislation enacted by various political bodies. Alliance Technologies provided estimates of the control efficiencies for gasoline-exhaust hydrocarbon emissions. These estimates, or "county control efficiency factors", were made on a city or county basis. We translated the county-based factors to the NAPAP grid system, and apply the gridded factors to the weekday and weekend data sets. The nonstandard input file BASECEFFAC contains these gridded factors.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

PF192 from TEMPROC.

2. Nonstandard Input Files

P26G uses seven nonstandard input files.

a. MOBLSAZ, MOBLSUZ, and MOBLWKZ.

These three files contain the hourly mobile-source emissions data for Saturday, Sunday, and weekdays respectively. The READ and FORMAT statements for these files are as follows, and the files' parameters are shown in Table 2.178.

```
      READ(INUNIT, 100, IOSTAT=IOST)ICOL, IROW, JHOUR,
     &             (MBEM(ISPC), ISPC = 1, NSPCMB), CNTYID
100   FORMAT(2I4, I2, 8E10.3, A6)
```

where NSPCMB = 8.

TABLE 2.178. MOBLSAZ, MOBLSUZ, AND MOBLWKZ PARAMETERS

Parm No.	Parm Name	Unit [a]	Data Type	Description
1	ICOL		Integer*4	NAPAP column number
2	IROW		Integer*4	NAPAP row number
3	JHOUR	GMT	Integer*4	Start hour
4	MBEM(1)	$tons \cdot h^{-1}$	Real*4	Mobile CO emissions
5	MBEM(2)	$tons \cdot h^{-1}$	Real*4	Mobile NO emissions
6	MBEM(3)	$tons \cdot h^{-1}$	Real*4	Mobile NO_2 emissions
7	MBEM(4)	$tons \cdot h^{-1}$	Real*4	Mobile evap. emissions
8	MBEM(5)	$tons \cdot h^{-1}$	Real*4	Mobile exhaust emissions
9	MBEM(6)	$tons \cdot h^{-1}$	Real*4	Mobile diesel emissions
10	MBEM(7)	$tons \cdot h^{-1}$	Real*4	Mobile VOC emissions
11	MBEM(8)	$tons \cdot h^{-1}$	Real*4	Mobile THC emissions
12	CNTYID		Char*6	AEROS State/County code [b]

a. Units are in English short $tons \cdot h^{-1}$; 1 ton = 907.20 kg. NO emissions are reported as weight of NO_2.

b. AEROS = Aerometric and Emissions Reporting System. For more information on AEROS, refer to AEROS Manual Volume 5, *AEROS Manual of Codes*, EPA-450/2-76-005a.

b. PCTCELL

PCTCELL is based on a Geographic Information System (GIS) file; it assigns percentages of grid cell areas that belong to different counties. For each grid cell, the number of counties within that cell is given as one record. Below that record, there is a record for each county in that cell. These records contain the county's name, identification number, and percent of the grid cell consisting of that county. The READ and FORMAT statements used for this file are listed below. Note that there are other variables in this file that P26G does not use. PCTCELL parameters are described in Table 2.179.

```
      READ(INUNIT, 100, IOSTAT=IOST) NCNTY
100   FORMAT(1X, I2)
      DO 200 I = 1, NCNTY
        READ(INUNIT, 110, IOSTAT=IOST) ICOL, IROW, CCODE, PCTCL
110     FORMAT(1X, I2, 1X, I2, 1X, A6, 39X, F7.3)
200   CONTINUE
```

TABLE 2.179. PCTCELL PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ICOL		Integer*4	Column number
2	IROW		Integer*4	Row number
3	CCODE		Char*6	County AEROS code
4	PCTCL	%	Real*4	Percent of cell in the county

c. REFORMTFAC

This file originated from Alliance Technologies, and has been made into a direct access file (by including a header portion) to speed processing in the ROM. It contains the temperature adjustment factors that are used on the mobile-source emissions data. These factors are required because Alliance's mobile-source emissions data are reported as "unadjusted," i.e., only at a temperature of 85 °F with a 20 °F range. These "unadjusted" mobile-source emissions need to be adjusted for air temperature, and must be corrected for missing running losses. Further information on mobile-source emissions can be obtained from the ROMNET Final Project Report (EPA, 1990). REFORMTFAC has data for one or more years. The header on the file contains information regarding (1) which years are included, and (2) the temperature ranges; this information also details how many partitions separate the maximum and minimum of the range. New data, consisting of a year's temperature adjustment factors, must be appended to the file, and the header changed to reflect this addition. An example of the header follows; the first line indicates that the file contains three years of data. Clearly, the years 1995 and 2005 represent projections. The second line indicates that the temperatures are partitioned into 12 segments from 40.00 °F to 95.00 °F. Lines 3-14 indicate that the 12 segments are equally incremented by 5 °F, and that the range for each segment lies between 0.00 °F and 40.00 °F. Lines 2-14 are repeated for each year.

```
3  1985  1995  2005
       12  40.00  95.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
               5    0.00     40.00
```

READ and FORMAT statements for REFORMTFAC are listed below, and its parameters are described in Table 2.180.

```
             READ(INUNIT, 100, REC=IREC) YEAR, TFAC_MEAN, TFAC_RANGE,
            &       VOC_FAC, (EM_FAC(IFAC), IFAC = 1, NFAC)
       100   FORMAT(1X, I4, 5X, 2(1X, F4.0), (1X, F8.5), 5(1X, F8.5))
```

where NFAC = 5.

TABLE 2.180. REFORMTFAC PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	YEAR		Integer*4	Control year
2	TFAC_MEAN	°F	Real*4	Mean temperature
3	TFAC_RANGE	°F	Real*4	Temperature range
4	VOC_FAC		Real*4	VOC adjustment factor
5	EM_FAC(1)		Real*4	Evaporative adjustment factor
6	EM_FAC(2)		Real*4	Gasoline adjustment factor
7	EM_FAC(3)		Real*4	Diesel adjustment factor
8	EM_FAC(4)		Real*4	NO_x adjustment factor
9	EM_FAC(5)		Real*4	CO adjustment factor

d. BASECEFFAC

Motor vehicle "county control efficiency factors" are kept in this file. The file has a seven line header containing a description of its contents. The header lines are read using the following statements:

```
      DO 101 IREC = 1, 7
        READ(INUNIT, 1001, IOSTAT=IOST) CLABEL
1001    FORMAT(A51)
 101  CONTINUE
```

READ and FORMAT statements for BASECEFFAC are listed below, and its parameters are described in Table 2.181.

```
      READ(INUNIT, 1003, IOSTAT=IOST) BCEID(ICNTY),
     &     (BCEBUF(ICNTY, IFC), IFC = 1, NFAC
1003  FORMAT(A6, 5(F9.5))
```

TABLE 2.181. BASECEFFAC PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	BCEID(I)		Char*6	State/county id.
2	BCEBUF(I,1)		Real*4	Evaporative efficiency factor
3	BCEBUF(I,2)		Real*4	Gas exhaust efficiency factor
4	BCEBUF(I,3)		Real*4	Diesel efficiency factor
5	BCEBUF(I,4)		Real*4	NO_x efficiency factor
6	BCEBUF(I,5)		Real*4	CO efficiency factor

e. SPECFAC

SPECFAC files contain annual speciation factors for mobile-source emissions data, and originate from Alliance Technologies. The file contains 33 speciation factors, 11 each for evaporative species, diesel exhausts, and gasoline exhausts. Speciation factors are used to derive the 11 species of hydrocarbons used by the ROM from the three types of exhausts above. The evaporative species factors derive from non-exhaust sources, such as oil and gas leakage from vehicles, and are read using the following statements:

```
      DO 100 ISPC = 1, 11
        READ(INUNIT, 1005, IOSTAT=IOST) SFACV(ISPC), FACNAM
1005    FORMAT(1X, E8.3, 11X, A4)
100   CONTINUE
```

The 11 diesel exhaust factors are then read with the following statement:

```
      DO 200 ISPC = 1, 11
        READ(INUNIT, 1005, IOSTAT=IOST) SFACD(ISPC), FACNAM
200   CONTINUE
```

Finally, the 11 gasoline exhaust speciation factors are read.

```
      DO 300 ISPC = 1, 11
        READ(INUNIT, 1005, IOSTAT=IOST) SFACG(ISPC), FACNAM
300   CONTINUE
```

SPECFAC parameters are described in Table 2.182.

TABLE 2.182. SPECFAC PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
Evaporative				
				Speciation factors for:
1	SFACV(1)	mol·kg^{-1}	Real*4	olefins
2	SFACV(2)	mol·kg^{-1}	Real*4	paraffins
3	SFACV(3)	mol·kg^{-1}	Real*4	toluene
4	SFACV(4)	mol·kg^{-1}	Real*4	xylene
5	SFACV(5)	mol·kg^{-1}	Real*4	formaldehyde
6	SFACV(6)	mol·kg^{-1}	Real*4	aldehydes
7	SFACV(7)	mol·kg^{-1}	Real*4	ethylene
8	SFACV(8)	mol·kg^{-1}	Real*4	isoprenes
9	SFACV(9)	mol·kg^{-1}	Real*4	nonreactive hydrocarbons
10	SFACV(10)	mol·kg^{-1}	Real*4	methane
11	SFACV(11)	mol·kg^{-1}	Real*4	methanol
12	FACNAM		Char*4	Species names
Diesel exhaust				
13	SFACD(1)	mol·kg^{-1}	Real*4	Diesel exhaust speciation factors as above.
.	.			
.	.			
.	.			
23	SFACD(11)	mol·kg^{-1}	Real*4	
24	FACNAM		Char*4	Species names
Gasoline exhaust				
25	SFACV(1)	mol·kg^{-1}	Real*4	Gasoline exhaust speciation factors as above.
.	.			
.	.			
.	.			
35	SFACV(11)	mol·kg^{-1}	Real*4	
36	FACNAM		Char*4	Species names

B. Output Files

1. Standard Output Files

P26G produces two standard output files, PF191 and PF167. PF167 is used by P14G. PF191 is not used by any processor; it is a quality-control file.

a. PF167

This file contains hourly, gridded, temperature-adjusted, mobile-source VOC, NO_x, and CO emissions. Table 2.183 shows the file's parameters.

TABLE 2.183. PF167 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Emission rates for:
1	ALD2	$mol \cdot h^{-1}$	Real*4	aldehydes
2	CO	$mol \cdot h^{-1}$	Real*4	CO
3	ETH	$mol \cdot h^{-1}$	Real*4	ethylene
4	FORM	$mol \cdot h^{-1}$	Real*4	formaldehydes
5	ISOP	$mol \cdot h^{-1}$	Real*4	isoprenes
6	NO	$mol \cdot h^{-1}$	Real*4	NO
7	NO2	$mol \cdot h^{-1}$	Real*4	NO_2
8	OLE	$mol \cdot h^{-1}$	Real*4	olefins
9	PAR	$mol \cdot h^{-1}$	Real*4	paraffins
10	TOL	$mol \cdot h^{-1}$	Real*4	toluene
11	XYL	$mol \cdot h^{-1}$	Real*4	xylene
12	NONR	$mol \cdot h^{-1}$	Real*4	nonreactive hydrocarbons
13	METH	$mol \cdot h^{-1}$	Real*4	methane
14	MTHL	$mol \cdot h^{-1}$	Real*4	methanol
15	VOC	$kg \cdot h^{-1}$	Real*4	volatile organic compounds
16	RATIO		Real*4	Nonmethane hydrocarbons:NO_x ratio

b. PF191

This file contains the gridded-temperature and control-efficiency factors that are applied to the Alliance "unadjusted" mobile-source emissions data. Table 2.184 shows the file's parameters.

2. Nonstandard Output Files

None

III. CONTROL CARDS

Two control cards are used to input control data in the format shown below. Table 2.185 defines the control card variables.

YEAR REGION TADJ IMPE
IDATE IHOUR NHRS

TABLE 2.184. PF191 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Temperature adjustment factors for:
1	TAFACV		Real*4	evaporative THC's
2	TAFACG		Real*4	gasoline exhaust THC's
3	TAFACD		Real*4	diesel exhaust THC's
4	TAFACN		Real*4	NO_x
5	TAFACC		Real*4	CO
				Control efficiency adjustment factors for:
6	CEFACV		Real*4	evaporative THC's
7	CEFACG		Real*4	gasoline exhaust THC's
8	CEFACD		Real*4	diesel exhaust THC's
9	CEFACN		Real*4	NO_x
10	CEFACC		Real*4	CO

TABLE 2.185. CONTROL CARD VARIABLES

Variable Name	Unit	Description
YEAR		Emissions data year as *YYYY*
REGION		Modeling domain name
TADJ		Temperature adjustment factors flag: .FALSE. ⇒ do not apply .TRUE. ⇒ apply
IMPE		Control efficiency factors flag: .FALSE. ⇒ do not apply .TRUE. ⇒ apply
IDATE		Julian scenario start date as *YYDDD*
IHOUR	EST	Scenario start time as *HH*
NHRS		No. of hours in the scenario

Example:

1985 'ROMNET1 ' .TRUE. .TRUE.
88184 12 72

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	6	files	92	blocks
FORTRAN INCLUDE files:	5	files	29	blocks
Object files:	6	files	57	blocks
Executable file:	1	file	87	blocks
	18	files	265	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:09:52
Buffered I/O count:	242
Direct I/O count:	9304
Peak working-set size:	2019
Peak virtual size:	4242

C. Space Requirements: Log and Print Files

P26GX1.LOG:	26 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.186 shows the input file and output file space requirements.

TABLE 2.186. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type [a]	Storage (in blocks)	Scenario Data Span
Input	PF192	5	1014	72 hours
	MOBLSAZ.CS0	Nonstd	26075	24 hours
	MOBLSUZ.CS0	Nonstd	26075	24 hours
	MOBLWKZ.CS0	Nonstd	26075	24 hours
	PCTCELL	Nonstd	695	
	REFORMTFAC	Nonstd	33	
	BASECEFFAC	Nonstd	9	
	SPECFAC	Nonstd	3	
			79979	
Output	PF191	5	1066	72 hours
	PF167	5	29978	72 hours
			31044	

a. Note that PF191 and PF192 are pseudo-type 5 files; their data is listed (by *YYDDD HH*) as, e.g., 88194 12, 88195 12, 88196 12... instead of 88194 12, 88194 13, 88194 14....

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P26GV1B.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P26GV1B.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P26GV1B.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P26GV1B.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P26G.OBJ              A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]BLKMOB.OBJ            A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CTREFF.OBJ            A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]OFFNDX.OBJ            A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]PROCMB.OBJ            A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]RDTFAC.OBJ            A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ             A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ            A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ            A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ             A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]INDEX1.OBJ            A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ              A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ            A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE2.OBJ            A14
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JHOUR1.OBJ            A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ             A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]KDAY.OBJ              A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ            A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ             A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ            A20
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ            A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT5.OBJ             A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ             A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT5.OBJ             A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ             A25
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P26GV1B.EXE -
  /MAP = MET1:[S01S]P26GV1B.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A23, -
  A24, -
  A25
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P26GV1B.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P26GX1.COM follows.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM in USER2$DISK:[FOREMAN.FV1503A]P26GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                       PFMFDIR
$ ! inputs:
$ ASSIGN MET52:[EMISSIONS.RNSUM85M]MOBLSAZ.CSO          MBSA
$ ASSIGN MET52:[EMISSIONS.RNSUM85M]MOBLSUZ.CSO          MBSU
$ ASSIGN MET52:[EMISSIONS.RNSUM85M]MOBLWKZ.CSO          MBWK
$ ASSIGN MET52:[EMISSIONS.RNSUM85M]SPFAC85.DAT          SPECFAC
$ !
$ ASSIGN MET1:[EMISSIONS.RNSUM85M]PCTCELLMOB.DAT        PCTCELL
$ ASSIGN MET1:[EMISSIONS.RNSUM85M]RETFAC1.DAT           REFORMTFAC
```

```
$ ASSIGN MET1:[EMISSIONS.RNSUM85M]IM85CNTY.DAT        BASECEFFAC
$ ASSIGN MET1:[ROMNET1.D85182]PF192A.DAT              PF192
$ ! outputs:
$ ASSIGN MET58:[ROMNET1.D85188]PF191A.DAT             PF191
$ ASSIGN MET58:[ROMNET1.D85188]PF167A.DAT             PF167
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1503A]P26GX1.CPR
$ ASSIGN USER2$DISK:[FOREMAN.FV1503A]P26GX1.CPR       FOR005
$ !
$ ASSIGN $2$DUA24:[ROMEXE.ROMNET]P26GV1A.EXE          PROCP26G
$ DIRECTORY / FULL PROCP26G
$ RUN PROCP26G
$ !
```
(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P26GV1B

B. Subroutines

CTREFF
OFFNDX
PROCMB
RDTFAC

C. Functions

None

D. Block Data Files

BLKMOB

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	JFILE1	PROGID	RDREC
FLSMRY	OPDIR2	RDDIR1	WRFT5
FSTAT1	OPEN2	RDFT5	WRREC
IBLKR			

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

INDEX1	JHOUR1
IOCL	JUNIT
JFILE2	KDAY

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

ADJFAC.EXT
MOBPRM.EXT
PRID.EXT
SPCFAC.EXT

2.25 PROCESSOR P27G (BIOGENIC EMISSIONS)

I. PROCESSOR FUNCTION

This stage 6 processor prepares biogenic emissions data for use in the emissions processors. The concentration data generated by P27G (the Biogenic Emissions Inventory System) are grouped into seven of the carbon-bond-chemistry species carried in the ROM: olefins, paraffins, isoprenes, aldehydes, nonreactive hydrocarbons, NO, and NO_2.

II. I/O COMPONENTS

In the following tables, *i* and *j* refer to the grid column and row number respectively, and *mm* is a two-letter month abbreviation.

A. Input Files

1. Standard Input Files

PF103 from P04G
MF139 from P09G
PF190 from subroutine SOLENG within P27G[11]

11. SOLENG computes total (air plus clouds) atmospheric attenuation of incident solar energy flux as a function of wavelength and solar zenith angle.

2. Nonstandard Input Files

a. FLUX.DAT contains actinic (spherically integrated) flux data as a function of zenith angle and wavelength, but not as a function of latitude (Demerjian *et al.*, 1980). The file's gridded columns represent ten different zenith angles: 0°, 10°, 20°, 30°, 40°, 50°, 60°, 70°, 78°, and 86°. There are 52 rows in the grid that represent different wavelengths of the solar spectrum, from 290 nm (ultraviolet) to 800 (near infrared). Each wavelength increment is 10 nm. The READ and FORMAT statements for FLUX.DAT are listed below, and its parameters are shown in Table 2.187.

```
     READ (INUNIT,100) (XJ(I,J), J=1,10)
100  FORMAT (10F10.7)
```

TABLE 2.187. FLUX.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Actinic flux at:
1	XJ(1,1)	photons·cm^{-2}s^{-1}	Real*4	zenith angle = 10°, λ = 290 nm
2	XJ(1,2)	photons·cm^{-2}s^{-1}	Real*4	zenith angle = 20°, λ = 290 nm
3	XJ(1,3)	photons·cm^{-2}s^{-1}	Real*4	zenith angle = 30°, λ = 290 nm
.	.	.		.
.	.	.		.
.	.	.		.
520	XJ(10,52)	photons·cm^{-2}s^{-1}	Real*4	zenith angle = 86°, λ = 800 nm

Note: λ = wavelength in nanometers.

b. SRFC_*MMMYY*.DAT (from the ROM meteorology preprocessors) contains surface meteorology and cloud data. The READ and FORMAT statements for the header portion of this file are as follows:

```
     READ (INUNIT,100,END=75,ERR=80) YR, IM, ID, MTHOUR, NSTA
100  FORMAT (4I2,I8)
```

If the data contained in the file are in WBAN format, then the following statements are used:

```
        READ ( INUNIT , 121 , END=75 , ERR=80 )
     &      LAT, LONG, COVER, DESCRI,
     &      PERCNT(1), HGT(1), PERCNT(2),
     &      HGT(2), PERCNT(3), HGT(3)
121     FORMAT (5X,F6.2,F7.2,F5.0,I3,1X,3(1X,F4.0,F6.0))
```

If the data contained in the file are not in WBAN format, then the following statements are used:

```
      READ ( INUNIT , 120 , END=75 , ERR=80 )
     &       LAT, LONG, COVER, DESCRI,
     &       PERCNT(1), HGT(1), PERCNT(2),
     &       HGT(2), PERCNT(3), HGT(3)
120   FORMAT (4X,2F7.2,F5.0,I3,1X,3(1X,F4.0,F6.0))
```

Table 2.188 shows the parameters for SRFC_*MMMYY*.DAT; since the station ID information is skipped by the READ statement, there are no differences between WBAN and non-WBAN data in this input file.

TABLE 2.188. SRFC_*MMMYY*.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	YR		Integer*4	Year
2	IM		Integer*4	Month
3	ID		Integer*4	Day
4	MTHOUR	GMT	Integer*4	Hour
5	NSTA		Integer*4	Number of Stations
6	LAT_n	° N	Real*4	Latitude
7	$LONG_n$	° W	Real*4	Longitude
8	$COVER_n$		Real*4	Total Opaque Skycover
9	$DESCRI_n$		Integer*4	Sky Coverage 0 = clear 1 = partial obscuration 2 = thin scattered 3 = thin broken 4 = thin overcast 5 = scattered 6 = broken 7 = overcast 8 = obscured -99 = indeterminate
10	$PERCNT(1)_n$	%	Real*4	Coverage, lowest cloud layer
11	$HGT(1)_n$	m	Real*4	Lowest cloud base
12	$PERCNT(2)_n$	%	Real*4	Coverage, second-lowest cloud layer
13	$HGT(2)_n$	m	Real*4	Second-lowest cloud base
14	$PERCNT(3)_n$	%	Real*4	Coverage, third-lowest cloud layer
15	$HGT(3)_n$	m	Real*4	Third-lowest cloud base

Note: n = station number (one or more digits)

c. GCNBIO*mm*.DAT

This file contains the gridded canopy foliar biomass by layer, and is derived from the biogenics preprocessors (see Section 6, Part 1). The canopy layering is based on the assumption that the biomass has a vertical distribution through the canopy. This file's READ and FORMAT statements are listed below, and its parameters are shown in Table 2.189.

```
      READ(GCBIO, 2001, IOSTAT = IOST1)CCOL, CROW, (CNBIO(I),I=1,32)
2001  FORMAT(2(1X,I2),32(F12.3))
```

TABLE 2.189. GCBIO*mm*.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CCOL		Integer*4	Column number
2	CROW		Integer*4	Row number
3	CNBIO(1)	kg	Real*4	High isoprene layer 1
4	CNBIO(2)	kg	Real*4	High isoprene layer 2
5	CNBIO(3)	kg	Real*4	High isoprene layer 3
6	CNBIO(4)	kg	Real*4	High isoprene layer 4
7	CNBIO(5)	kg	Real*4	High isoprene layer 5
8	CNBIO(6)	kg	Real*4	High isoprene layer 6
9	CNBIO(7)	kg	Real*4	High isoprene layer 7
10	CNBIO(8)	kg	Real*4	High isoprene layer 8
11	CNBIO(9)	kg	Real*4	Low isoprene layer 1
12	CNBIO(10)	kg	Real*4	Low isoprene layer 2
13	CNBIO(11)	kg	Real*4	Low isoprene layer 3
14	CNBIO(12)	kg	Real*4	Low isoprene layer 4
15	CNBIO(13)	kg	Real*4	Low isoprene layer 5
16	CNBIO(14)	kg	Real*4	Low isoprene layer 6
17	CNBIO(15)	kg	Real*4	Low isoprene layer 7
18	CNBIO(16)	kg	Real*4	Low isoprene layer 8
19	CNBIO(17)	kg	Real*4	No isoprene layer 1
20	CNBIO(18)	kg	Real*4	No isoprene layer 2
21	CNBIO(19)	kg	Real*4	No isoprene layer 3
22	CNBIO(20)	kg	Real*4	No isoprene layer 4
23	CNBIO(21)	kg	Real*4	No isoprene layer 5
24	CNBIO(22)	kg	Real*4	No isoprene layer 6
25	CNBIO(23)	kg	Real*4	No isoprene layer 7
26	CNBIO(24)	kg	Real*4	No isoprene layer 8
27	CNBIO(25)	kg	Real*4	Coniferous layer 1
28	CNBIO(26)	kg	Real*4	Coniferous layer 2
29	CNBIO(27)	kg	Real*4	Coniferous layer 3
30	CNBIO(28)	kg	Real*4	Coniferous layer 4
31	CNBIO(29)	kg	Real*4	Coniferous layer 5
32	CNBIO(30)	kg	Real*4	Coniferous layer 6
33	CNBIO(31)	kg	Real*4	Coniferous layer 7
34	CNBIO(32)	kg	Real*4	Coniferous layer 8

d. GNCBIO*mm*.DAT

This file contains the gridded noncanopy land-use area, and is derived from the biogenics preprocessors (see Section 6, Part 1). This file's READ and FORMAT statements are listed below, and its parameters are shown in Table 2.190.

```
      READ(GNBIO, 3001, IOSTAT = IOST2)NCCOL, NCROW, (NCNBIO(I),I=1,19)
3001  FORMAT(2(1X,I2),19(F12.3))
```

TABLE 2.190. GNCBIO*mm*.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	NCCOL		Integer*4	Column number
2	NCROW		Integer*4	Row number
				Gridded land use area for
3	NCBIO(1)	ha	Real*4	Alfalfa
4	NCBIO(2)	ha	Real*4	Sorghum
5	NCBIO(3)	ha	Real*4	Hay
6	NCBIO(4)	ha	Real*4	Soybean
7	NCBIO(5)	ha	Real*4	Corn
8	NCBIO(6)	ha	Real*4	Potato
9	NCBIO(7)	ha	Real*4	Tobacco
10	NCBIO(8)	ha	Real*4	Wheat
11	NCBIO(9)	ha	Real*4	Cotton
12	NCBIO(10)	ha	Real*4	Rye
13	NCBIO(11)	ha	Real*4	Rice
14	NCBIO(12)	ha	Real*4	Peanut
15	NCBIO(13)	ha	Real*4	Barley
16	NCBIO(14)	ha	Real*4	Oats
17	NCBIO(15)	ha	Real*4	Scrub
18	NCBIO(16)	ha	Real*4	Grass
19	NCBIO(17)	ha	Real*4	Urban Grass
20	NCBIO(18)	ha	Real*4	Miscellaneous Crops
21	NCBIO(19)	ha	Real*4	Water

e. GUTBIO*mm*.DAT

This file contains the gridded urban tree-foliage biomass, and is derived from the biogenics preprocessors (see Section 6, Part 1). This file's READ and FORMAT statements are listed below, and its parameters are shown in Table 2.191.

```
      READ(GUBIO, 4001, IOSTAT = IOST3)UCOL, UROW, (URBIO(I),I=1,4)
4001  FORMAT(2(1X,I2),4(F12.3))
```

TABLE 2.191. GUTBIO*mm*.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	UCOL		Integer*4	Column number
2	UROW		Integer*4	Row number
3	URBIO(1)	kg	Real*4	High isoprene biomass
4	URBIO(2)	kg	Real*4	Low isoprene biomass
5	URBIO(3)	kg	Real*4	Nonisoprene biomass
6	URBIO(4)	kg	Real*4	Coniferous biomass

B. Output Files

1. Standard Output Files

PF144

This file contains hourly gridded concentrations of biogenic emissions for olefins, paraffins, isoprenes, aldehydes, nonreactive hydrocarbons, NO, and NO_2; it is used by P10G. Table 2.192 shows the file's parameters.

TABLE 2.192. PF144 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	BIOEMIS(*i,j*,1)	$mol \cdot h^{-1}$	Real*4	Olefins
2	BIOEMIS(*i,j*,2)	$mol \cdot h^{-1}$	Real*4	Paraffins
3	BIOEMIS(*i,j*,3)	$mol \cdot h^{-1}$	Real*4	Isoprenes
4	BIOEMIS(*i,j*,4)	$mol \cdot h^{-1}$	Real*4	Aldehydes
5	BIOEMIS(*i,j*,5)	$mol \cdot h^{-1}$	Real*4	Nonreactive hydrocarbons
6	BIOEMIS(*i,j*,6)	$mol \cdot h^{-1}$	Real*4	Nitric oxide (NO)
7	BIOEMIS(*i,j*,7)	$mol \cdot h^{-1}$	Real*4	Nitrogen dioxide (NO_2)

Note: *i* = grid column number, *j* = grid row number.

2. Nonstandard Output Files

None

III. CONTROL CARDS

Two control cards are used to input control data in the format shown below. Table 2.193 defines the control card variables.

SDATE SHOUR NHRS ITMZON LISTNG
INTFLAG

TABLE 2.193. CONTROL CARD VARIABLES

Variable Name	Unit	Description
SDATE		Julian scenario start date as *YYDDD*
SHOUR	EST	Scenario start time as *HH*
NHRS		No. of hours in the scenario
ITMZON		No. of hours difference from GMT
LISTNG		Attenuation factors flag (y/n) (0.05 to 1.00) 'NO' ⇒ no listing 'YES' ⇒ printout of gridded solar energy attenuation by the atmosphere
INTFLAG		WBAN flag (y/n)

Example:

88196, 12, 72, 05, 'NO'
YES

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	6	files	117	blocks
FORTRAN INCLUDE files:	2	files	25	blocks
Object files:	6	files	42	blocks
Executable file:	1	file	109	blocks
	15	files	293	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	02:21:30
Buffered I/O count:	333
Direct I/O count:	4815
Peak working-set size:	6067
Peak virtual size:	8634

C. Space Requirements: Log and Print Files

P27GX1.LOG:	204 blocks for LISTNG = 'NO' 4812 blocks for LISTNG = 'YES'
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.194 shows the input file and output file space requirements.

TABLE 2.194. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF103	5	5642	72 hours
	PF190	5	3770	72 hours
	MF139	5	1898	72 hours
	FLUX.DAT	Nonstd	10	
	SRFC_*MMMYY*.DAT	Nonstd	40250	1 month
	GCNBIO*mm*.DAT	Nonstd	1863	1 month
	GNCBIO*mm*.DAT	Nonstd	1138	1 month
	GUTBIO*mm*.DAT	Nonstd	172	1 month
			54743	
Output	PF144	5	13130	72 hours

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P27GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P27GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P27GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMCOM.ROMNET]P27GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P27G.OBJ              A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]CLDATN.OBJ            A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]CORRECT.OBJ           A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]FLXEIN.OBJ            A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]PRNTGRD.OBJ           A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]SOLENG.OBJ            A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ             A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]BGRID1.OBJ            A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL0.OBJ             A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL2.OBJ             A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ            A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ            A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ             A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ              A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GETYN.OBJ             A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ            A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ            A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILEX.OBJ            A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ            A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JULIAN.OBJ            A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ             A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ            A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ             A23
```

```
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ                A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ                A25
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD1.OBJ                A26
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD2.OBJ                A27
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ                 A28
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ                 A29
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]SOLELV.OBJ                A30
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT5.OBJ                 A31
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ                 A32
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMCOM.ROMNET]P27GV1A.EXE -
  /MAP = MET1:[S01S]P27GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A30, -
  A31, -
  A32
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P27GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P27GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN USER62$DISK:[XHS.P27G]P27GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                        PFMFDIR
$ ASSIGN USER2$DISK:[FOREMAN.BEIS]FLUX.DAT               FLUX
$ ASSIGN MET23:[FOREMAN.RAWDATA]SRFC_JUL88.DAT           SURMET1
$ ASSIGN MET51:[ROMNET1.D88184]MF139A.DAT                MF139
$ ASSIGN MET51:[ROMNET1.D88184]PF103A.DAT                PF103
$ ASSIGN MET1:[ROMNET.BIOGENIC]GCNBIOJL.DAT              GCBIOMASS
$ ASSIGN MET1:[ROMNET.BIOGENIC]GNCBIOJL.DAT              GNBIOMASS
$ ASSIGN MET1:[ROMNET.BIOGENIC]GUTBIOJL.DAT              GUBIOMASS
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET51:[ROMNET1.D88184]PF190A.DAT                PF190
$ ASSIGN MET51:[ROMNET1.D88184]PF144A.DAT                PF144
$ SHOW PROCESS/ACCOUNTING
$ !
$ ! ASSIGN CONTROL CARDS FILE
$ TYPE USER62$DISK:[XHS.P27G]P27GX1.CPR
$ ASSIGN / USER_MODE USER62$DISK:[XHS.P27G]P27GX1.CPR    SYS$INPUT
$ !
$ RUN USER62$DISK:[XHS.P27G]P27G.EXE
$ SHOW PROCESS/ACCOUNTING
$ EXIT
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P27GV1A

B. Subroutines

CLDATN	PRNTGRD
CORRECT	SOLENG
FLXEIN	

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	IBLKR	OPEN2	RDREC
BGRID1	FSTAT1	PROGID	SOLELV
CELL0	JFILE1	RDDIR1	WRFT5
CELL2	JFILEX	RDFHD1	WRREC
FLSMRY	OPDIR2	RDFT5	

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETYN	JHOUR1
IOCL	JULIAN
JFILE2	JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

PARMS.EXT

2.26 PROCESSOR P29G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 5 processor calculates the wind components that are used by the Core Model. These wind components are calculated as *backtracks* (see Figure 2.2), initially in degrees of latitude and longitude, and output finally in λ-ϕ coordinates. (λ-ϕ coordinates are normalized grid coordinates; the southwest corner of the grid has values $\lambda=1$, $\phi=1$, and the northeast corner has values of λ=(NCOLS), ϕ=(NROWS).) A backtrack is the calculated position of a parcel of air (start at time *t*) at time *t*-1, where the model's time steps have durations of 30 minutes. (Note that the input data are at 60-minute intervals.) Although P29G computes back trajectories at 10-minute intervals to determine the 30-minute position (backtrack) of an air parcel, the trajectories are not stored. P29G calculates backtracks for each grid cell and for each time step.

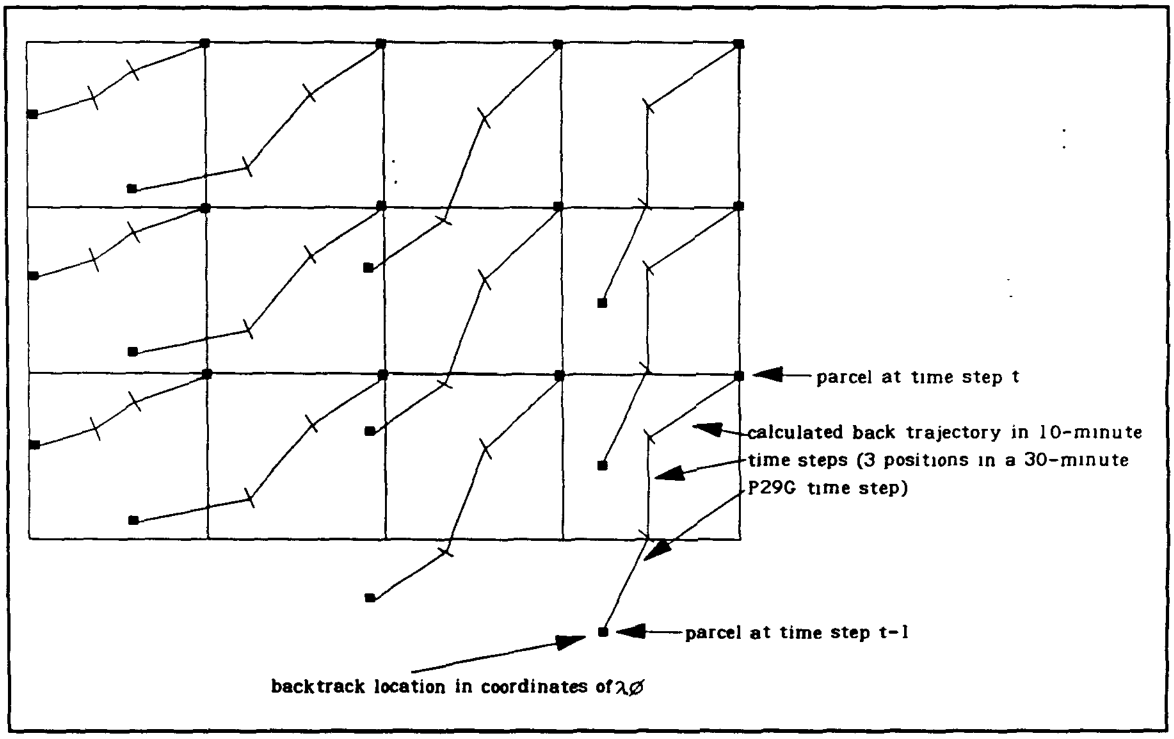


Figure 2.2. Backtracks and back trajectories.

The first scenario simulation of an episode requires the initial conditions for P29G from the input files, via processors P11G (layer wind components), P32G (layer eddy diffusivities), P07G and P08G (layer thicknesses), and P09G (air density correction factors). Note that you must use program PFMFCOPY (Section 7.6, Part 1) to copy the first hour's data from these processors to the files recorded as a - 1 below; these data, with an effective time of 11:00 if the scenario start time is 12:00, are identical to the

12:00 data. Initial conditions for subsequent scenarios are provided by P29G itself, in the form of standard output files a - l (see below).[12] These files contain data from the last hour of a scenario's simulation.

Backtracks are our computational method of choice because the biquintic interpolation schemes of the ROM are faster and easier to implement when regularly-arrayed data are input; backtracks provide regularly-arrayed grids, while calculations forward in time would generate randomly-arrayed grids.

We show below the sequence of calculations of the ROM. Assume that we are start with the second scenario of an episode, i.e., we are at 12:00 p.m. EST.

- Read regularly-arrayed grid concentrations at 11:00.
- Read the backtrack locations for the 11:30 to 11:00 time step.
- Interpolate the backtrack position concentrations biquintically.
- Advect these concentrations forward in time to the grid cell value at 11:30, with corrections for layer thicknesses and diffusion.
- Read the backtrack locations for the 12:00 to 11:30 time step.
- Interpolate the backtrack position concentrations biquintically.
- Advect these concentrations forward in time to the grid cell value at 12:00, with corrections for layer thicknesses and diffusion.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

P29G uses 12 standard input files, plus the P29G recursive output files PF168 - PF179. The 12 files generated by other processors are:

PF113 from P11G	MF109 from P07G
PF114 from P11G	MF112 from P08G
PF115 from P11G	MF113 from P08G
PF128 from P32G	MF133 from P09G
PF129 from P32G	MF134 from P09G
PF130 from P32G	MF135 from P09G

2. Nonstandard Input Files

None

12. Note that these standard output files have *non*standard usage, since the date and hour in the header record are fictitious.

B. Output Files

1. Standard Output Files

a. PF168

This file contains the gridded layer 1 wind components for the last hour of a simulation. PF168 is used by P29G, and its parameters are shown in Table 2.195.

TABLE 2.195. PF168 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	<u>1	$m \cdot s^{-1}$	Real*4	Layer 1 cell west-east wind component
2	<v>1	$m \cdot s^{-1}$	Real*4	Layer 1 cell south-north wind component

b. PF169

This file contains the gridded layer 2 wind components for the last hour of a simulation. PF169 is used by P29G, and its parameters are shown in Table 2.196.

TABLE 2.196. PF169 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	<u>2	$m \cdot s^{-1}$	Real*4	Layer 2 cell west-east wind component
2	<v>2	$m \cdot s^{-1}$	Real*4	Layer 2 cell south-north wind component

c. PF170

This file contains the gridded layer 3 wind components for the last hour of a simulation. PF170 is used by P29G, and its parameters are shown in Table 2.197.

TABLE 2.197. PF170 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	<u>3	$m \cdot s^{-1}$	Real*4	Layer 3 cell west-east wind component
2	<v>3	$m \cdot s^{-1}$	Real*4	Layer 3 cell south-north wind component

d. PF171

This file contains the gridded layer 1 cell thicknesses for the last hour of a simulation. PF171 is used by P29G, and its parameter is shown in Table 2.198.

TABLE 2.198. PF171 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	h1	m	Real*4	Layer 1 cell thicknesses

e. PF172

This file contains the gridded layer 2 cell thicknesses for the last hour of a simulation. PF172 is used by P29G, and its parameter is shown in Table 2.199.

TABLE 2.199. PF172 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	h2	m	Real*4	Layer 2 cell thicknesses

f. PF173

This file contains the gridded layer 3 cell thicknesses for the last hour of a simulation. PF173 is used by P29G, and its parameter is shown in Table 2.200.

TABLE 2.200. PF173 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	h3	m	Real*4	Layer 3 cell thicknesses

g. PF174

This file contains the gridded layer 1 horizontal eddy diffusivities for the last hour of a simulation. PF174 is used by P29G, and its parameter is shown in Table 2.201.

TABLE 2.201. PF174 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	K1	$m \cdot s^{-1}$	Real*4	Layer 1 cell horizontal eddy diffusivity

h. PF175

This file contains the gridded layer 2 horizontal eddy diffusivities for the last hour of a simulation. PF175 is used by P29G, and its parameter is shown in Table 2.202.

TABLE 2.202. PF175 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	K2	$m \cdot s^{-1}$	Real*4	Layer 2 cell horizontal eddy diffusivity

i. PF176

This file contains the gridded layer 3 horizontal eddy diffusivities for the last hour of a simulation. PF176 is used by P29G, and its parameter is shown in Table 2.203.

TABLE 2.203. PF176 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	K3	$m \cdot s^{-1}$	Real*4	Layer 3 cell horizontal eddy diffusivity

j. PF177

This file contains the gridded layer 1 air density correction factors (used in chemical rate constant calculations) for the last hour of a simulation. PF177 is used by P29G, and its parameter is shown in Table 2.204.

TABLE 2.204. PF177 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	<RHO>1	$mol \cdot m^{-1}$	Real*4	Layer 1 cell air density correction factors

k. PF178

This file contains the gridded layer 2 air density correction factors (used in chemical rate constant calculations) for the last hour of a simulation. PF178 is used by P29G, and its parameter is shown in Table 2.205.

TABLE 2.205. PF178 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	<RHO>2	$mol \cdot m^{-1}$	Real*4	Layer 2 cell air density correction factors

l. PF179

This file contains the gridded layer 3 air density correction factors (used in chemical rate constant calculations) for the last hour of a simulation. PF179 is used by P29G, and its parameter is shown in Table 2.206.

TABLE 2.206. PF179 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	<RHO>3	$mol \cdot m^{-1}$	Real*4	Layer 3 cell air density correction factors

m. MF103

This file contains the hourly gridded layer 1 horizontal eddy diffusivities in $(\lambda - \phi)$coordinates. MF103 is used by P38G, and its parameters are shown in Table 2.207.

TABLE 2.207. MF103 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$K^*x1_{i,j}$	$rad^2 \cdot s^{-1}$	Real*4	λ-component layer 1 horizontal eddy diffusivity
2	$K^*y1_{i,j}$	$rad^2 \cdot s^{-1}$	Real*4	ϕ-component layer 1 horizontal eddy diffusivity

Note: i = column number, j = row number.

n. MF104

This file contains the hourly gridded layer 2 horizontal eddy diffusivities in (λ–ϕ)coordinates. MF104 is used by P38G, and its parameters are shown in Table 2.208.

TABLE 2.208. MF104 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$K^*x2_{i,j}$	$rad^2 s^{-1}$	Real*4	λ-component layer 2 horizontal eddy diffusivity
2	$K^*y2_{i,j}$	$rad^2 s^{-1}$	Real*4	ϕ-component layer 2 horizontal eddy diffusivity

Note: i = column number, j = row number.

o. MF105

This file contains the hourly gridded layer 3 horizontal eddy diffusivities in (λ–ϕ)coordinates. MF105 is used by P38G, and its parameters are shown in Table 2.209.

TABLE 2.209. MF105 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$K^*x3_{i,j}$	$rad^2 s^{-1}$	Real*4	λ-component layer 3 horizontal eddy diffusivity
2	$K^*y3_{i,j}$	$rad^2 s^{-1}$	Real*4	ϕ-component layer 3 horizontal eddy diffusivity

Note: i = column number, j = row number.

p. MF106

This file contains hourly layer 1 backtrack positions, in (λ–ϕ) coordinates, of grid cell nodes. Each hour begins at grid cell nodes; P29G calculates the backtrack coordinates at (hour - 30 min); then, start at (hour - 30) and again at the grid cell nodes, P29G calculates the (hour - 00 min) position. MF106 is used by P38G, and its parameters are shown in Table 2.210.

TABLE 2.210. MF106 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$LAMBDAbt1\text{-}00_{i,j,h}$		Real*4	λ (column) layer 1 location at (hour - 00 min)
2	$PHIbt1\text{-}00_{i,j,h}$		Real*4	ϕ (row) layer 1 location at (hour - 00 min)
3	$LAMBDAbt1\text{-}30_{i,j,h}$		Real*4	λ (column) layer 1 location at (hour - 30 min)
4	$PHIbt1\text{-}30_{i,j,h}$		Real*4	ϕ (row) layer 1 location at (hour - 30 min)

Note: i = column node number, j = row node number, h = hour.

q. MF107

This file contains hourly layer 2 backtrack positions, in (λ–ϕ) coordinates, of grid cell nodes. Each hour begins at grid cell nodes; P29G calculates the backtrack coordinates at (hour - 30 min); then, start at (hour - 30) and again at the grid cell nodes, P29G calculates the (hour - 00 min) position. MF107 is used by P38G, and its parameters are shown in Table 2.211.

TABLE 2.211. MF107 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	LAMBDAbt2-00$_{i,j,h}$		Real*4	λ (column) layer 2 location at (hour - 00 min)
2	PHIbt2-00$_{i,j,h}$		Real*4	ϕ (row) layer 2 location at (hour - 00 min)
3	LAMBDAbt2-30$_{i,j,h}$		Real*4	λ (column) layer 2 location at (hour - 30 min)
4	PHIbt2-30$_{i,j,h}$		Real*4	ϕ (row) layer 2 location at (hour - 30 min)

Note: i = column node number, j = row node number, h = hour.

r. MF108

This file contains hourly layer 3 backtrack positions, in (λ–ϕ) coordinates, of grid cell nodes. Each hour begins at grid cell nodes; P29G calculates the backtrack coordinates at (hour - 30 min); then, start at (hour - 30) and again at the grid cell nodes, P29G calculates the (hour - 00 min) position. MF108 is used by P38G, and its parameters are shown in Table 2.212.

TABLE 2.212. MF108 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	LAMBDAbt3-00$_{i,j,h}$		Real*4	λ (column) layer 3 location at (hour - 00 min)
2	PHIbt3-00$_{i,j,h}$		Real*4	ϕ (row) layer 3 location at (hour - 00 min)
3	LAMBDAbt3-30$_{i,j,h}$		Real*4	λ (column) layer 3 location at (hour - 30 min)
4	PHIbt3-30$_{i,j,h}$		Real*4	ϕ (row) layer 3 location at (hour - 30 min)

Note: i = column node number, j = row node number, h = hour.

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 2.213 defines the control card variables.

STDATE STHOUR OUTHRS INHRS SCDATE PRHOUR NSUBS

TABLE 2.213. CONTROL CARD VARIABLES

Variable Name	Unit	Description
STDATE		Julian scenario start date as *YYDDD*
STHOUR	EST	Scenario start time as *HH*
OUTHRS	h	Number of hours to process
INHRS	h	Number of hours required for input [a]
SCDATE		Julian episode start date as *YYDDD*
PRHOUR	EST	Last hour of previous scenario, used as input to current scenario, as *HH* [b]
NSUBSS		Number of subintervals per time step for back trajectory computation

a. Used by P29G to check the value of OUTHRS.

b. PRHOUR is indexed start with *HH* = 12 for the first day of an episode; subsequent records are written as though they are one hour apart (refer to the examples below).

Examples (control cards by scenario through an episode):

```
88194 12 72 72 88194 12 3        [first scenario]
88197 12 72 72 88194 13 3        [second scenario]
88200 12 72 72 88194 14 3        [third scenario]
          :
```

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	16	files	138	blocks
FORTRAN INCLUDE files:	8	files	51	blocks
Object files:	16	files	87	blocks
Executable file:	1	file	102	blocks
	41	files	378	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:50:44
Buffered I/O count:	517
Direct I/O count:	10106
Peak working-set size:	3583
Peak virtual size:	6127

C. Space Requirements: Log and Print Files

P29GX1.LOG:	69 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.214 shows the input file and output file space requirements.

TABLE 2.214. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF113	5	3798	72 hours
	PF114	5	3798	72 hours
	PF115	5	3798	72 hours
	PF128	5	1899	72 hours
	PF129	5	1899	72 hours
	PF130	5	1899	72 hours
	MF109	5	1899	72 hours
	MF112	5	1899	72 hours
	MF113	5	1899	72 hours
	MF133	5	1899	72 hours
	MF134	5	1899	72 hours
	MF135	5	1898	72 hours
			28485	
Output	PF168	5	3806	1 hours
	PF169	5	3806	1 hours
	PF170	5	3806	1 hours
	PF171	5	1903	1 hours
	PF172	5	1903	1 hours
	PF173	5	1903	1 hours
	PF174	5	1903	1 hours
	PF175	5	1903	1 hours
	PF176	5	1903	1 hours
	PF177	5	1903	1 hours
	PF178	5	1903	1 hours
	PF179	5	1903	1 hours
	MF103	5	3798	72 hours
	MF104	5	3798	72 hours
	MF105	5	3798	72 hours
	MF106	5	5696	72 hours
	MF107	5	5696	72 hours
	MF108	5	5696	72 hours
			62718	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P29GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE $2$DUA24:[ROMCOM.ROMNET]P29GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P29GV1A.MAP"
$ EXEFILE = "$2$DUA24:[ROMCOM.ROMNET]P29GV1A.EXE"
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]P29G.OBJ               A1
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]AREAC.OBJ              A2
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]BQ3LAM.OBJ             A3
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]BQ3PHI.OBJ             A4
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CONDIF.OBJ             A5
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]CONWIN.OBJ             A6
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]DATHR.OBJ              A7
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]DAY.OBJ                A8
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]DIFLAM.OBJ             A9
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]DIFPHI.OBJ             A10
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]DLNLAM.OBJ             A11
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]DLNPHI.OBJ             A12
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]LWRHVO.OBJ             A13
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]POLY.OBJ               A14
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]RDCONV.OBJ             A15
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.ROMNET]VOLCAL.OBJ             A16
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]ADATE.OBJ              A17
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELL0.OBJ              A18
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]CELL2.OBJ              A19
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]DATHR3.OBJ             A20
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FLSMRY.OBJ             A21
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]FSTAT1.OBJ             A22
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IBLKR.OBJ              A23
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]IOCL.OBJ               A24
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JFILE1.OBJ             A25
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JHOUR1.OBJ             A26
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]JUNIT.OBJ              A27
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPDIR2.OBJ             A28
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]OPEN2.OBJ              A29
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]PROGID.OBJ             A30
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDDIR1.OBJ             A31
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFHD1.OBJ             A32
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDFT5.OBJ              A33
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]RDREC.OBJ              A34
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRFT5.OBJ              A35
$ ASSIGN/USER_MODE $2$DUA24:[ROMOBJ.UTILIO]WRREC.OBJ              A36
$ LINK -
  /EXECUTABLE = $2$DUA24:[ROMCOM.ROMNET]P29GV1A.EXE -
  /MAP = MET1:[S01S]P29GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A34, -
  A35, -
  A36
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P29GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P29GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P29GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                 PFMFDIR
$ ASSIGN MET56:[ROMNET1.D85188]PF113A.DAT         PF113
$ ASSIGN MET56:[ROMNET1.D85188]PF114A.DAT         PF114
$ ASSIGN MET56:[ROMNET1.D85188]PF115A.DAT         PF115
$ ASSIGN MET56:[ROMNET1.D85188]PF128A.DAT         PF128
$ ASSIGN MET56:[ROMNET1.D85188]PF129A.DAT         PF129
$ ASSIGN MET56:[ROMNET1.D85188]PF130A.DAT         PF130
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT         MF109
$ ASSIGN MET56:[ROMNET1.D85188]MF112A.DAT         MF112
$ ASSIGN MET56:[ROMNET1.D85188]MF113A.DAT         MF113
$ ASSIGN MET56:[ROMNET1.D85188]MF133A.DAT         MF133
$ ASSIGN MET56:[ROMNET1.D85188]MF134A.DAT         MF134
$ ASSIGN MET56:[ROMNET1.D85188]MF135A.DAT         MF135
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET58:[ROMNET1.D85188]MF103A.DAT         MF103
$ ASSIGN MET58:[ROMNET1.D85188]MF104A.DAT         MF104
$ ASSIGN MET58:[ROMNET1.D85188]MF105A.DAT         MF105
$ ASSIGN MET58:[ROMNET1.D85188]MF106A.DAT         MF106
$ ASSIGN MET58:[ROMNET1.D85188]MF107A.DAT         MF107
$ ASSIGN MET58:[ROMNET1.D85188]MF108A.DAT         MF108
$ ASSIGN MET1:[ROMNET1.D85188]PF168A.DAT          PF168
$ ASSIGN MET1:[ROMNET1.D85188]PF169A.DAT          PF169
$ ASSIGN MET1:[ROMNET1.D85188]PF170A.DAT          PF170
$ ASSIGN MET1:[ROMNET1.D85188]PF171A.DAT          PF171
$ ASSIGN MET1:[ROMNET1.D85188]PF172A.DAT          PF172
$ ASSIGN MET1:[ROMNET1.D85188]PF173A.DAT          PF173
$ ASSIGN MET1:[ROMNET1.D85188]PF174A.DAT          PF174
$ ASSIGN MET1:[ROMNET1.D85188]PF175A.DAT          PF175
$ ASSIGN MET1:[ROMNET1.D85188]PF176A.DAT          PF176
$ ASSIGN MET1:[ROMNET1.D85188]PF177A.DAT          PF177
$ ASSIGN MET1:[ROMNET1.D85188]PF178A.DAT          PF178
$ ASSIGN MET1:[ROMNET1.D85188]PF179A.DAT          PF179
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P29GX1.CPR
$ !
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P29GX1.CPR   SYS$INPUT
$ !
$ RUN $2$DUA24:[ROMEXE.ROMNET]P29GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A      P29GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P29GV1A

B. *Subroutines*

AREAC	CONWIN	DIFPHI	POLY
BQ3LAM	DATHR	DLNLAM	RDCONV
BQ3PHI	DAY	DLNPHI	VOLCAL
CONDIF	DIFLAM	LNRHVO	

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	FSTAT1	OPEN2	RDFT5
CELL0	IBLKR	PROGID	RDREC
CELL2	JFILE1	RDDIR1	WRFT5
FLSMRY	OPDIR2	RDFHD1	WRREC

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

DATHR3
IOCL
JHOUR1
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

ABGUTS.EXT
CNTRL.EXT
DAYHR.EXT
FILES.EXT
P29PRM.EXT
SOLPRM.EXT
ZGUTS.EXT

2.27 PROCESSOR P31G (EMISSIONS)

I. PROCESSOR FUNCTION

This stage 0 processor extracts the hourly point-source emissions for three representative day types (Saturdays, Sundays, and weekdays) from the SEANPT.CS0 data file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

SEANPT.CS0 contains the annual point-source emissions data plus seasonal temporal allocation factors; it is generated by PTEXTR. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 2.215.

```
      READ(UNITTR, 1003, IOSTAT = IOSTR)
     &     STKID, SCC,
     &     NSPCIN, (CLASS(ISPC), SEANEM(ISPC), ISPC = 1, NSPCIN),
     &     MXDYIN, (S_AFAC(IDAY), D_SFAC(IDAY),
     &             (H_DFAC(IHR, IDAY), IHR = 1, 24), IDAY = 1, MXDYIN)

 1003 FORMAT(A12, A8,
     &  I3, <NSPECS>(A5, E12.0),
     &  I2, <MXDAY>(F5.0, F6.0, 24F6.0))
```

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

PNTWK.CS0, PNTSA.CS0, and PNTSU.CS0

P31G generates three binary output files named PNTWK.CS0 (weekdays), PNTSA.CS0 (Saturdays), and PNTSU.CS0 (Sundays), that contain the seasonal hourly concentrations of the 15 chemical species shown in Table 2.216. The output files contain the point sources sorted by their AEROS state/county/plant/point identification codes. The WRITE statement for these files is listed below.

```
      WRITE(IUNIT) STKSR, HOURSR, EMISSR
```

TABLE 2.215. SEANPT.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	STKID		Char*12	Stack ID
2	SCC		Char*8	Source classification code
3	NSPCIN		Integer*4	Number of species
4	CLASS(1)		Char*5	AEROS code for VOC
5	$SEANEM(1)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, VOC
6	CLASS(2)		Char*5	AEROS code for CO
7	$SEANEM(2)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, CO
8	CLASS(3)		Char*5	AEROS code for olefins
9	$SEANEM(3)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, olefins
10	CLASS(4)		Char*5	AEROS code for paraffins
11	$SEANEM(4)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, paraffins
12	CLASS(5)		Char*5	AEROS code for toluene
13	$SEANEM(5)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, toluene
14	CLASS(6)		Char*5	AEROS code for xylene
15	$SEANEM(6)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, xylene
16	CLASS(7)		Char*5	AEROS code for formaldehyde
17	$SEANEM(7)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, formald.
18	CLASS(8)		Char*5	AEROS code for aldehydes
19	$SEANEM(8)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, aldehydes
20	CLASS(9)		Char*5	AEROS code for ethene
21	$SEANEM(9)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, ethene
22	CLASS(10)		Char*5	AEROS code for nonr.
23	$SEANEM(10)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, nonr.
24	CLASS(11)		Char*5	AEROS code for isoprenes
25	$SEANEM(11)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, isoprenes
26	CLASS(12)		Char*5	AEROS code for methane
27	$SEANEM(12)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, methane
28	CLASS(13)		Char*5	AEROS code for NO
29	$SEANEM(13)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, NO
30	CLASS(14)		Char*5	AEROS code for NO_2
31	$SEANEM(14)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, NO_2
32	CLASS(15)		Char*5	AEROS code for methanol
33	$SEANEM(15)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, methanol
34	MXDAYIN		Integer*4	No. of day types per season
35	S_AFAC_p		Real*4	Seasonal allocation factor
36	D_SFAC_p		Real*4	Weekday allocation factor
37	$H\_DFAC(1)_p$		Real*4	Hour 1 allocation factor
⋮	⋮		⋮	⋮
61	$H\_DFAC(24)_p$		Real*4	Hour 24 allocation factor
62	S_AFAC_p		Real*4	Seasonal allocation factor
63	D_SFAC_p		Real*4	Saturday allocation factor
64	$H\_DFAC(1)_p$		Real*4	Hour 1 allocation factor
⋮	⋮		⋮	⋮
88	$H\_DFAC(24)_p$		Real*4	Hour 24 allocation factor
89	S_AFAC_p		Real*4	Seasonal allocation factor
90	D_SFAC_p		Real*4	Sunday allocation factor
91	$H\_DFAC(1)_p$		Real*4	Hour 1 allocation factor
⋮	⋮		⋮	⋮
115	$H\_DFAC(24)_p$		Real*4	Hour 24 allocation factor

Note: p = point source; nonr. = nonreactive hydrocarbons; formald. = formaldehyde. Units are English short tons. NO emissions are reported as weight of NO_2.

TABLE 2.216. PNTWK.CS0, PNTSA.CS0, AND PNTSU.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	STKSR		Char*12	AEROS stack ID
2	HOURSR	LST	Integer*4	Hour as *HH*
3	$EMISSR(1)_p$	$tons \cdot h^{-1}$	Real*4	Annual emission rate, VOC
4	$EMISSR(2)_p$	$tons \cdot h^{-1}$	Real*4	Annual emission rate, CO
5	$EMISSR(3)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, olefins
6	$EMISSR(4)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, paraffins
7	$EMISSR(5)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, toluene
8	$EMISSR(6)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, xylene
9	$EMISSR(7)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, formald.
10	$EMISSR(8)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, aldehydes
11	$EMISSR(9)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, ethene
12	$EMISSR(10)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, nonr.
13	$EMISSR(11)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, isoprenes
14	$EMISSR(12)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, methane
15	$EMISSR(13)_p$	$tons \cdot h^{-1}$	Real*4	Annual emission rate, NO
16	$EMISSR(14)_p$	$tons \cdot h^{-1}$	Real*4	Annual emission rate, NO_2
17	$EMISSR(15)_p$	$mol \cdot h^{-1}$	Real*4	Annual emission rate, methanol

Note: p = point source; nonr. = nonreactive hydrocarbons; formald. = formaldehyde.

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	3	files	30	blocks
FORTRAN INCLUDE files:	3	files	9	blocks
Object files:	3	files	15	blocks
Executable file:	1	file	27	blocks
	10	files	81	blocks

B. Execution Time Requirements (Representative Values for One Season's Extractions):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:44:18
Buffered I/O count:	2416
Direct I/O count:	19686
Peak working-set size:	6751
Peak virtual size:	8663

C. Space Requirements: Log and Print Files

PTEXTR.LOG:	6 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.217 shows the input file and output file space requirements.

TABLE 2.217. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SEANPT.CS0	Nonstd	26204	seasonal
Output	PNTSA.CS0	Nonstd	57960	daily
	PNTSU.CS0	Nonstd	57960	daily
	PNTWK.CS0	Nonstd	57960	daily
			173880	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P31GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P31GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P31GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P31GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]P31G.OBJ          A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]RDSAPT.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]SRTREC.OBJ        A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ         A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GETMSG.OBJ        A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]INDEX1.OBJ        A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ          A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ        A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE4.OBJ        A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ         A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ        A11
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P31GV1A.EXE -
  /MAP = MET1:[S01S]P31GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A9, -
  A10, -
```

```
  A11
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P31GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P31GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1503A]P31GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! Assign work files for sorting - currently about 100,000 blocks each:
$ ASSIGN MET20: SORTWORK0
$ ASSIGN MET27: SORTWORK1
$ ! ASSIGN INPUT FILES--BASE CASE (NO CONTROL STRATEGY)
$ ASSIGN MET53:[EMISSIONS.RNSUM05P]SEANPT.CSO      SEANLPT
$ ! ASSIGN OUTPUT FILES
$ ASSIGN WORK_SCRATCH:PNTWKZ.CSO                   PNTWKZ
$ ASSIGN WORK_SCRATCH:PNTSAZ.CSO                   PNTSAZ
$ ASSIGN WORK_SCRATCH:PNTSUZ.CSO                   PNTSUZ
$ !
$ SHOW PROCESS / ACCOUNTING
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P31GV1A.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P31GV1A

B. Subroutines

RDSAPT
SRTREC

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE
GETMSG
PROGID

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

INDEX1
IOCL
JFILE2
JFILE4
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

HRFILE.EXT
SPCLIS.EXT
SRFILE.EXT

2.28 PROCESSOR P32G (METEOROLOGY)

I. PROCESSOR FUNCTION

This stage 5 processor calculates (1) the horizontal eddy diffusivities for layers 1, 2, and 3, and (2) the vertical turbulence for layers 0 and 1. The horizontal eddy diffusivities are used to parameterize the dispersion of pollutants within a layer. Vertical turbulence values are used in the calculation of volume fluxes through the layer interfaces.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

P32G uses 10 standard input files:

PF106 from P03G	PF156 from P08G
PF107 from P04G	PF157 from P05G
PF109 from P04G	MF109 from P07G
PF110 from P04G	MF112 from P08G
PF132 from P07G	MF113 from P08G

2. Nonstandard Input Files

None

B. Output Files

1. Standard Output Files

a. PF128

This file contains the hourly gridded horizontal eddy diffusivity for layer 1. PF128 is used by P29G, and its parameter is shown in Table 2.218.

TABLE 2.218. PF128 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	K1	$m^2 \cdot s^{-1}$	Real*4	Layer 1 horizontal eddy diffusivity

b. PF129

This file contains the hourly gridded horizontal eddy diffusivity for layer 2. PF129 is used by P29G, and its parameter is shown in Table 2.219.

TABLE 2.219. PF129 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	K2	$m^2 \cdot s^{-1}$	Real*4	Layer 2 horizontal eddy diffusivity

c. PF130

This file contains the hourly gridded horizontal eddy diffusivity for layer 3. PF130 is used by P29G, and its parameter is shown in Table 2.220.

TABLE 2.220. PF130 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	K3	$m^2 \cdot s^{-1}$	Real*4	Layer 3 horizontal eddy diffusivity

d. PF160

This file contains the hourly gridded vertical turbulence in layer 0. PF160 is used by P12G, and its parameter is shown in Table 2.221.

TABLE 2.221. PF160 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SIGMAw0	$m \cdot s^{-1}$	Real*4	Layer 0 vertical turbulence

e. PF161

This file contains the hourly gridded vertical turbulence in layer 1. PF161 is used by P12G, and its parameter is shown in Table 2.222.

TABLE 2.222. PF161 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SIGMAw1	$m \cdot s^{-1}$	Real*4	Layer 1 vertical turbulence

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 2.223 defines the control card variables.

IDATE IHRBGN NUMHRS

TABLE 2.223. CONTROL CARD VARIABLES

Variable Name	Unit	Description
IDATE		Julian scenario start date as *YYDDD*
IHRBGN	EST	Scenario start time as *HH*
NUMHRS	h	Number of hours to process

Example:

80194 12 72

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	31	blocks
FORTRAN INCLUDE files:	3	files	29	blocks
Object files:	2	files	14	blocks
Executable file:	1	file	76	blocks
	8	files	150	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:09:33
Buffered I/O count:	315
Direct I/O count:	5895
Peak working-set size:	1716
Peak virtual size:	4411

C. Space Requirements: Log and Print Files

P32GX1.LOG:	32 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.224 shows the input file and output file space requirements.

E. Space Requirements: Tape Files

None

TABLE 2.224. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF106	3	2140	72 hours
	PF107	5	1898	72 hours
	PF109	5	1898	72 hours
	PF110	5	1898	72 hours
	PF132	5	1898	72 hours
	PF156	5	1898	72 hours
	PF157	5	1898	72 hours
	MF109	5	1898	72 hours
	MF112	5	1898	72 hours
	MF113	5	1898	72 hours
			19222	
Output	PF128	5	1898	72 hours
	PF129	5	1898	72 hours
	PF130	5	1898	72 hours
	PF160	5	1898	72 hours
	PF161	5	1898	72 hours
			9490	

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P32GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P32GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P32GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMCOM.ROMNET]P32GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P32G.OBJ              A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]DIFTRB.OBJ            A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ             A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]BGRID1.OBJ.           A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ             A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ            A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ            A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]GETMEM.OBJ            A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ             A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ              A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ            A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ            A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ             A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]NEXTHR.OBJ            A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ            A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ             A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ            A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ            A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD1.OBJ            A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD2.OBJ            A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT3.OBJ             A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ             A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ             A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT5.OBJ             A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ             A25
```

```
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMCOM.ROMNET]P32GV1A.EXE -
  /MAP = MET1:[S01S]P32GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A23, -
  A24, -
  A25
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P32GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P32GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1501A]P32GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                 PFMFDIR
$ ASSIGN MET56:[ROMNET1.D85188]PF106A.DAT         PF106
$ ASSIGN MET56:[ROMNET1.D85188]PF107A.DAT         PF107
$ ASSIGN MET56:[ROMNET1.D85188]PF109A.DAT         PF109
$ ASSIGN MET56:[ROMNET1.D85188]PF110A.DAT         PF110
$ ASSIGN MET56:[ROMNET1.D85188]PF132A.DAT         PF132
$ ASSIGN MET56:[ROMNET1.D85188]PF156A.DAT         PF156
$ ASSIGN MET56:[ROMNET1.D85188]PF157A.DAT         PF157
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT         MF109
$ ASSIGN MET56:[ROMNET1.D85188]MF112A.DAT         MF112
$ ASSIGN MET56:[ROMNET1.D85188]MF113A.DAT         MF113
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET56:[ROMNET1.D85188]PF128A.DAT         PF128
$ ASSIGN MET56:[ROMNET1.D85188]PF129A.DAT         PF129
$ ASSIGN MET56:[ROMNET1.D85188]PF130A.DAT         PF130
$ ASSIGN MET56:[ROMNET1.D85188]PF160A.DAT         PF160
$ ASSIGN MET56:[ROMNET1.D85188]PF161A.DAT         PF161
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1501A]P32GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ !
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1501A]P32GX1.CPR   SYS$INPUT
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P32GV1A
$ SHOW PROCESS / ACCOUNTING
$ @USER2$DISK:[FOREMAN.COM1]MESSAGE FV1501A       P32GX1
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P32GV1A

B. Subroutines

DIFTRB

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	IBLKR	PROGID	RDFT5
BGRID1	JFILE1	RDDIR1	RDREC
CELL0	NEXTHR	RDFHD1	WRFT5
FLSMRY	OPDIR2	RDFHD2	WRREC
FSTAT1	OPEN2	RDFT3	

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETMEM
IOCL
JHOUR1
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

METDAT.EXT
PARMS.EXT

2.29 PROCESSOR P33G (EMISSIONS)

I. PROCESSOR FUNCTION

This stage 6 processor generates the locations and strength of a hypothetical tracer species that might emanate from a constant-source emitter. The tracer's grid cell locations and strength is defined by the set of locations and strengths of the grid cells with the top 50 concentrations of NO_x and VOC.[13] The processor searches for the highest NO_x value from among these 50 cells, and assigns this value to each of the 50 cells. Note that the output file from P33G is used only in the calculation of pollutant transport; the actual numeric value of the parameter has no significance.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

P33G uses 13 standard input files:

PF106 from P03G	MF110 from P07G
PF107 from P04G	MF111 from P07G
PF115 from P11G	MF112 from P08G
PF119 from P03G	MF132 from P09G
PF132 from P07G	MF133 from P09G
PF145 from P14G	MF134 from P09G
MF109 from P07G	

2. Nonstandard Input Files

P33G uses one binary output file, PTE3.DAT, that contains the major point-source hourly emissions data. The READ statement for this file is listed below, and its parameters are shown in Table 2.225.

```
      READ (IUNIT,IOSTAT=IEOF) STKID,PSLON,PSLAT,EXFL,PLMT
     &     DK,ZSK,TK,WK,(EK(L),L=1,NP)
```

13. Within the ROMNET domain, we have found these locations to be predominantly cities.

TABLE 2.225. PTE3.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$STKID_p$		Char*12	AEROS state/county/plant/point ID
2	$PSLON_p$	° W	Real*4	Longitude (fractional)
3	$PSLAT_p$	° N	Real*4	Latitude (fractional)
4	$EXFL_p$	$m^3 \cdot s^{-1}$	Real*4	Exhaust flow rate
5	$PLMT_p$	m	Real*4	Plume height
6	DK_p	m	Real*4	Stack diameter
7	ZSK_p	m	Real*4	Stack height
8	TK_p	°C	Real*4	Exhaust gas temperature
9	WK_p	$m \cdot s^{-1}$	Real*4	Exhaust gas exit velocity
10	$EK(1)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for aldehydes
11	$EK(2)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for CO
12	$EK(3)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for ethene
13	$EK(4)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for formaldehyde
14	$EK(5)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for isoprenes
15	$EK(6)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for NO
16	$EK(7)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for NO_2
17	$EK(8)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for olefins
18	$EK(9)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for paraffins
19	$EK(10)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for toluene
20	$EK(11)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for xylene
21	$EK(12)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for nonr.
22	$EK(13)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methane
23	$EK(14)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methanol
24	$EK(15)_{p,h}$	$kg \cdot h^{-1}$	Real*4	Emission rate for VOC
25	$EK(16)_{p,h}$		Real*4	$NMHC/NO_x$ ratio

Note: p = point source, h = hour; nonr. = nonreactive hydrocarbons, VOC = volatile organic compounds, NMHC = nonmethane hydrocarbons.

B. Output Files

1. Standard Output Files

P33G generates one output file, PF155, that contains the concentration of the hypothetical tracer species that is placed in the grid cells that originally had the top 50 concentrations of NO_x and VOC. PF155 is used by P10G, and its parameter is shown in Table 2.226.

TABLE 2.226. PF155 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$TRCEMIS_{i,j}$	$ppm \cdot m \cdot s^{-1}$	Real*4	Emissions concentration for tracer species (50 cells only)

Note: i = column number, j = row number.

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 2.227 defines the control card variables.

JDATE, JHOUR

TABLE 2.227. CONTROL CARD VARIABLES

Variable Name	Unit	Description
JDATE		Julian scenario start date as *YYDDD*
JHOUR	EST	Scenario start time as *HH*

Example:

88184,12

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	6	files	69	blocks
FORTRAN INCLUDE files:	6	files	36	blocks
Object files:	6	files	36	blocks
Executable file:	1	file	75	blocks
	19	files	216	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:03:35
Buffered I/O count:	269
Direct I/O count:	1202
Peak working-set size:	3465
Peak virtual size:	6850

C. Space Requirements: Log and Print Files

P33G.LOG:	87 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.228 shows the input file and output file space requirements.

TABLE 2.228. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF106	3	2142	72 hours
	PF107	5	1899	72 hours
	PF115	5	3798	72 hours
	PF119	4	55	
	PF132	5	1899	72 hours
	PF145	5	30369	72 hours
	MF109	5	1899	72 hours
	MF110	5	1899	72 hours
	MF111	5	1899	72 hours
	MF112	5	1899	72 hours
	MF132	5	1899	72 hours
	MF133	5	1899	72 hours
	MF134	5	1899	72 hours
	PTE3.DAT	Nonstd	13833	72 hours
			67288	
Output	PF155	4	55	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P33GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P33GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P33GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P33GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]P33G.OBJ              A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]RDSTAK.OBJ            A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]SATEMP.OBJ            A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]SETCLK.OBJ            A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]SORTAI.OBJ            A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]SORTAR.OBJ            A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ             A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL0.OBJ             A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL1.OBJ             A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELL2.OBJ             A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLAR.OBJ            A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ            A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ            A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ             A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ              A15
```

```
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ          A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ          A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE3.OBJ          A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ          A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ           A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ          A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ           A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ          A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ          A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT3.OBJ           A25
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT4.OBJ           A26
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ           A27
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ           A28
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFT4.OBJ           A29
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRREC.OBJ           A30
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P33GV1A.EXE -
  /MAP = MET1:[S01S]P33GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A28, -
  A29, -
  A30
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P33GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P33GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1503A]P33GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                 PFMFDIR
$ ASSIGN MET58:[ROMNET1.D85188]PTE3A.DAT          PTE3
$ ASSIGN MET56:[ROMNET1.D85188]PF106A.DAT         PF106
$ ASSIGN MET56:[ROMNET1.D85188]PF107A.DAT         PF107
$ ASSIGN MET56:[ROMNET1.D85188]PF115A.DAT         PF115
$ ASSIGN MET1:[ROMNET1.D00000]PF119A.DAT          PF119
$ ASSIGN MET56:[ROMNET1.D85188]PF132A.DAT         PF132
$ ASSIGN MET58:[ROMNET1.D85188]PF145A.DAT         PF145
$ ASSIGN MET56:[ROMNET1.D85188]MF109A.DAT         MF109
$ ASSIGN MET56:[ROMNET1.D85188]MF110A.DAT         MF110
$ ASSIGN MET56:[ROMNET1.D85188]MF111A.DAT         MF111
$ ASSIGN MET56:[ROMNET1.D85188]MF112A.DAT         MF112
$ ASSIGN MET56:[ROMNET1.D85188]MF132A.DAT         MF132
$ ASSIGN MET56:[ROMNET1.D85188]MF133A.DAT         MF133
$ ASSIGN MET56:[ROMNET1.D85188]MF134A.DAT         MF134
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET1:[ROMNET1.D00000]PF155A.DAT          PF155
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1503A]P33GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.1503A]P33GX1.CPR   SYS$INPUT
$ !
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P33GV1A.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P33GV1A

B. Subroutines

RDSTAK	SORTAI
SATEMP	SORTAR
SETCLK	

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	CELLAR	JFILE1	RDDIR1	RDFT5
CELL0	FLSMRY	OPDIR2	RDFT3	WRFT4
CELL1	FSTAT1	OPEN2	RDFT4	WRREC
CELL2	IBLKR	PROGID		

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL	JHOUR1
JFILE2	JUNIT
JFILE3	

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

EMISP.EXT	STKDAT.EXT
FILEID.EXT	U1.EXT
METPRM.EXT	

2.30 PROCESSOR P34G (EMISSIONS)

I. PROCESSOR FUNCTION

Time information for the emissions data received under the National Acid Precipitation Assessment Program (NAPAP) inventory is given as Greenwich Mean Time (GMT). However, the ROM requires information based on local standard time (LST). This stage 0 processor performs the required time shift from GMT to LST on the point-, area-, and mobile-source data sets. Each of these source types has an associated day type (Saturdays, Sundays, and weekdays) that are fully independent from each other. Thus, for example data that occur at 0300 GMT on Sunday are not transposed to 2200 EST on Saturday; instead, they are assigned a "wrap-around" time within the "Sunday" file of 2200 EST. Note that we have designated the LST for the ROMNET domain to be EST, although the western portion of the domain falls within the Central time zone.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

P34G uses up to nine nonstandard input files; there can be three day-specific files for each of the point-, area-, and mobile-source emissions data. The second control card defines how many input files will be read into the processor.

a. AREASAZ.CS0, AREASUZ.CS0, and AREAWKZ.CS0

These are the day-specific area-source emissions data files, generated by preprocessor ARMERG, for Saturdays, Sundays, and weekdays respectively. The READ and FORMAT statements for these files are listed below, and their parameters are shown in Table 2.229.

```
      READ(IUNIT1,90,IOSTAT=IOST)COL, ROW, HOUR,(STEA(I),I=1,17)
90    FORMAT(2I4,I2,17E10.3)
```

TABLE 2.229. AREASAZ.CS0, AREASUZ.CS0, AND AREAWKZ.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Integer*4	Column number
2	ROW		Integer*4	Row number
3	HOUR	LST	Integer*4	Hour as *HH*
4	$STEA(1)_{i,j,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for NO_x
5	$STEA(2)_{i,j,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for VOC
6	$STEA(3)_{i,j,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for THC
7	$STEA(4)_{i,j,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for CO
8	$STEA(5)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for olefins
9	$STEA(6)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for paraffins
10	$STEA(7)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for toluene
11	$STEA(8)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for xylene
12	$STEA(9)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for formaldehyde
13	$STEA(10)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for aldehydes
14	$STEA(11)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for ethene
15	$STEA(12)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for isoprenes
16	$STEA(13)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for nonr.
17	$STEA(14)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for methane
18	$STEA(15)_{i,j,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for NO
19	$STEA(16)_{i,j,h}$	$tons{\cdot}h^{-1}$	Real*4	Emission rate for NO_2
20	$STEA(17)_{i,j,h}$	$mol{\cdot}h^{-1}$	Real*4	Emission rate for methanol

Note: i = column number; j = row number; h = hour; VOC = volatile organic compounds; THC = total hydrocarbons; nonr. = nonreactive hydrocarbons.

b. PNTSAZ.CS0, PNTSUZ.CS0, and PNTWKZ.CS0

These are the day-specific point-source emissions binary data files, generated by P31G, for Saturdays, Sundays, and weekdays respectively. The READ statement for these files are listed below, and their parameters are shown in Table 2.230.

```
READ(IUNIT2,IOSTAT=IOST2)STACK, PTIME, (SEK(I),I=1,15)
```

c. MOBLSAZ.CS0, MOBLSUZ.CS0, and MOBLWKZ.CS0

These are the day-specific mobile-source emissions data files, generated by preprocessor MBEXTR, for Saturdays, Sundays, and weekdays respectively. The READ and FORMAT statements for these files are listed below, and their parameters are shown in Table 2.231.

```
      READ(IUNIT3,90,IOSTAT=IOST3) COL, ROW, HOUR,(STEM(I),I=1,8),
     &                             STATE, CNTY
 90   FORMAT(2I4,I2,8E10.3,I2,I4)
```

TABLE 2.230. PNTSAZ.CS0, PNTSUZ.CS0, AND PNTWKZ.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	STACK		Char*12	AEROS state/county/plant/point ID
2	PTIME	LST	Integer*4	Hour as *HH*
3	$SEK(1)_{p,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for VOC
4	$SEK(2)_{p,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for CO
5	$SEK(3)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for olefins
6	$SEK(4)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for paraffins
7	$SEK(5)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for toluene
8	$SEK(6)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for xylene
9	$SEK(7)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for formaldehyde
10	$SEK(8)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for aldehydes
11	$SEK(9)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for ethene
12	$SEK(10)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for nonr.
13	$SEK(11)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for isoprenes
14	$SEK(12)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methane
15	$SEK(13)_{p,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO
16	$SEK(14)_{p,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO_2
17	$SEK(15)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methanol

Note: p = point source; h = hour; VOC = volatile organic compounds; nonr. = nonreactive hydrocarbons.

TABLE 2.231. MOBLSAZ.CS0, MOBLSUZ.CS0, AND MOBLWKZ.CS0 PARAMETERS

Parm No.	Parm Name	Unit [a]	Data Type	Description
1	COL		Integer*4	Column number
2	ROW		Integer*4	Row number
3	HOUR	LST	Integer*4	Hour as *HH*
4	STEM(1)	$tons \cdot h^{-1}$	Real*4	Mobile CO emissions
5	STEM(2)	$tons \cdot h^{-1}$	Real*4	Mobile NO emissions
6	STEM(3)	$tons \cdot h^{-1}$	Real*4	Mobile NO_2 emissions
7	STEM(4)	$tons \cdot h^{-1}$	Real*4	Mobile evap. emissions
8	STEM(5)	$tons \cdot h^{-1}$	Real*4	Mobile exhaust emissions
9	STEM(6)	$tons \cdot h^{-1}$	Real*4	Mobile diesel emissions
10	STEM(7)	$tons \cdot h^{-1}$	Real*4	Mobile VOC emissions
11	STEM(8)	$tons \cdot h^{-1}$	Real*4	Mobile THC emissions
12	STATE		Integer*4	State AEROS code
13	CNTY		Integer*4	County AEROS code

a. Units are in English short $tons \cdot h^{-1}$; 1 ton = 907.20 kg.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

P34G generates up to nine nonstandard input files; there can be three day-specific files for each of the point-, area-, and mobile-source emissions data. The second control card defines how many files will be generated by the processor.

a. AREASA.CS0, AREASU.CS0, and AREAWK.CS0

These are the area-source emissions data files, with data registered in LST, for Saturdays, Sundays, and weekdays respectively. The WRITE and FORMAT statements for these files are listed below, and their parameters are shown in Table 2.232. The files are used by P14G.

```
      WRITE(OUNIT(A),150) COL, ROW, HOUR, (STEA(I),I=1,17)
150   FORMAT(2I4,I2,17E10.3)
```

TABLE 2.232. AREASA.CS0, AREASU.CS0, AND AREAWK.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Integer*4	Column number
2	ROW		Integer*4	Row number
3	HOUR	LST	Integer*4	Hour as *HH*
4	$STEA(1)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO_x
5	$STEA(2)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for VOC
6	$STEA(3)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for THC
7	$STEA(4)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for CO
8	$STEA(5)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for olefins
9	$STEA(6)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for paraffins
10	$STEA(7)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for toluene
11	$STEA(8)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for xylene
12	$STEA(9)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for formaldehyde
13	$STEA(10)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for aldehydes
14	$STEA(11)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for ethene
15	$STEA(12)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for isoprenes
16	$STEA(13)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for nonr.
17	$STEA(14)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methane
18	$STEA(15)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO
19	$STEA(16)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO_2
20	$STEA(17)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methanol

Note: i = column number; j = row number; h = hour; VOC = volatile organic compounds; THC = total hydrocarbons; nonr. = nonreactive hydrocarbons.

b. PNTSA.CS0, PNTSU.CS0, and PNTWK.CS0

These are the unformatted point-source emissions data files, with data registered in LST, for Saturdays, Sundays, and weekdays respectively. The WRITE statements for these files are listed below, and their parameters are shown in Table 2.233. The files are used by P14G.

```
WRITE(OUNIT(A)) STACK, PTIME, (SEK(I), I=1,15).
```

TABLE 2.233. PNTSA.CS0, PNTSU.CS0, AND PNTWK.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	STACK		Char*12	AEROS state/county/plant/point ID
2	PTIME	LST	Integer*4	Hour as *HH*
3	$SEK(1)_{p,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for VOC
4	$SEK(2)_{p,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for CO
5	$SEK(3)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for olefins
6	$SEK(4)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for paraffins
7	$SEK(5)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for toluene
8	$SEK(6)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for xylene
9	$SEK(7)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for formaldehyde
10	$SEK(8)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for aldehydes
11	$SEK(9)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for ethene
12	$SEK(10)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for nonr.
13	$SEK(11)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for isoprenes
14	$SEK(12)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methane
15	$SEK(13)_{p,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO
16	$SEK(14)_{p,h}$	$tons \cdot h^{-1}$	Real*4	Emission rate for NO_2
17	$SEK(15)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emission rate for methanol

Note: p = point source; h = hour; VOC = volatile organic compounds; nonr. = nonreactive hydrocarbons.

c. MOBLSA.CS0, MOBLSU.CS0, and MOBLWK.CS0

These are the mobile-source emissions data files, with data registered in LST, for Saturdays, Sundays, and weekdays respectively. The WRITE and FORMAT statements for these files are listed below, and their parameters are shown in Table 2.234. The files are used by P26G.

```
      WRITE(OUNIT(A),1051) COL, ROW, HOUR, (STEM(I),I=1,8)
     &                     STATE, CNTY
1051  FORMAT(2I4,I2,8(1PE10.3),I2,I4)
```

TABLE 2.234. MOBLSA.CS0, MOBLSU.CS0, AND MOBLWK.CS0 PARAMETERS

Parm No.	Parm Name	Unit [a]	Data Type	Description
1	COL		Integer*4	Column number
2	ROW		Integer*4	Row number
3	HOUR	LST	Integer*4	Hour as *HH*
4	STEM(1)	$tons \cdot h^{-1}$	Real*4	Mobile CO emissions
5	STEM(2)	$tons \cdot h^{-1}$	Real*4	Mobile NO emissions
6	STEM(3)	$tons \cdot h^{-1}$	Real*4	Mobile NO_2 emissions
7	STEM(4)	$tons \cdot h^{-1}$	Real*4	Mobile evap. emissions
8	STEM(5)	$tons \cdot h^{-1}$	Real*4	Mobile exhaust emissions
9	STEM(6)	$tons \cdot h^{-1}$	Real*4	Mobile diesel emissions
10	STEM(7)	$tons \cdot h^{-1}$	Real*4	Mobile VOC emissions
11	STEM(8)	$tons \cdot h^{-1}$	Real*4	Mobile THC emissions
12	STATE		Integer*4	State AEROS code
13	CNTY		Integer*4	County AEROS code

a. Units are in English short $tons \cdot h^{-1}$; 1 ton = 907.20 kg.

III. CONTROL CARDS

Two control cards are used to input control data in the format shown below. Table 2.235 defines the control card variables.

A1 P1 A2 P2 AENDHR PENDHR OFFSET INCR
MOBLFLG AREAFLG PNTFLG

TABLE 2.235. CONTROL CARD VARIABLES

Variable Name	Unit	Description
A1	GMT	Area source beginning hour
P1	GMT	Point source beginning hour
A2	LST	Area source beginning hour
P2	LST	Point source beginning hour
AENDHR	GMT	Area source end hour
PENDHR	GMT	Point source end hour
OFFSET	h	Time difference between GMT and LST
INCR	h	Time added to first OFFSET number of GMT hours (24 - OFFSET)
MOBLFLG		Flag for mobile-source emissions: .TRUE. ⇒ include, .FALSE. ⇒ exclude
AREAFLG		Flag for area-source emissions: .TRUE. ⇒ include, .FALSE. ⇒ exclude
PNTFLG		Flag for point-source emissions: .TRUE. ⇒ include, .FALSE. ⇒ exclude

Example:

1,0,6,5,24,23,5,19
.TRUE.,.TRUE.,.TRUE.

Note that although the ROMNET domain spans two time zones (Eastern and Central), we use only EST.

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	4	files	42	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	4	files	33	blocks
Executable file:	1	file	30	blocks
	9	files	105	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	01:56:32
Buffered I/O count:	4589
Direct I/O count:	55168
Peak working-set size:	565
Peak virtual size:	2827

C. Space Requirements: Log and Print Files

P34G.LOG:	36 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.236 shows the input file and output file space requirements.

E. Space Requirements: Tape Files

None

TABLE 2.236. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	AREASAZ	Nonstd	28392	daily
	AREASUZ	Nonstd	28392	daily
	AREAWKZ	Nonstd	28392	daily
	PNTSAZ	Nonstd	57960	daily
	PNTSUZ	Nonstd	57960	daily
	PNTWKZ	Nonstd	57960	daily
	MOBLSAZ	Nonstd	26076	daily
	MOBLSUZ	Nonstd	26076	daily
	MOBLWKZ	Nonstd	26076	daily
			337284	
Output	AREASA	Nonstd	28392	daily
	AREASU	Nonstd	28392	daily
	AREAWK	Nonstd	28392	daily
	PNTSA	Nonstd	57960	daily
	PNTSU	Nonstd	57960	daily
	PNTWK	Nonstd	57960	daily
	MOBLSA	Nonstd	26076	daily
	MOBLSU	Nonstd	26076	daily
	MOBLWK	Nonstd	26076	daily
			337284	

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P34GV1A.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P34GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P34GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P34GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]P34G.OBJ          A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]ARSHFT.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]MBSHFT.OBJ        A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.PROCES]PTSHFT.OBJ        A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ         A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ          A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ        A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE4.OBJ        A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ         A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ        A10
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P34GV1A.EXE -
  /MAP = MET1:[S01S]P34GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
```

```
A8, -
A9, -
A10
$ PRINT -
/NOHEADER -
/NOFEED -
/NOTIFY -
MET1:[S01S]P34GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P34GX1.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1503A]P34GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ !ASSIGN MET52:[EMISSIONS.RNSUM85M]MOBLSAZ.CSO         MOBLSAZ
$ !ASSIGN MET52:[EMISSIONS.RNSUM85M]MOBLSUZ.CSO         MOBLSUZ
$ !ASSIGN MET52:[EMISSIONS.RNSUM85M]MOBLWKZ.CSO         MOBLWKZ
$ ASSIGN MET53:[EMISSIONS.RNSUM05A]ARSAZ.DAT            AREASAZ
$ ASSIGN MET53:[EMISSIONS.RNSUM05A]ARSUZ.DAT            AREASUZ
$ ASSIGN MET53:[EMISSIONS.RNSUM05A]ARWKZ.DAT            AREAWKZ
$ ASSIGN WORK_SCRATCH:PNTSAZ.CS1                        PNTSAZ
$ ASSIGN WORK_SCRATCH:PNTSUZ.CS1                        PNTSUZ
$ ASSIGN WORK_SCRATCH:PNTWKZ.CS1                        PNTWKZ
$ ! ASSIGN OUTPUT FILES
$ !ASSIGN MET52:[EMISSIONS.RNSUM85M]MOBLSA.CSO          MOBLSA
$ !ASSIGN MET52:[EMISSIONS.RNSUM85M]MOBLSU.CSO          MOBLSU
$ !ASSIGN MET52:[EMISSIONS.RNSUM85M]MOBLWK.CSO          MOBLWK
$ ASSIGN MET53:[EMISSIONS.RNSUM05A]AREASA.CS1           AREASA
$ ASSIGN MET53:[EMISSIONS.RNSUM05A]AREASU.CS1           AREASU
$ ASSIGN MET53:[EMISSIONS.RNSUM05A]AREAWK.CS1           AREAWK
$ ASSIGN MET53:[EMISSIONS.RNSUM05P]PNTSA.CS1            PNTSA
$ ASSIGN MET53:[EMISSIONS.RNSUM05P]PNTSU.CS1            PNTSU
$ ASSIGN MET53:[EMISSIONS.RNSUM05P]PNTWK.CS1            PNTWK
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1503A]P34GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN / USER_MODE USER2$DISK:[FOREMAN.FV1503A]P34GX1.CPR   SYS$INPUT
$ !
$ RUN DISK$VAXSET:[ROMEXE.ROMNET]P34GV1A.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P34GV1A

B. Subroutines

ARSHFT
MBSHFT
PTSHFT

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE
PROGID

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JFILE4
JUNIT

IX. INCLUDE FILES

None

2.31 PROCESSOR P36G (EMISSION CONTROL STRATEGIES)

I. PROCESSOR FUNCTION

This stage 0 processor applies NO_x and VOC emission controls at the county level to 17 area-source species and 7 mobile-source species. P36G applies controls to either or both area- and mobile-source emissions data. The control cards contain the flags for emission control strategies for the area and mobile sources. If the flag is .TRUE., P36G processes the control strategy by reading the control strategy default values and the reduction file. Different control strategy default values and reduction values exist for the area sources and mobile sources.

Note that this processor is not currently used, since the control strategies are applied at the raw data stage (i.e., on the data tapes output from the FREDS).

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. PCTCELL.DAT

This file contains the grid cell county allocation factors (i.e., the percentage of a grid cell that is comprised of each county within the cell). The READ and FORMAT statements for PCTCELL.DAT are listed below, and the file parameters are shown in Table 2.237.

```
      READ(PUNIT, 1001, IOSTAT = IOST) NCNTY
1001  FORMAT(1X, I2)

      READ(PUNIT, 1003, IOSTAT = IOST) ICOL, IROW, CCODE,
     &                                 IPCTCL
1003  FORMAT(1X, I2, 1X, I2, 1X, A6, 39X, I3)
```

TABLE 2.237. PCTCELL.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ICOL		Integer*4	ROMNET region column
2	IROW		Integer*4	ROMNET region row
3	CCODE		Char*6	AEROS county code
4	IPCTCL		Integer*4	Percentage of county in grid cell

b. ARCTRLNOX.CS*x*, ARCTRLVOC.CS*x*, MBCTRLNOX.CS*x*, MBCTRLVOC.CS*x*

These files, originating from control strategies set by the EPA's OAQPS for ROMNET (EPA,1990), contain county codes and their associated control strategies. Sample READ and FORMAT statements for these files are listed below, and the file parameters are shown in Table 2.238.

```
      READ(CUNIT, 1001, IOSTAT=IOST) MNXCID(IREC), MNXRED(IREC)
1001  FORMAT(1X, A6, F6.2)
```

TABLE 2.238. CONTROL STRATEGY FILE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IREC		Integer*4	Record count
2	MNXCID		Char*6	County code
3	MNXRED	%	Real*4	Control strategy reduction

c. AREASA.CS0, AREASU.CS0, AREAWK.CS0

These files are generated by P34G, and their parameters can be found in Table 2.232. The READ and FORMAT statements for the files are listed below.

```
      READ(ARIUN,1001,IOSTAT=IOST) ICOL, IROW, IHOUR,
     &                             (AREM(I),ISPC =1,NSPCAR)
1001  FORMAT(2I4, I2, <NSPCAR>E10.3)
```

d. MOBLSA.CS0, MOBLSU.CS0, MOBLWK.CS0

These files are generated by P34G, and their parameters can be found in Table 2.234. The READ and FORMAT statements for the files are listed below.

```
      READ(MBIUN,1001,IOSTAT=IOST) ICOL, IROW, IHOUR,
     &                             (MBEM(ISPC), ISPC=1,NSPCMB),
     &                             COUNTY
1001  FORMAT(2I4, I2, <NSPCMB>E10.3, A6)
```

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

P36G can generate as many as six nonstandard output files, depending on the control cards used: AREASA.CSx, AREASU.CSx, AREAWK.CSx, and/or MOBLSA.CSx, MOBLSU.CSx, MOBLWK.CSx. The files contain the same parameters as the .CS0 files (see Processor P34G), with data values modified by the applied control strategy. The WRITE and FORMAT statements for the AREAxx files (used by P14G) are listed below.

```
      WRITE(AROUN,1001, IOSTAT = IOST) ICOL, IROW, IHOUR,
     &                              (AREM(ISPC),ISPC =1, NSPCAR)
1001  FORMAT(2I4, I2, <NSPCAR>E10.3)
```

The WRITE and FORMAT statements for the MOBLxx files (used by P26G) are listed below.

```
      WRITE(MBOUN,1009, IOSTAT = IOST) ICOL, IROW, IHOUR,
     &              (MBEM(ISPC), ISPC =1, NSPCMB), COUNTY
1009  FORMAT(2I4, I2, <NSPCMB>(1PE10.3), A6)
```

III. CONTROL CARDS

Two control cards, together with four title cards and two default-value cards are used to input control data in the following format shown below. The control cards indicate the type of control strategy used, as well as selecting the emissions data type that is processed. Table 2.239 defines the control card variables.

CLABEL
AR_NOX, AR_VOC
CLABEL
ARDEF_NOX, ARDEF_VOC
CLABEL
MB_NOX, MB_VOC
CLABEL
MBDEF_NOX, MBDEF_VOC

TABLE 2.239. CONTROL CARD VARIABLES

Variable Name	Unit	Description
CLABEL		Title of the following card
AR_NOX		Area source NO_x control flag: .FALSE. ⇒ no NO_x control file, use default value
AR_VOC		Area source VOC control flag: .FALSE. ⇒ no VOC control file, use default value
MB_NOX		Mobile source NO_x control flag: .FALSE. ⇒ no NO_x control file, use default value
MB_VOC		Mobile source VOC control flag: .FALSE. ⇒ no VOC control file, use default value
ARDEF_NOX	%	Default control of NO_x for area sources
ARDEF_VOC	%	Default control of VOC for area sources
MBDEF_NOX	%	Default control of NO_x for mobile sources
MBDEF_VOC	%	Default control of VOC for mobile sources

Example:

AR_NOX AR_VOC

.FALSE. .FALSE.

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	11	files	76	blocks
FORTRAN INCLUDE files:	5	files	11	blocks
Object files:	11	files	71	blocks
Executable file:	1	file	40	blocks
	28	files	198	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	01:56:32
Buffered I/O count:	4589
Direct I/O count:	55168
Peak working-set size:	565
Peak virtual size:	3117

C. Space Requirements: Log and Print Files

P36G.LOG:	20 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.240 shows the input file and output file space requirements.

E. Space Requirements: Tape Files

None

TABLE 2.240. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type [a]	Storage (in blocks)	Scenario Data Span
Input	PCTCELL	Nonstd	variable	N/A
	controls	Nonstd	variable	N/A
	AREASA.CS0	Nonstd	28392	24 hours
	AREASU.CS0	Nonstd	28392	24 hours
	AREAWK.CS0	Nonstd	28392	24 hours
	MOBLSA.CS0	Nonstd	26076	24 hours
	MOBLSU.CS0	Nonstd	26076	24 hours
	MOBLWK.CS0	Nonstd	26076	24 hours
			>163404	
Output	AREASA.CSx	Nonstd	28392	24 hours
	AREASU.CSx	Nonstd	28392	24 hours
	AREAWK.CSx	Nonstd	28392	24 hours
	MOBLSA.CSx	Nonstd	26076	24 hours
	MOBLSU.CSx	Nonstd	26076	24 hours
	MOBLWK.CSx	Nonstd	26076	24 hours
			163404	

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P36G.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ !
$ ASSIGN /USER USER0$DISK:[LBW.P36G]P36G.OBJ                A1
$ ASSIGN /USER USER0$DISK:[LBW.P36G]BLKCTR.OBJ              A2
$ ASSIGN /USER USER62$DISK:[FUG.MOBILE.QA]RDPCTC.OBJ        A3
$ ASSIGN /USER USER0$DISK:[LBW.P36G]ARPROC.OBJ              A4
$ ASSIGN /USER USER62$DISK:[FUG.MOBILE.QA]MBPROC.OBJ        A5
$ ASSIGN /USER USER0$DISK:[LBW.P36G]ARCTRL.OBJ              A6
$ ASSIGN /USER USER0$DISK:[LBW.P36G]RDARCS.OBJ              A7
$ ASSIGN /USER USER62$DISK:[FUG.MOBILE.QA]RDMBCS.OBJ        A8
$ ASSIGN /USER USER0$DISK:[LBW.P36G]AGRID.OBJ               A9
$ ASSIGN /USER USER0$DISK:[LBW.P36G]RDATE.OBJ               A10
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ          A11
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ           A12
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ         A13
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ         A14
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]JFILE3.OBJ         A15
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ          A16
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]JFILEX.OBJ         A17
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ         A18
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ          A19
$ ASSIGN/USER DISK$VAXSET:[ROMOBJ.UTILIO]INDEX1.OBJ         A20
$ LINK -
  /DEBUG -
  /EXECUTABLE = USER0$DISK:[LBW.P36G]P36G.EXE -
  A1, A2, A3, A4, A5, A6, A7, A8, A9, A10, -
  A11, A12, A13, A14, A15, A16, A17, A18, A19, A20
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P36G.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! run stream is in USERO$DISK:[LBW.P36G]P36G.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! assign input control strategy files
$ ASSIGN MET51:[EMISSIONS.COUNTY]PCTCELL.DAT          PCTCELL
$ ASSIGN MET1:[EMISSIONS.NENOXCTRL]ARCTRLNOX.CS4      ARCTLNOX
$ ASSIGN MET1:[EMISSIONS.NEVOCCTRL]ARCTRLVOC.CS4      ARCTLVOC
$ ASSIGN MET1:[EMISSIONS.NENOXCTRL]MBCTRLNOX.CS4      MBCTLNOX
$ ASSIGN MET1:[EMISSIONS.NEVOCCTRL]MBCTRLVOC.CS3      MBCTLVOC
$ ! assign input data files
$ ASSIGN MET66:[EMISSIONS.RNSUM85A.BASE]AREASA.CSO    ARSA
$ ASSIGN MET66:[EMISSIONS.RNSUM85A.BASE]AREASU.CSO    ARSU
$ ASSIGN MET66:[EMISSIONS.RNSUM85A.BASE]AREAWK.CSO    ARWK
$ ASSIGN MET66:[EMISSIONS.RNSUM85M.BASE]MOBLSA.CSO    MBSA
$ ASSIGN MET66:[EMISSIONS.RNSUM85M.BASE]MOBLSU.CSO    MBSU
$ ASSIGN MET66:[EMISSIONS.RNSUM85M.BASE]MOBLWK.CSO    MBWK
$ !
$ ! assign output print files
$ ASSIGN MET66:[SCRATCH]NOXCSFPRF1N.DAT               NOXCSF
$ ASSIGN MET66:[SCRATCH]VOCCSFPRF1N.DAT               VOCCSF
$ !
$ ! assign output data files
$ ASSIGN MET66:[SCRATCH]AREASA.CST                    ARSACS
$ ASSIGN MET66:[SCRATCH]AREASU.CST                    ARSUCS
$ ASSIGN MET66:[SCRATCH]AREAWK.CST                    ARWKCS
$ ASSIGN MET66:[SCRATCH]MOBLSA.CST                    MBSACS
$ ASSIGN MET66:[SCRATCH]MOBLSU.CST                    MBSUCS
$ ASSIGN MET66:[SCRATCH]MOBLWK.CST                    MBWKCS
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER62$DISK:[FUG.MOBILE.QA]P36GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN / USER_MODE USER62$DISK:[FUG.MOBILE.QA]P36GX1.CPR   SYS$INPUT
$ !
$ RUN/NODEBUG USERO$DISK:[LBW.P36G]P36G.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P36G

B. Subroutines

BLKCTR	RDARCS	AGRID
RDPCTC	MBPROC	RDATE
ARPROC	MBCTRL	
ARCTRL	RDMBCS	

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE
CELL0
JFILE1
JFILEX
PROGID

B. Functions

DISK$VAXSET:[ROMNET.UTILIO]

INDEX1
IOCL
JFILE2
JFILE3

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

CTRLDF
CNTYGR
CSFGRD
CTRLSP

2.32 PROCESSOR P38G (BTRK FILE CREATION)

I. PROCESSOR FUNCTION

This stage 7 processor reads the model input files containing the wind fields gridded backtrack locations and the horizontal diffusivities data and produces the Backtrack file (BTRK).[14] Together with the boundary conditions file BCON, BTRK contains the data necessary for the Core Model to compute the horizontal advection component of the predicted concentrations of the model's 35 chemical species.

14. The definition of *backtrack* can be found in Processor P29G.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

MF103 from P29G
MF104 from P29G
MF105 from P29G
MF106 from P29G
MF107 from P29G
MF108 from P29G

2. Nonstandard Input Files

None

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

BTRK contains data elements for each grid cell, model layer, and time step required by the Core Model. The structure of the BTRK file conforms to the requirements for all ROM Core Model files, and therefore contains a file header and time-step header structure. Descriptions of the records containing this information are given below.

a. Record 1

The first record contains character strings representing alphanumeric data that describe the file's contents. The data are prepared and written to a character buffer, which is then written unformatted to the BTRK file by subroutine WRCHAR. The parameters of record 1 are shown in Table 2.241.

```
C first segment record . . . . . . . . . . . . . . . . . . . . . . . .
      WRITE(RECONE, 1001, IOSTAT = IOST)
     &      CDATBT, CTIMBT, SDATBT, STHRBT, TSTPBT, FRSTBT,
     &      GRDNBT,
     &      SWLNBT, SWLTBT, NELNBT, NELTBT,
     &      DLONBT, DLATBT,
     &      NCOLBT, NROWBT, NMIFBT, ICNTBT
```

```
1001  FORMAT(6I8.8, A8, 4F8.3, 2F8.5, 4I4.4)
C
      CALL WRCHAR(IBTRK, RECONE, IOST)

      SUBROUTINE WRCHAR (IUNIT, CHBUF, IOST)
      IMPLICIT NONE
      INTEGER*4 IUNIT, IOST
      CHARACTER*(*) CHBUF
      WRITE(IUNIT, IOSTAT = IOST) CHBUF
      RETURN
      END
```

TABLE 2.241. BTRK RECORD 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CDATBT		Integer*4	File creation date as *MMDDYY*
2	CTIMBT	EST	Integer*4	File creation time as *HHMMSS*
3	SDATBT		Integer*4	Julian start date of scenario as *YYDDD*
4	STHRBT	EST	Integer*4	Start hour of scenario (00 to 23)
5	TSTPBT	s	Integer*4	Time step size for simulation
6	FRSTBT	s	Integer*4	Time to first step on BTRK file
7	GRDNBT		Char*8	Grid definition name
8	SWLNBT	° W	Real*4	Longitude of southwest corner of grid
9	SWLTBT	° N	Real*4	Latitude of southwest corner of grid
10	NELNBT	° W	Real*4	Longitude of northeast corner of grid
11	NELTBT	° N	Real*4	Latitude of northeast corner of grid
12	DLONBT	° W	Real*4	Grid cell longitudinal increment
13	DLATBT	° N	Real*4	Grid cell latitudinal increment
14	NCOLBT		Integer*4	Number of columns in grid
15	NROWBT		Integer*4	Number of rows in grid
16	NMIFBT		Integer*4	Number of model input files used to generate the BTRK file
17	ICNTBT		Integer*4	Number of text records

b. Records 2 - 5

These records contain the list of model input file logical names that are read to generate the BTRK file data. The parameters of these records are shown in Table 2.242.

```
MIF segment records . . . . . . . . . . . . . . . . . . . . . . . .
. .
      DO 201 IMIF = 1, NMIFBT
      WRITE(RECMIF, 1003, IOSTAT = IOST) MFNMBT(IMIF), CDMFBT(IMIF),
     &                                   CTMFBT(IMIF), UDMFBT(IMIF),
     &                                   UTMFBT(IMIF)
1003  FORMAT(A12, 4I8.8)
      CALL WRCHAR(IBTRK, RECMIF, IOST)
201   CONTINUE
```

TABLE 2.242. BTRK RECORDS 2 - 5 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	MFNMBT(J)		Char*12	File name of MIF file *J*
2	CDMFBT(J)		Integer*4	Creation date of MIF file *J*
3	CTMFBT(J)		Integer*4	Creation time of MIF file *J*
4	UDMFBT(J)		Integer*4	Last update date of MIF file *J*
5	UTMFBT(J)		Integer*4	Last update time of MIF file *J*

c. Records 6 - (25 maximum)

These records contain descriptive text that you can optionally enter as part of the control cards data set. One 80-character string is written to each record by means of the following statements. A blank string *must* denote the end of the descriptive text records. The WRITE and FORMAT statements for these records are listed below, and their parameter is shown in Table 2.243.

```
C text segment records . . . . . . . . . . . . . . . . . . . . . .
      DO 301 ITXT = 1, ICNTBT
      WRITE(RECTXT, 1005, IOSTAT = IOST) TEXTBT(ITXT)
1005  FORMAT(A80)
      CALL WRCHAR(IBTRK, RECTXT, IOST)
301   CONTINUE
```

TABLE 2.243. RECORDS 6 - (6+ICNTBT-1) PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	TEXTBT(ITXT)		Char*80	Text description

These end the file header records. The following are the data records, headed by one time-step record for each time interval.

d. Time-Step Header Record

There is one time-step header record written for each time step increment on the BTRK file. The time-step header data are contained in a common block. The following statement writes the data to the BTRK file by referencing the first variable of the common block, and specifying the number of the block's words to be written. The data are then written to the file by subroutine WRFILE. The WRITE and FORMAT statements for this record are listed below, and its parameters are shown in Table 2.244.

```
COMMON / RTSHBT / DATBT, TIMBT, ELPBT, STPBT
NWRDST = 4
CALL WRFILE(IBTRK, NWDSTS, DATBT, IOST)

SUBROUTINE WRFILE (IUNIT, NWORDS, BUFFER, IOST)
IMPLICIT NONE
INTEGER*4 IUNIT, NWORDS, IOST
REAL*4 BUFFER (NWORDS)
WRITE(IUNIT, IOSTAT = IOST) BUFFER
RETURN
END
```

TABLE 2.244. TIME-STEP HEADER RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATBT		Integer*4	Julian current time step date as *YYDDD*
2	TIMBT	EST	Integer*4	Current time step time as *HHMMSS*
3	ELPBT	s	Integer*4	Elapsed time since scenario start
4	STPBT		Integer*4	Current time-step no. on BTRK file

e. Data Records

The data records are written by equivalencing common blocks of row data for each group. The following parameter is used for dimensioning the common blocks and controlling the write operations:

NVARBT = 4 is the number of level-dependent BCKTRAK variables, specifically: UU, VV, KKHU, KKHV.

We use a named common block, BTRKVR, for the specific variables, and the same common block for a buffer, BTRKBF, that is written unformatted to the BTRK file by subroutine WRFILE:

```
 COMMON / BTRKVR /
&       U1(NCOLS1), U2(NCOLS1), U3(NCOLS1),
&       V1(NCOLS1), V2(NCOLS1), V3(NCOLS1),
&       KH1U(NCOLS1), KH2U(NCOLS1), KH3U(NCOLS1),
&       KH1V(NCOLS1), KH2V(NCOLS1), KH3V(NCOLS1)

 COMMON / BTRKVR / BTRKBF(NCOLS1, NLEVS * NVARBT)
```

The procedure to write these data is listed below, and the file's parameters are shown in Table 2.245.

```
      NWRDSB = NLEVS * NVARBT * NCOLS1

      IROW = 1
201   CONTINUE
      IF (IROW .GT. NROWS1) GO TO 301
C
C write BTRK row
      CALL WRFILE (IBTRK, NWRDSB, BTRKBF(1,1), IOST)
C
C read BT row
C
      IGO = 3
      RETURN
30001 CONTINUE
C
      IROW = IROW + 1
      GO TO 201
C
301   CONTINUE
```

TABLE 2.245. BTRK DATA PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Backtrack locations:
1	UU1	column	Real*4	Longitudinal-component in layer 1
2	UU2	column	Real*4	Longitudinal-component in layer 2
3	UU3	column	Real*4	Longitudinal-component in layer 3
4	VV1	row	Real*4	Latitudinal-component in layer 1
5	VV2	row	Real*4	Latitudinal-component in layer 2
6	VV3	row	Real*4	Latitudinal-component in layer 3
				Horizontal diffusivities:
7	KKH1U	$rad^2 \cdot s^{-1}$	Real*4	Longitudinal-component in layer 1
8	KKH2U	$rad^2 \cdot s^{-1}$	Real*4	Longitudinal-component in layer 2
9	KKH3U	$rad^2 \cdot s^{-1}$	Real*4	Longitudinal-component in layer 3
10	KKH1V	$rad^2 \cdot s^{-1}$	Real*4	Latitudinal-component in layer 1
11	KKH2V	$rad^2 \cdot s^{-1}$	Real*4	Latitudinal-component in layer 2
12	KKH3V	$rad^2 \cdot s^{-1}$	Real*4	Latitudinal-component in layer 3

III. CONTROL CARDS

Two control cards are used to input the control data using the code shown below. Following these control cards, you can optionally enter file-description text in as many as 20 80-character records. This text is subsequently written to the BTRK file's header. Note that one blank text record is required at the end of these character records, even if you opt to enter no text. Table 2.246 defines the control card variables.

You will see that dummy records are read into a character variable, CLABEL. These records exist in the input stream to describe what the following card of data represents.

The control cards are input using the following format:

SDATBT STHRBT SSDATE SSHOUR ESDATE ESHOUR
VAXFMT
TEXTBT(K)

The code that reads the control card information follows.

```
C read control cards
      READ(5, 1005) CLABEL
1005  FORMAT(A80)
      READ(5, *) SDATBT, STHRBT, SSDATE, SSHOUR,
     &          ESDATE, ESHOUR
      SSTIME = SSHOUR * 10000
      ESTIME = ESHOUR * 10000
      READ(5, 1005) CLABEL
      READ(5, *) VAXFMT
      READ(5, 1003) CLABEL

      DO 101 ITXT = 1, NTEXT
      READ(5, 1007, IOSTAT = IOST) TEXTBT(ITXT)
1007  FORMAT(A80)
      IF (TEXTBT(ITXT) .EQ. BLANK) GO TO 111
101   CONTINUE
111   CONTINUE
      ICNTBT = ITXT - 1
```

TABLE 2.246. CONTROL CARD VARIABLES

Variable Name	Unit	Description
SDATBT		Julian scenario start date as *YYDDD*
STHRBT	EST	Scenario start hour as *HH*
SSDATE		Julian simulation start date as *YYDDD*
SSHOUR	EST	Simulation start hour as *HH*
ESDATE		Julian simulation end date as *YYDDD*
ESHOUR	EST	Simulation end hour as *HH*
VAXFMT		Translation flag: .TRUE.⇒ do not translate to IBM binary .FALSE.⇒ translate to IBM binary
TEXTBT(K)		Descriptive text records to be recorded on the BTRK file header

Example:

88187 12 88187 12 88191 12
.FALSE.

```
    ...          BACKTRACK FILE (BTRK)        ...
            for ROM2.1 on the ROMNET grid
   generated from formulation FV1501A for 144  steps
    using  USER2$DISK:[FOREMAN.FV1503A]P39GX1.COM
           USER2$DISK:[FOREMAN.FV1503A]P39GX1.CPR
           DISK$VAXSET:[ROMEXE.ROMNET]P39GV1B.EXE
                                                  21 JUL 89
.  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
```

{blank line of text}

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	8	files	79	blocks
FORTRAN INCLUDE files:	16	files	90	blocks
Object files:	8	files	43	blocks
Executable file:	1	file	72	blocks
	27	files	284	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:04:09
Buffered I/O count:	274
Direct I/O count:	4401
Peak working-set size:	1549
Peak virtual size:	3971

C. Space Requirements: Log and Print Files

P38GX1.LOG:	54 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.247 shows the input file and output file space requirements.

TABLE 2.247. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	MF103	5	3798	72 hours
	MF104	5	3798	72 hours
	MF105	5	3798	72 hours
	MF106	5	5696	72 hours
	MF107	5	5696	72 hours
	MF108	5	5696	72 hours
			28482	72 hours
Output	BTRK	Nonstd	45000	72 hours

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P38GV1B.LNK follows.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P38GV1B.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P38GV1B.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P38GV1B.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P38G.OBJ          A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]BLKBTR.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]INTRPD.OBJ        A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDDIFF.OBJ        A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDWIND.OBJ        A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]TIMEBT.OBJ        A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRBTRK.OBJ        A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRHDBT.OBJ        A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ         A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ         A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CLOCK1.OBJ        A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CLOCK2.OBJ        A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR3.OBJ        A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR5.OBJ        A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ERROUT.OBJ        A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ        A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ        A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ         A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ          A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ        A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE6.OBJ        A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ        A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ         A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ        A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ         A25
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ        A26
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ        A27
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD1.OBJ        A28
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ         A29
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ         A30
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]TRNSLT.OBJ        A31
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRCHAR.OBJ        A32
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFILE.OBJ        A33
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P38GV1B.EXE -
  /MAP = MET1:[S01S]P38GV1B.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A31, -
  A32, -
  A33
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P38GV1B.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P38GX1.COM follows.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1503A]P38GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ !ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT            PFMFDIR
$ ASSIGN MET56:[ROMNET1.D85188]MF103A.DAT    MF103
$ ASSIGN MET56:[ROMNET1.D85188]MF104A.DAT    MF104
$ ASSIGN MET56:[ROMNET1.D85188]MF105A.DAT    MF105
$ ASSIGN MET56:[ROMNET1.D85188]MF106A.DAT    MF106
$ ASSIGN MET56:[ROMNET1.D85188]MF107A.DAT    MF107
$ ASSIGN MET56:[ROMNET1.D85188]MF108A.DAT    MF108
$ ! ASSIGN OUTPUT FILES
$ ASSIGN -
NCCIBM::"""ADRNER1.NOBKUP.BTRK.FV1503A/NOTRANSLATE/SINGLE/SUPERSEDE""" -
BCKTRAK
$ !SYNCHRONIZE/ENTRY=592
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1503A]P38GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN USER2$DISK:[FOREMAN.FV1503A]P38GX1.CPR   SYS$INPUT
$ !
$ ASSIGN DISK$VAXSET:[ROMEXE.ROMNET]P38GV1B.EXE   PROCP38G
$ DIRECTORY / FULL PROCP38G
$ RUN PROCP38G
$ SHOW PROCESS / ACCOUNTING
$ SHOW STATUS
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P38GV1B

B. Subroutines

INTRPD	TIMEBT
RDDIFF	WRBTRK
RDWIND	WRHDBT

C. Functions

None

D. Block Data Files

BLKBTR

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	ERROUT	OPDIR2	RDFT5
CELL0	FLSMRY	OPEN2	RDREC
CLOCK1	FSTAT1	PROGID	TRNSLT
CLOCK2	IBLKR	RDDIR1	WRCHAR
DATHR5	JFILE1	RDFHD1	WRFILE

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

DATHR3	JFILE6
IOCL	JHOUR1
JFILE2	JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

BTLUNS.EXT
BTMFLS.EXT
BTRPRM.EXT
BTRKBF.EXT
BTRKVR.EXT

DISK$VAXSET:[ROMLIB.UTILIO]

DATIME.EXT
PRID.EXT

DISK$VAXSET:[ROMLIB.CORE]

HEADBT.EXT
RTSHBT.EXT

2.33 PROCESSOR P39G (IMET FILE CREATION)

I. PROCESSOR FUNCTION

This stage 7 processor reads all the model input files that contain (1) meteorology data, and (2) the deposition velocity and upper boundary condition concentrations (C_∞) for 12 of the model's 35 species. P39G writes the Intermediate Meteorology file, IMET. The purpose of IMET is to separate all the meteorology data from the emissions data. P40G then takes the IMET data and adds the data from the model input files generated by the emissions processors to produce the B-Matrix file. We designed the ROM this way to allow for the production of many different emissions-reduction scenarios for any one set of meteorological conditions. The design also permits outside users to run their own emissions reductions while accessing IMET files that have been previously generated in-house at the EPA.

To produce the IMET data for a multiday scenario, P39G uses multiple 3-day simulations. For example, one 15-day scenario requires five 3-day simulations; each simulation is contained in a separate IMET file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

MF110 from P07G	MF140 from P09G	MF161 from P09G
MF111 from P07G	MF146 from P15G	MF162 from P09G
MF129 from P12G	MF147 from P23G	MF163 from P09G
MF132 from P09G	MF151 from P12G	MF164 from P17G
MF133 from P09G	MF152 from P12G	MF165 from P17G
MF134 from P09G	MF153 from P12G	MF166 from P17G
MF135 from P09G	MF154 from P12G	MF167 from P17G
MF136 from P09G	MF155 from P12G	MF174 from P09G
MF137 from P09G	MF156 from P12G	MF175 from P09G
MF138 from P09G	MF157 from P12G	MF176 from P09G
MF139 from P09G	MF160 from P09G	

2. Nonstandard Input Files

None

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

The IMET file contains the meteorology data for the model's chemistry rate-constant adjustments as well as the deposition velocity and C_{∞} values for some of the model's 35 chemical species. All data on the IMET file have been interpolated from hourly to 1800-second time steps using a scheme that weights (1) each hour's data by 75% and the subsequent hour's data by 25% for the first 1800-s time step, and (2) each hour's data by 25% and the subsequent hour's data by 75% for the second 1800-s time step. Although the IMET file is not strictly a ROM Core Model file, it nevertheless conforms to the requirements for all ROM Core Model files, and therefore contains a file header and time-step header structure. Descriptions of the records containing this information are given below.

a. Record 1

The first record contains character strings representing alphanumeric data that describe the file's contents. The data are prepared and written to a character buffer, which is then written unformatted to the IMET file by subroutine WRCHAR. The parameters of record 1 are shown in Table 2.248.

```
C first segment record . . . . . . . . . . . . . . . . . . . . . .
      WRITE(RECONE, 1001, IOSTAT = IOST)
     &       CDATIM, CTIMIM, SDATIM, STHRIM, TSTPIM, FRSTIM,
     &       GRDNIM,
     &       SWLNIM, SWLTIM, NELNIM, NELTIM,
     &       DLONIM, DLATIM,
     &       NCOLS1, NROWS1, NLEVS, NSPCIM, NMIFIM, ICNTIM
1001  FORMAT(6I8.8, A8, 4F8.3, 2F8.5, 6I4.4)
C
      CALL WRCHAR(IIMET, RECONE, IOST)

      SUBROUTINE WRCHAR (IUNIT, CHBUF, IOST)
      IMPLICIT NONE
      INTEGER*4 IUNIT, IOST
      CHARACTER*(*) CHBUF
      WRITE(IUNIT, IOSTAT = IOST) CHBUF
      RETURN
      END
```

b. Record 2

This record contains the list of species names for which the Core Model computes concentration outputs. As for record 1 above, the data are written to a character buffer, which is then written unformatted to the IMET file. The parameter of record 2 is shown in Table 2.249.

```
C species names record . . . . . . . . . . . . . . . . . . . . . . . . .
. .
      WRITE(RECSPN, 1003, IOSTAT = IOST) (SPNMIM(ISPC), ISPC = 1, NSPCIM)
1003  FORMAT(<NSPCIM>A4)
      CALL WRCHAR(IIMET, RECSPN, IOST)
```

TABLE 2.248. IMET RECORD 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CDATIM		Integer*4	File creation date as *MMDDYY*
2	CTIMIM	EST	Integer*4	File creation time as *HHMMSS*
3	SDATIM		Integer*4	Julian start date of scenario as *YYDDD*
4	STHRIM	EST	Integer*4	Start hour of scenario (00 to 23)
5	TSTPIM	s	Integer*4	Time step size for simulation
6	FRSTIM	s	Integer*4	Time to first step on IMET file
7	GRDNIM		Char*8	Grid definition name
8	SWLNIM	° W	Real*4	Longitude of southwest corner of grid
9	SWLTIM	° N	Real*4	Latitude of southwest corner of grid
10	NELNIM	° W	Real*4	Longitude of northeast corner of grid
11	NELTIM	° N	Real*4	Latitude of northeast corner of grid
12	DLONIM	° W	Real*4	Grid cell longitudinal increment
13	DLATIM	° N	Real*4	Grid cell latitudinal increment
14	NCOLS1		Integer*4	Number of columns in grid
15	NROWS1		Integer*4	Number of rows in grid
16	NLEVS		Integer*4	Number of levels
17	NSPCIM		Integer*4	Number of model species
18	NMIFIM		Integer*4	Number of model input files used to generate the IMET file
19	ICNTIM		Integer*4	Number of text records

TABLE 2.249. IMET RECORD 2 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SPNMIM(I)		Char*4	Name of species *I*

c. Record 3

This record contains the expansion list (look-up table) of species names for the deposition velocities for which the IMET file has values. (Refer to processor P23G for an explanation of the need for an expansion list.) These values are mapped to the full core-model list of species names in processor P40G. The WRITE and FORMAT statements for this record are listed below, and its parameter is shown in Table 2.250.

```
      WRITE(RECSPN, 1003, IOSTAT = IOST) (DEPVNX(ISPC), ISPC = 1, NSPCIM)
1003  FORMAT(<NSPCIM>A4)
      CALL WRCHAR(IBMAT, RECSPN, IOST)
```

TABLE 2.250. IMET RECORD 3 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	DEPVNX(K)		Integer*4	Index array for species expansion of deposition velocities from the reduced IMET set to the full model set

d. Record 4

This record contains the expansion list (look-up table) of species names for the C_∞ concentrations for which the IMET file has values. These values are mapped to the full core-model list of species names in processor P40G. The WRITE and FORMAT statements for this record are listed below, and its parameter is shown in Table 2.251.

```
      WRITE(RECSPN, 1003, IOSTAT = IOST) (CINFNX(ISPC), ISPC = 1, NSPCIM)
1003  FORMAT(<NSPCIM>A4)
      CALL WRCHAR(IBMAT, RECSPN, IOST)
```

TABLE 2.251. IMET RECORD 4 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	CINFNX(K)		Integer*4	Index array for species expansion of C_∞ concentrations from the reduced IMET set to the full model set

e. Record 5

This record contains the list of Core Model level names. The WRITE and FORMAT statements for this record are listed below, and its parameter is shown in Table 2.252.

```
C level names record . . . . . . . . . . . . . . . . . . . . . . .
      WRITE(RECLVN, 1005, IOSTAT = IOST) (LVNMIM(ILEV), ILEV = 1, NLEVS)
1005  FORMAT(<NLEVS>A4)
      CALL WRCHAR (IIMET, RECLVN, IOST)
```

TABLE 2.252. IMET RECORD 5 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	LVNMIM(J)		Char*4	Name of level *J*

f. Records 6 - 37

These records contain the list of logical names of the model input files (MIF) that are used to generate the IMET file data. The WRITE and FORMAT statements for these records are listed below, and their parameters are shown in Table 2.253.

```
C MIF segment records . . . . . . . . . . . . . . . . . . . . . .
      DO 501 IMIF = 1, NMIFIM
      WRITE(RECMIF, 1007, IOSTAT = IOST) MFNMIM(IMIF), CDMFIM(IMIF),
     &                                   CTMFIM(IMIF), UDMFIM(IMIF),
     &                                   UTMFIM(IMIF)
1007  FORMAT(A12, 4I8.8)
      CALL WRCHAR(IIMET, RECMIF, IOST)
501   CONTINUE
```

TABLE 2.253. IMET RECORDS 6 - 37 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	MFNMIM(J)		Char*12	File name of MIF file *J*
2	CDMFIM(J)		Integer*4	Julian creation date *(YYDDD)* of MIF file *J*
3	CTMFIM(J)	EST	Integer*4	Creation time *(HH)* of MIF file *J*
4	UDMFIM(J)		Integer*4	Last update Julian date of MIF file *J*
5	UTMFIM(J)	EST	Integer*4	Last update time of MIF file *J*

g. Records 38 - (57 maximum)

These records contain descriptive text that you can optionally enter as part of the control cards data set. One 80-character string is written to each record by means of the following statements. A blank string *must* denote the end of the descriptive text records. The WRITE and FORMAT statements for these records are listed below, and their parameter is shown in Table 2.254.

```
C text segment records . . . . . . . . . . . . . . . . . . . . . .
      DO 601 ITXT = 1, ICNTIM
      WRITE(RECTXT, 1009, IOSTAT = IOST) TEXTIM(ITXT)
1009  FORMAT(A80)
      CALL WRCHAR(IIMET, RECTXT, IOST)
601   CONTINUE
```

TABLE 2.254. RECORDS 38 - (38+ICNTIM-1) PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	TEXTIM(ITXT)		Char*80	Text description

These end the file header records. The following are the data records, headed by one time-step record for each time interval.

h. Time-Step Header Record

There is one time-step header record written for each time step increment on the IMET file. The time-step header data are contained in a common block. The following statement writes the data to the IMET file by referencing the first variable of the common block, and specifying the number of the block's words to be written. The data are then written to the file by subroutine WRFILE. The WRITE and FORMAT statements for this record are listed below, and its parameters are shown in Table 2.255.

```
COMMON / RTSHIM / DATIM, TIMIM, ELPIM, STPIM
NWRDST = 4
CALL WRFILE(IIMET, NWDSTS, DATIM, IOST)

SUBROUTINE WRFILE (IUNIT, NWORDS, BUFFER, IOST)
IMPLICIT NONE
INTEGER*4 IUNIT, NWORDS, IOST
REAL*4 BUFFER (NWORDS)
WRITE(IUNIT, IOSTAT = IOST) BUFFER
RETURN
END
```

TABLE 2.255. TIME-STEP HEADER RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATIM		Integer*4	Julian current time step date as *YYDDD*
2	TIMIM	EST	Integer*4	Current time step time as *HHMMSS*
3	ELPIM	s	Integer*4	Elapsed time since scenario start
4	STPIM		Integer*4	Current time-step no. on IMET file

i. Data Records

There are two groups of data records for each IMET time step increment. The first group (GROUP 1 IMET PARAMETERS, Table 2.256) contains one record of variables for each model domain row. The second group (GROUP 2 IMET PARAMETERS, Table 2.257) contains a data record for each model domain row, for each species for which the IMET file has values.

The data records are written by equivalencing common blocks of row data for each group. The following parameters are used for dimensioning the common blocks and controlling the write operations:

NVRIM1 = 30;

NVRIM1 is the number of meteorology parameter INTRMET variables (Part 1), specifically: SIGT0, SIGT1, BH3DOT, SCRV1, W2MNS, W2PLS, W1MNS, W1PLS, W0MNS, W0PLS, SRHO0, SRHO1, SRHO2, SRHO3, RHO0, RHO1, RHO2, RHO3, Z0, Z1, Z2, Z3, TEMP1, TEMP2, TEMP3, WVC1, WVC2, WVC3, THETA, PSI2;

NVRIM1 = 2;

NVRIM2 is the number of species-dependent INTRMET variables (Part 2), specifically BETA and CINF.

The first group (Part 1) uses a named common block, METVRS, for the specific variables, and the same common block for a buffer, METBUF, that is written unformatted to the IMET file by subroutine WRFILE:

```
      COMMON / METVRS /
     &        SIGTO(NCOLS1), SIGT1(NCOLS1),
     &        BH3DOT(NCOLS1), SCRV3(NCOLS1),
     &        W2MNS(NCOLS1), W2PLS(NCOLS1),
     &        W1MNS(NCOLS1), W1PLS(NCOLS1),
     &        WOMNS(NCOLS1), WOPLS(NCOLS1),
     &        SRHO0(NCOLS1), SRHO1(NCOLS1),
     &        SRHO2(NCOLS1), SRHO3(NCOLS1),
     &        RHO0(NCOLS1), RHO1(NCOLS1),
     &        RHO2(NCOLS1), RHO3(NCOLS1),
     &        Z0(NCOLS1), Z1(NCOLS1),
     &        Z2(NCOLS1), Z3(NCOLS1),
     &        TEMP1(NCOLS1), TEMP2(NCOLS1), TEMP3(NCOLS1),
     &        WVC1(NCOLS1), WVC2(NCOLS1), WVC3(NCOLS1),
     &        THETA(NCOLS1), PSI2(NCOLS1)

      COMMON / METVRS / METBUF(NCOLS1, NVRIM1)
```

The second group (Part 2) uses a named common block, SPCVAR, for the specific variables, and the same common block for a buffer, SPCBUF, which is written unformatted to the IMET file by means of routine WRFILE:

```
      COMMON / SPCVAR / BETA(NCOLS1, MXPRM),
     &                  CINF(NCOLS1, MXPRM)

      COMMON / SPCVAR / SPCBUF(NCOLS1, MXPRM, NVRIM2)
```

The procedure to write these data is as follows:

```
      REAL*4 BUF(NCOLS1, NVRIM2)
      NWDSMT = NVRIM1 * NCOLS1
      NWDSSP = NVRIM2 * NCOLS1

      IROW = 1
201   CONTINUE
      IF (IROW .GT. NROWS1) GO TO 301
C
C write block 1 for the row
      CALL WRFILE (IIMET, NWDSMT, METBUF(1,1), IOST)
C
C row body . . . . . . . . . . . . . . . . . . species-reduced-list*
C
C write block 2 for the row
      DO 251 IPRM = 1, MXPRM
C
      DO 231 IFL = 1, NVRIM2
      DO 231 ICOL = 1, NCOLS1
C transfer and reorder data from block 2 buffer
      BUF(ICOL, IFL) = SPCBUF(ICOL, IPRM, IFL)
231   CONTINUE
      CALL WRFILE (IIMET, NWDSSP, BUF, IOST)
C
251   CONTINUE
C
C row body . . . . . . . . . . . . . . . . . . . . . . . . end
C
C read IM row
C
      IGO = 3
      RETURN
30001 CONTINUE
C
      IROW = IROW + 1
      GO TO 201
C
C tstep body . . . . . . . . . . . . . . . . . . . . . . . . . . .end
C
301   CONTINUE
```

where MXPRM = 12, the maximum of the species represented in the deposition velocities and the C_∞ concentrations. MXPRM is the number of species in the reduced set representing the full set of model species.

TABLE 2.256. IMET PART 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				Terrain penetration fraction through:
1	SIGTO		Real*4	layer 0
2	SIGT1		Real*4	layer 1
3	BH3DOT	$m \cdot s^{-1}$	Real*4	Volume flux through the top of layer 3
4	SCRV3	$m \cdot s^{-1}$	Real*4	Cumulus-root volume flux cloud-detrainment velocity
				Volume flux relative to top surface of:
5	W2MNS	$m \cdot s^{-1}$	Real*4	layer 2, downward flux
6	W2PLS	$m \cdot s^{-1}$	Real*4	layer 2, upward flux
7	W1MNS	$m \cdot s^{-1}$	Real*4	layer 1, downward flux
8	W1PLS	$m \cdot s^{-1}$	Real*4	layer 1, upward flux
9	W0MNS	$m \cdot s^{-1}$	Real*4	layer 0, downward flux
10	W0PLS	$m \cdot s^{-1}$	Real*4	layer 0, upward flux
				Interface air density at top surface of:
11	SRHO0	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 0
12	SRHO1	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 1
13	SRHO2	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 2
14	SRHO3	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 3
				Layer-averaged air density in:
15	RHO0	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 0
16	RHO1	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 1
17	RHO2	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 2
18	RHO3	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 3
				Mean sea level elevation of top surface of:
19	Z0	m	Real*4	layer 0
20	Z1	m	Real*4	layer 1
21	Z2	m	Real*4	layer 2
22	Z3	m	Real*4	layer 3
				Air temperature in:
23	TEMP1	K	Real*4	layer 1
24	TEMP2	K	Real*4	layer 2
25	TEMP3	K	Real*4	layer 3
				Water vapor concentration in:
26	WVC1	ppm	Real*4	layer 1
27	WVC2	ppm	Real*4	layer 2
28	WVC3	ppm	Real*4	layer 3
29	THETA	deg	Real*4	Solar zenith angle
30	PSI2		Real*4	Cloud cover correction factor

TABLE 2.257. PART 2 IMET PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
31	BETA	$m \cdot s^{-1}$	Real*4	Deposition velocity
32	CINF	ppm	Real*4	Upper boundary condition concentration (C_{∞}) for the 12 species of the reduced species list

III. CONTROL CARDS

Three control cards are used to input the control data using the code shown below. Following these control cards, you can optionally enter file-description text in as many as 20 80-character records. This text is subsequently written to the IMET file's header. Note that one blank text record is required at the end of these character records, even if you opt to enter no text. Table 2.258 defines the control card variables.

You will see that dummy records are read into a character variable, CLABEL. These records exist in the input stream to describe what the following card of data represents.

The control cards are input using the following format:

SDATIM STHRIM SSDATE SSHOUR ESDATE ESHOUR
NSTGRP LBSTP(I) LESTP(I)
NCELL NNC(J) NNR(J)
TEXTIM(K)

The code that reads the control card information follows.

```
C read control cards
      READ(5, 1003) CLABEL
1003  FORMAT(A80)
      READ(5, *) SDATIM, STHRIM, SSDATE, SSHOUR,
     &          ESDATE, ESHOUR
      SSTIME = SSHOUR * 10000
      ESTIME = ESHOUR * 10000
      READ(5, 1003) CLABEL
      READ(5, *) NSTGRP, (LBSTP(IGRP), LESTP(IGRP), IGRP = 1, NSTGRP)
      READ(5, 1003) CLABEL
      READ(5, *) NCELL, (NNC(ICELL), NNR(ICELL), ICELL = 1, NCELL)
C
      DO 101 ITXT = 1, NTEXT
      READ(5, 1003, IOSTAT = IOST) TEXTIM(ITXT)
1003  FORMAT(A80)
      IF (TEXTIM(ITXT) .EQ. BLANK) GO TO 111
101   CONTINUE
111   CONTINUE
      ICNTIM = ITXT - 1
```

TABLE 2.258. CONTROL CARD VARIABLES

Variable Name	Unit	Description
SDATIM		Julian scenario start date as *YYDDD*
STHRIM	EST	Scenario start hour as *HH*
SSDATE		Julian simulation start date as *YYDDD*
SSHOUR	EST	Simulation start hour as *HH*
ESDATE		Julian simulation end date as *YYDDD*
ESHOUR	EST	Simulation end hour as *HH*
		For reports to a print file:
NSTGRP		Number of time step groups to report
LBSTP(I)		Start step for time step group *I*
LESTP(I)		End step for time step group *I*
NCELL		Number of cell values to report
NNC(J)		Column index for reporting cell *J*
NNR(J)		Row index for reporting cell *J*
TEXTIM(K)		Descriptive text records to be recorded on the IMET file header

Example:

```
88187 12 88187 12 88191 12
02  19 22  94 97
03  15 11  41 17  49 24
 ... RURAL OZONE INTERMEDIATE METEOROLOGY FILE (INTRMET) ...
          for ROM2.1 on the ROMNET grid
    generated from formulation FV1501A for 144  steps
     using  USER2$DISK:[FOREMAN.FV1503A]P39GX1.COM
            USER2$DISK:[FOREMAN.FV1503A]P39GX1.CPR
            DISK$VAXSET:[ROMEXE.ROMNET]P39GV1A.EXE
                                                  21 JUL 89
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
                                                  {blank line of text}
```

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	11	files	85	blocks
FORTRAN INCLUDE files:	12	files	36	blocks
Object files:	11	files	83	blocks
Executable file:	1	file	114	blocks
	35	files	218	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:17:28
Buffered I/O count:	9999
Direct I/O count:	333333
Peak working-set size:	4096
Peak virtual size:	5000

C. Space Requirements: Log and Print Files

P39GX1.LOG:	99 blocks
Print Files:	Variable; none if not implemented by control cards

D. Space Requirements: Input and Output Files

Table 2.259 shows the input file and output file space requirements.

E. Space Requirements: Tape Files

None

This space deliberately left blank.

TABLE 2.259. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	MF110	5	1899	72 hours
	MF111	5	1899	72 hours
	MF129	5	1899	72 hours
	MF132	5	1899	72 hours
	MF133	5	1899	72 hours
	MF134	5	1899	72 hours
	MF135	5	1899	72 hours
	MF136	5	1899	72 hours
	MF137	5	1899	72 hours
	MF138	5	1899	72 hours
	MF139	5	1899	72 hours
	MF140	5	1899	72 hours
	MF146	5	18684	72 hours
	MF147	5	18684	72 hours
	MF151	5	1899	72 hours
	MF152	5	1899	72 hours
	MF153	5	1899	72 hours
	MF154	5	1899	72 hours
	MF155	5	1899	72 hours
	MF156	5	1899	72 hours
	MF157	5	1899	72 hours
	MF160	5	1899	72 hours
	MF161	5	1899	72 hours
	MF162	5	1899	72 hours
	MF163	5	1899	72 hours
	MF164	5	1899	72 hours
	MF165	5	1899	72 hours
	MF166	5	1899	72 hours
	MF167	5	1899	72 hours
	MF174	5	1899	72 hours
	MF175	5	1899	72 hours
	MF176	5	1899	72 hours
			94338	
Output	IMET	Nonstd	203124	72 hours

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P39GV1A.LNK follows.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P39GV1A.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P39GV1A.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P39GV1A.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P39G.OBJ              A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]BLKIMT.OBJ            A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]LISTIM.OBJ            A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]INTRPM.OBJ            A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]INTRPS.OBJ            A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDMET.OBJ             A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDSPC.OBJ             A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]TIMEIM.OBJ            A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]VRSPIM.OBJ            A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRHDIM.OBJ            A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRIMET.OBJ            A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ             A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ             A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CLOCK1.OBJ            A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CLOCK2.OBJ            A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR3.OBJ            A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR5.OBJ            A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ERROUT.OBJ            A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ            A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSKIP1.OBJ            A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ            A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ             A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]INDEX1.OBJ            A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ              A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ            A25
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ            A26
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE3.OBJ            A27
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ            A28
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ             A29
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ            A30
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ             A31
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ            A32
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ            A33
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD1.OBJ            A34
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD2.OBJ            A35
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ             A36
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ             A37
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRCHAR.OBJ            A38
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFILE.OBJ            A39
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P39GV1A.EXE -
  /MAP = MET1:[S01S]P39GV1A.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A37, -
  A38, -
  A39
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P39GV1A.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P39GX1.COM follows.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM in USER2$DISK:[FOREMAN.FV1883A]P39GX1.COM
$ SHOW DEVICE / MOUNT
$ ASSIGN MLAB: SYS$PRINT
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                 PFMFDIR
$ ASSIGN MET51:[ROMNET1.D88187]MF110A.DAT         MF110
$ ASSIGN MET51:[ROMNET1.D88187]MF111A.DAT         MF111
$ ASSIGN MET51:[ROMNET1.D88187]MF129A.DAT         MF129
$ ASSIGN MET51:[ROMNET1.D88187]MF132A.DAT         MF132
$ ASSIGN MET51:[ROMNET1.D88187]MF133A.DAT         MF133
$ ASSIGN MET51:[ROMNET1.D88187]MF134A.DAT         MF134
$ ASSIGN MET51:[ROMNET1.D88187]MF135A.DAT         MF135
$ ASSIGN MET51:[ROMNET1.D88187]MF136A.DAT         MF136
$ ASSIGN MET51:[ROMNET1.D88187]MF137A.DAT         MF137
$ ASSIGN MET51:[ROMNET1.D88187]MF138A.DAT         MF138
$ ASSIGN MET51:[ROMNET1.D88187]MF139A.DAT         MF139
$ ASSIGN MET51:[ROMNET1.D88187]MF140A.DAT         MF140
$ ASSIGN MET51:[ROMNET1.D88187]MF146A.DAT         MF146
$ ASSIGN MET51:[ROMNET1.D88187]MF147A.DAT         MF147
$ ASSIGN MET51:[ROMNET1.D88187]MF151A.DAT         MF151
$ ASSIGN MET51:[ROMNET1.D88187]MF152A.DAT         MF152
$ ASSIGN MET51:[ROMNET1.D88187]MF153A.DAT         MF153
$ ASSIGN MET51:[ROMNET1.D88187]MF154A.DAT         MF154
$ ASSIGN MET51:[ROMNET1.D88187]MF155A.DAT         MF155
$ ASSIGN MET51:[ROMNET1.D88187]MF156A.DAT         MF156
$ ASSIGN MET51:[ROMNET1.D88187]MF157A.DAT         MF157
$ ASSIGN MET51:[ROMNET1.D88187]MF160A.DAT         MF160
$ ASSIGN MET51:[ROMNET1.D88187]MF161A.DAT         MF161
$ ASSIGN MET51:[ROMNET1.D88187]MF162A.DAT         MF162
$ ASSIGN MET51:[ROMNET1.D88187]MF163A.DAT         MF163
$ ASSIGN MET51:[ROMNET1.D88187]MF164A.DAT         MF164
$ ASSIGN MET51:[ROMNET1.D88187]MF165A.DAT         MF165
$ ASSIGN MET51:[ROMNET1.D88187]MF166A.DAT         MF166
$ ASSIGN MET51:[ROMNET1.D88187]MF167A.DAT         MF167
$ ASSIGN MET51:[ROMNET1.D88187]MF174A.DAT         MF174
$ ASSIGN MET51:[ROMNET1.D88187]MF175A.DAT         MF175
$ ASSIGN MET51:[ROMNET1.D88187]MF176A.DAT         MF176
$ !
$ ASSIGN MET58:[BMATRIX]METFV1883A.DAT            INTRMET
$ ASSIGN MET58:[BMATRIX]METFV1883A.PRF            PRFILE
$ SHOW PROCESS / ALL
$ ASSIGN USER2$DISK:[FOREMAN.FV1883A]P39GX1.CPR   SYS$INPUT
$ TYPE SYS$INPUT
$ ASSIGN $2$DUA24:[ROMEXE.ROMNET]P39GV1A.EXE      PROCP39G
$ DIRECTORY / FULL PROCP39G
$ RUN PROCP39G
$ SHOW PROCESS / ALL
$ PRINT / DEVICE=MLAB: / NOTE = BIN_12            PRFILE
$ SET FILE / NOBACKUP / PROTECTION = G:RWED       INTRMET
$ SHOW STATUS
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P39GV1A

B. Subroutines

LISTIM	RDMET	VRSPIM
INTRPM	RDSPC	WRHDIM
INTRPS	TIMEIM	WRIMET

C. Functions

None

D. Block Data Files

BLKIMT

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	DATHR5	IBLKR	PROGID	RDFT5
CELL0	ERROUT	JFILE1	RDDIR1	RDREC
CLOCK1	FLSMRY	OPDIR2	RDFHD1	WRCHAR
CLOCK2	FSTAT1	OPEN2	RDFHD2	WRFILE

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

DATHR3	JFILE3
INDEX1	JHOUR1
IOCL	JUNIT
JFILE2	

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

HEADIM.EXT	IMTPRM.EXT	RTSHIM.EXT
IMLUNS.EXT	METBUF.EXT	SPCBUF.EXT
IMMFLS.EXT	METVRS.EXT	SPCVAR.EXT

DISK$VAXSET:[ROMLIB.UTILIO]

DATIME.EXT
PRID.EXT

2.34 PROCESSOR P40G (B-MATRIX FILE CREATION)

I. PROCESSOR FUNCTION

This stage 8 processor produces the B-matrix file from the model input files that contain meteorology and emissions data. The B-matrix file contains (1) the interlayer mass-flux coefficients (the *B-matrix coefficients*) for the parameterization of the vertical fluxes across the surfaces of the three model layers, (2) the parameterizations for the emissions sources, and (3) the meteorology data for the Core Model, required by the Core Model to make the necessary rate-constant adjustments in the Carbon Bond 4.2 chemical mechanism for local air temperature, atmospheric density, water vapor concentration, and altitude.

P40G calculates the B-matrix coefficients, and interpolates these and the meteorology data from the model input files' hourly data to the 1800-second time step of the Core Model. The emissions data are not interpolated, but the hourly data are recorded for each time step on the hour and repeated once for each subsequent model time step.

The B-matrix file requires a large amount of disk space. P40G allows you the option of writing this file to several smaller subfiles on different disk packs -- each of which may not individually have sufficient space to contain the entire file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

MF123 from P10G
MF143 from P10G
MF144 from P10G
MF145 from P10G

2. Nonstandard Input Files

P40G uses one nonstandard input file, IMET, which is generated by processor P39G. IMET contains the meteorology data for the model's chemistry rate-constant adjustments as well as the deposition velocity and C_∞ values for some of the model's 35 chemical species. All data on the IMET file have been interpolated from hourly to 1800-second time steps using a scheme that weights (1) each hour's data by 75% and the subsequent hour's data by 25% for the first 1800-s time step, and (2) each hour's data by 25% and the subsequent hour's data by 75% for the second 1800-s time step. Although the IMET file is not strictly a ROM Core Model file, it nevertheless conforms to the requirements for all ROM Core Model files, and therefore contains a file header and time-step header structure. Descriptions of the records containing this information are given below.

The first call to subroutine RDIM opens the IMET file and reads and checks the file header information:

```
      SUBROUTINE RDIM (NWORDS, DATA, IOST)
C
C opens and reads the IMET file
C
C input arguments:
C  NWORDS:    number of words in record to be read
C
C output arguments:
C  DATA:      buffer area which will receive record
C  IOST:      status of record written to intermediate file
C
C read 1st header record
C
      READ (IIMET, IOSTAT = IOST) RECONE
C convert character to mixed character & numeric
C
      READ (RECONE, 1001, IOSTAT = IOST)
     &      CDATIM, CTIMIM, SDATIM, STHRIM, TSTPIM, FRSTIM,
     &      GRDNIM, SWLNIM, SWLTIM, NELNIM, NELTIM, DLONIM,
     &      DLATIM, NCOLIM, NROWIM, NLEVIM, NSPCIM, NMIFIM,
     &      ICNTIM
1001  FORMAT(6I8.8, A8, 4F8.3, 2F8.5, 6I4.4)
C read species names record
      READ (IIMET, IOSTAT = IOST) (SPNMIM(ISPC), ISPC = 1, NSPCIM)
C
C read dep vel index record
      READ (IIMET, IOSTAT = IOST) (DEPVNX(ISPC), ISPC = 1, NSPCIM)
C
C read cinf index record
      READ (IIMET, IOSTAT = IOST) (CINFNX(ISPC), ISPC = 1, NSPCIM)
C
C read the level names record
      READ (IIMET, IOSTAT = IOST) (LVNMIM(ILEV), ILEV = 1, NLEVIM)
C
C read the MIF data records
      DO 301 IMIF = 1, NMIFIM
      READ (IIMET, IOSTAT = IOST) RECMIF
C
      READ (RECMIF, 1005, IOSTAT = IOST)
     &      MFNMIM(IMIF), CDMFIM(IMIF), CTMFIM(IMIF),
     &      UDMFIM(IMIF), UTMFIM(IMIF)
1005  FORMAT(A12, 4I8.8)
301   CONTINUE
C
C read header text records
      DO 501 ITXT = 1, ICNTIM
      READ (IIMET, IOSTAT = IOST) TBUF
C
      READ (TBUF, 1007, IOSTAT = IOST) TEXTIM(ITXT)
1007  FORMAT(A80)
501   CONTINUE
```

Subsequent calls to RDIM read either the time step header record or file body records:

```
C
C file body . . . . . . . . . . . . . . . . . . . . . . . .  record*
C
C read next record
     READ (IIMET, IOSTAT = IOST) DATA
```

where DATA is the NWORDS buffer to receive the record.

Rather than repeat the large amount of information contained in the IMET file headers and data body, as has been our convention in the rest of this document, we refer you to Tables 2.248 to 2.257.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

The B-matrix file contains data elements required by the Core Model for each grid cell, pollutant species, and time step. The structure of the B-matrix file conforms to the requirements for all ROM Core Model files, and therefore contains a file header and time-step header structure. Descriptions of the records containing this information are given below.

a. Record 1

The first record contains character strings of alphanumeric data that describe the file's contents. The data are prepared and written to a character buffer, which is then written unformatted to the B-matrix file by subroutine WRCHAR. The parameters of record 1 are shown in Table 2.260.

```
C first segment record . . . . . . . . . . . . . . . . . . . . .
      WRITE(RECONE, 1001, IOSTAT = IOST)
     &      CDATBM, CTIMBM, SDATBM, STHRBM, TSTPBM, FRSTBM,
     &      GRDNBM,
     &      SWLNBM, SWLTBM, NELNBM, NELTBM,
     &      DLONBM, DLATBM,
     &      NCOLBM, NROWBM, NLEVBM, NSPCBM, NMIFBM, ICNTBM
1001  FORMAT(6I8.8, A8, 4F8.3, 2F8.5, 6I4.4)
C
      CALL WRCHAR(IBMAT, RECONE, IOST)

      SUBROUTINE WRCHAR (IUNIT, CHBUF, IOST)
      IMPLICIT NONE
      INTEGER*4 IUNIT, IOST
      CHARACTER*(*) CHBUF
      WRITE(IUNIT, IOSTAT = IOST) CHBUF
      RETURN
      END
```

TABLE 2.260. B-MATRIX RECORD 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CDATBM		Integer*4	File creation date as *MMDDYY*
2	CTIMBM	EST	Integer*4	File creation time as *HHMMSS*
3	SDATBM		Integer*4	Julian start date of scenario as *YYDDD*
4	STHRBM	EST	Integer*4	Start hour of scenario (00 to 23)
5	TSTPBM	s	Integer*4	Time step size for simulation
6	FRSTBM	s	Integer*4	Time to first step
7	GRDNBM		Char*8	Grid definition name
8	SWLNBM	° W	Real*4	Longitude of southwest corner of grid
9	SWLTBM	° N	Real*4	Latitude of southwest corner of grid
10	NELNBM	° W	Real*4	Longitude of northeast corner of grid
11	NELTBM	° N	Real*4	Latitude of northeast corner of grid
12	DLONBM	° W	Real*4	Grid cell longitudinal increment
13	DLATBM	° N	Real*4	Grid cell latitudinal increment
14	NCOLBM		Integer*4	Number of columns in grid
15	NROWBM		Integer*4	Number of rows in grid
16	NLEVBM		Integer*4	Number of levels
17	NSPCBM		Integer*4	Number of model species
18	NMIFBM		Integer*4	Number of model input files used to generate the B-matrix file
19	ICNTBM		Integer*4	Number of text records

b. Record 2

This record contains the list of species names for which the Core Model computes concentration outputs. As for record 1 above, the data are written to a character buffer, which is then written unformatted to the B-matrix file. The parameter of record 2 is shown in Table 2.261.

```
C species names record . . . . . . . . . . . . . . . . . . . . . . . . .
. .
      WRITE(RECSPN, 1003, IOSTAT = IOST) (SPNMBM(ISPC), ISPC = 1, NSPCBM)
1003  FORMAT(<NSPCBM>A4)
      CALL WRCHAR(IBMAT, RECSPN, IOST)
```

TABLE 2.261. B-MATRIX RECORD 2 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	SPNMBM(I)		Char*4	Name of species *I*

c. Record 3

This record contains the expansion list (look-up table) of species names for which the B-matrix file has values. (Refer to PROCESSOR P23G for an explanation of the need for an expansion list.) These values are mapped to the full core-model list of species names. The WRITE and FORMAT statements for this record are listed below, and its parameter is shown in Table 2.262.

```
      WRITE(RECSPN, 1005, IOSTAT = IOST) (BMINDX(ISPC), ISPC = 1, NSPCBM)
1005  FORMAT(<NSPCBM>I4)
      CALL WRCHAR(IBMAT, RECSPN, IOST)
```

TABLE 2.262. B-MATRIX RECORD 3 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	BMINDX(K)		Integer*4	Index array for species expansion from the reduced B-matrix set to the full model set

d. Record 4

This record contains the list of Core Model level names. The WRITE and FORMAT statements for this record are listed below, and its parameter is shown in Table 2.263.

```
C level names record . . . . . . . . . . . . . . . . . . . . .
      WRITE(RECLVN, 1007, IOSTAT = IOST) (LVNMBM(ILEV), ILEV = 1, NLEVS)
1007  FORMAT(<NLEVS>A4)
      CALL WRCHAR (IBMAT, RECLVN, IOST)
```

TABLE 2.263. B-MATRIX RECORD 4 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	LVNMBM(J)		Char*4	Name of level *J*

e. Records 5 - 8

These records contain the list of logical names of the model input files (MIF) that are used to generate the B-matrix data. The WRITE and FORMAT statements for these records are listed below, and its parameters are shown in Table 2.264.

```
C MIF segment records . . . . . . . . . . . . . . . . . . . . .
      DO 401 IMIF = 1, NMIFBM
      WRITE(RECMIF, 1009, IOSTAT = IOST) MFNMBM(IMIF), CDMFBM(IMIF),
     &                                   CTMFBM(IMIF), UDMFBM(IMIF),
     &                                   UTMFBM(IMIF)
1009  FORMAT(A12, 4I8.8)
      CALL WRCHAR(IBMAT, RECMIF, IOST)
401   CONTINUE
```

TABLE 2.264. B-MATRIX RECORDS 5 - 8 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	MFNMBM(J)		Char*12	File name of MIF file *J*
2	CDMFBM(J)		Integer*4	Julian creation date (*YYDDD*) of MIF file *J*
3	CTMFBM(J)	EST	Integer*4	Creation time (*HH*) of MIF file *J*
4	UDMFBM(J)		Integer*4	Last update Julian date of MIF file *J*
5	UTMFBM(J)	EST	Integer*4	Last update time of MIF file *J*

f. Records 9 - (28 maximum)

These records contain descriptive text that you can optionally enter as part of the control cards data set. One 80-character string is written to each record by means of the following statements. A blank string *must* denote the end of the descriptive text records. The WRITE and FORMAT statements for these records are listed below, and their parameter is shown in Table 2.265.

```
C text segment records . . . . . . . . . . . . . . . . . . . . .
      DO 601 ITXT = 1, ICNTBM
      WRITE(RECTXT, 1011, IOSTAT = IOST) TEXTBM(ITXT)
1011  FORMAT(A80)
      CALL WRCHAR(IBMAT, RECTXT, IOST)
601   CONTINUE
```

TABLE 2.265. RECORDS 9 - (9+ICNTBM-1) PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	TEXTBM(ITXT)		Char*80	Text description

g. Subfile-Order Record

The purpose of this record is to ensure the correct ordering of input data when the Core Model reads several B-matrix subfiles. The WRITE and FORMAT statements for this record are listed below, and its parameters are shown in Table 2.266.

```
C file order record . . . . . . . . . . . . . . . . . . . . . .
      WRITE(FLCREC, 1013) SUBFL, NSUBFL, FRSTEP, LSSTEP
1013  FORMAT(4I4)
      CALL WRCHAR(IBMAT, FLCREC, IOST)
```

TABLE 2.266. SUBFILE-ORDER RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	SUBFL		Integer*4	Ordinal for this subfile
2	NSUBFL		Integer*4	Total number of subfiles for this B-matrix file
3	FRSTEP		Integer*4	First time step count for this subfile
4	LSSTEP		Integer*4	Last time step count for this subfile

These end the file header records. The following are the data records, headed by one time-step record for each time interval.

h. Time-Step Header Record

There is one time-step header record written for each time step increment on the B-matrix file. The time-step header data are contained in a common block. The following statement writes the data to the BMAT file by referencing the first variable of the common block, and specifying the number of the block's words to be written. The data are then written to the file by subroutine WRFILE. The WRITE and FORMAT statements for this record are listed below, and its parameters are shown in Table 2.267.

```
      COMMON / RTSHBM / DATBM, TIMBM, ELPBM, STPBM
      NWRDST = 4
      CALL WRFILE(IBMAT, NWRDST, DATBM, IOST)

      SUBROUTINE WRFILE (IUNIT, NWORDS, BUFFER, IOST)
      IMPLICIT NONE
      INTEGER*4 IUNIT, NWORDS, IOST
      REAL*4 BUFFER (NWORDS)
      WRITE(IUNIT, IOSTAT = IOST) BUFFER
      RETURN
      END
```

TABLE 2.267. TIME-STEP HEADER RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATBM		Integer*4	Julian current time step date as *YYDDD*
2	TIMBM	EST	Integer*4	Current time step time as *HHMMSS*
3	ELPBM	s	Integer*4	Elapsed time since scenario start
4	STPBM		Integer*4	Current time-step no. on B-matrix file

i. Data Records

There are two groups of data records for each B-matrix time step increment. The first group (GROUP 1 B-MATRIX PARAMETERS, Table 2.268) contains one record of variables for each model domain row. Some variables are only primary-oxidant-species dependent; the others are species independent (e.g., meteorological conditions). The second group (GROUP 2 B-MATRIX PARAMETERS, Table 2.269) contains a data record for each model domain row that consists of concentrations of species for which the B-matrix file has values.

The data records are written by equivalencing common blocks of row data for each group. The following parameters are used for dimensioning the common blocks and controlling the write operations:

NVRBM1 = 18 + 3 * NLEVS + 6 * NPOXSP, where NLEVS = 3 and NPOXSP = 3, the number of primary oxidant species, i.e., NO, NO_2, and O_3;

NVRBM1 is the number of Part 1 B-matrix variables, specifically: B12, B13, B21, B23, B32, B33, B11S, B11SS, B31S, B31SS, QO3FAC, SSONO, TTHETA, PPSI2, ZZ0, ZZ1, ZZ2, ZZ3; RRHO(NLEVS)'s, TTEMP(NLEVS)'s, WWVC(NLEVS)'s; and G1S, G1SS, G1FAC, G3S, G3SS, G3FAC, each dimensioned by NPOXSP;

NVRBM2 = 6;

NVRBM2 is the number of species-array B-matrix variables, (Part 2) specifically: B11, B22, B31, G1, G2, G3.

The first group (Part 1) uses a named common block, BM1VAR, for the specific variables, and the same common block for a buffer, BM1, that is written unformatted to the B-matrix file by subroutine WRFILE:

```
      COMMON / BM1VAR /
     &        B12(NCOLS1), B13(NCOLS1),
     &        B21(NCOLS1), B23(NCOLS1),
     &        B32(NCOLS1), B33(NCOLS1),
     &        B11S(NCOLS1), B11SS(NCOLS1),
     &        B31S(NCOLS1), B31SS(NCOLS1),
     &        QO3FAC(NCOLS1), SSONO(NCOLS1),
     &        G1S(NCOLS1, NPOXSP), G1SS(NCOLS1, NPOXSP),
     &        G1FAC(NCOLS1, NPOXSP),
     &        G3S(NCOLS1, NPOXSP), G3SS(NCOLS1, NPOXSP),
     &        G3FAC(NCOLS1, NPOXSP),
     &        RRHO1(NCOLS1), RRHO2(NCOLS1), RRHO3(NCOLS1),
     &        TTEMP1(NCOLS1), TTEMP2(NCOLS1), TTEMP3(NCOLS1),
     &        WWVC1(NCOLS1), WWVC2(NCOLS1), WWVC3(NCOLS1),
```

```
     &        TTHETA(NCOLS1), PPSI2(NCOLS1),
     &        ZZ0(NCOLS1), ZZ1(NCOLS1),
     &        ZZ2(NCOLS1), ZZ3(NCOLS1)

      COMMON / BM1VAR / BM1(NCOLS1, NVRBM1)
```

The second group (Part 2) uses a named common block, BM2VAR, for the specific variables, and the same common block for a buffer, BM2, which is written unformatted to the B-matrix file by means of routine WRFILE:

```
      COMMON /BM2VAR / B11(NCOLS1, NSPECS),
     &                 B22(NCOLS1, NSPECS),
     &                 B31(NCOLS1, NSPECS),
     &                 G1(NCOLS1, NSPECS),
     &                 G2(NCOLS1, NSPECS),
     &                 G3(NCOLS1, NSPECS)

      COMMON / BM2VAR / BM2(NCOLS1, NSPECS, NVRBM2)
```

The procedure to write these data is as follows:

```
      REAL*4 BUF(NCOLS1, NVRBM2)
      NWRDS1 = NVRBM1 * NCOLS1
      NWRDS2 = NVRBM2 * NCOLS1

      IROW = 1
301   CONTINUE
      IF (IROW .GT.  NROWS1) GO TO 601
C
C write block 1 for the row
      CALL WRFILE (IBMAT, NWRDS1, BM1(1,1), IOST)
C
C row body . . . . . . . . . . . . . . . . . . species-reduced-list*
C
C write block 2 for the row
      DO 501 ISPC = 1, BMSPRD
C
      DO 401 IPRM = 1, NVRBM2
      DO 401 ICOL = 1, NCOLS1
C transfer and reorder data from block 2 buffer
      BUF(ICOL, IPRM) = BM2(ICOL, ISPC, IPRM)
401   CONTINUE
      CALL WRFILE (IBMAT, NWRDS2, BUF, IOST)
C
501   CONTINUE
C
C row body . . . . . . . . . . . . . . . . . . . . . . . . end
C
      IROW = IROW + 1
      GO TO 301
C
601   CONTINUE
```

where BMSPRD = 27, the number of species in the B-matrix reduced set that represents all the model species.

TABLE 2.268. B-MATRIX PART 1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				B-matrix coefficient for:
1	B12	s^{-1}	Real*4	layer-1/surface-2 flux
2	B13	s^{-1}	Real*4	layer-1/surface-3 flux
3	B21	s^{-1}	Real*4	layer-2/surface-1 flux
4	B23	s^{-1}	Real*4	layer-2/surface-3 flux
5	B32	s^{-1}	Real*4	layer-3/surface-2 flux
6	B33	s^{-1}	Real*4	layer-3/surface-3 flux
				B-matrix coefficient for subgrid scale adjustment for:
7	B11S	s^{-1}	Real*4	layer-1/surface-1 flux
8	B11SS	s^{-1}	Real*4	alternate layer-1/surface-1 flux
9	B31S	s^{-1}	Real*4	layer-3/surface-1 flux
10	B31SS	s^{-1}	Real*4	alternate layer-3/surface-1 flux
				Run time subgrid scale adjustments:
11	QO3FAC	$m \cdot s^{-1}$	Real*4	ozone factor
12	SSONO	$ppm \cdot m \cdot s^{-1}$	Real*4	NO surface emissions source strength
13	G1S	$ppm \cdot s^{-1}$	Real*4	emissions source term in layer 1
14	G1SS	$ppm \cdot s^{-1}$	Real*4	alternate emissions source term in layer 1
15	G1FAC	s^{-1}	Real*4	emissions source factor in layer 1
16	G3S	$ppm \cdot s^{-1}$	Real*4	emissions source term in layer 3
17	G3SS	$ppm \cdot s^{-1}$	Real*4	alternate emissions source term in layer 3
18	G3FAC	s^{-1}	Real*4	emissions source factor in layer 3
				Rate constants density correction factor for:
19	RRHO1	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 1
20	RRHO2	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 2
21	RRHO3	$mol \cdot ppm \cdot m^{-3}$	Real*4	layer 3
				Absolute temperature for rate constants adjustment for:
22	TTEMP1	K	Real*4	layer 1
23	TTEMP2	K	Real*4	layer 2
24	TTEMP3	K	Real*4	layer 3
				Water vapor concentration for rate constants adjustment for:
25	WWVC1	ppm	Real*4	layer 1
26	WWVC2	ppm	Real*4	layer 2
27	WWVC3	ppm	Real*4	layer 3
28	TTHETA	deg	Real*4	Solar zenith angle for photolytic rate constants adjustment
29	PPSI2		Real*4	Cloud cover correction factor for photolytic rate constants adjustment
				Heights above sea level (used for rate constant adjustments):
30	ZZ0	m	Real*4	layer 0
31	ZZ1	m	Real*4	layer 1
32	ZZ2	m	Real*4	layer 2
33	ZZ3	m	Real*4	layer 3

TABLE 2.269. PART 2 B-MATRIX PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
				B-matrix coefficient for:
34	B11	s^{-1}	Real*4	layer-1/surface-1 flux
35	B22	s^{-1}	Real*4	layer-2/surface-2 flux
36	B31	s^{-1}	Real*4	layer-3/surface-1 flux
				Emissions source term for:
37	G1	$ppm \cdot s^{-1}$	Real*4	layer 1
38	G2	$ppm \cdot s^{-1}$	Real*4	layer 2
39	G3	$ppm \cdot s^{-1}$	Real*4	layer 3

III. CONTROL CARDS

Three control cards are used to input the control data using the code shown below. Following these control cards, you can optionally enter file-description text in as many as 20 80-character records. This text is subsequently written to the B-matrix file's header. Note that one blank text record is required at the end of these character records, even if you opt to enter no text. Table 2.270 defines the control card variables.

You will see that dummy records are read into a character variable, CLABEL. These records exist in the input stream to describe what the following card of data represents.

The control cards are input using the following format:

SDATBM STHRBM SSDATE SSHOUR
NSUBFL FLSIZE(I)
VAXFMT
TEXTBM(K)

The code that reads the control card information follows.

```
C read control cards
      READ(5, 1003) CLABEL
1003  FORMAT(A80)
      READ(5, *) SDATBM, STHRBM, SSDATE, SSHOUR
      SSTIME = SSHOUR * 10000
      READ(5, 1003) CLABEL
      READ(5, *) NSUBFL, (FLSIZE(IFL), IFL = 1, NSUBFL)
      READ(5, 1003) CLABEL
      READ(5, *) VAXFMT
C
      DO 101 ITXT = 1, NTEXT
      READ(5, 1003, IOSTAT = IOST) TEXTBM(ITXT)
      IF (TEXTBM(ITXT) .EQ.  BLANK) GO TO 111
101   CONTINUE
111   CONTINUE
      ICNTBM = ITXT - 1
```

TABLE 2.270. CONTROL CARD VARIABLES

Variable Name	Unit	Description
SDATBM		Julian scenario start date as *YYDDD*
STHRBM	EST	Scenario start hour as *HH*
SSDATE		Julian simulation start date as *YYDDD*
SSHOUR	EST	Simulation start hour as *HH*
NSUBFL		Number of subfiles to be generated
FLSIZE(I)		Number of time steps to be written to subfile *I*
VAXFMT		Translate flag: .TRUE. ⇒ do not translate to IBM binary .FALSE. ⇒ translate to IBM binary
TEXTBM(K)		Descriptive text records to be recorded on the B-matrix file header

Example:

```
88187 12 88187 12
1 144
.FALSE.
          ...  RURAL OZONE MULTIPLE BMATRIX FILE ...
                for ROM2.1 on the ROMNET grid
                ----------------------
      generated from formulation FV1881A for 144   steps
                ----------------------
      using  USER2$DISK:<FOREMAN.FV1883B>P40GX1.COM
             USER2$DISK:<FOREMAN.FV1883B>P40GX1.CPR
             DISK$VAXSET:<ROMEXE.ROMNET>P40GV1B.EXE
                                                  25 JAN 90
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
                                              {blank line of text}
```

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	15	files	165	blocks
FORTRAN INCLUDE files:	21	files	72	blocks
Object files:	15	files	126	blocks
Executable file:	1	file	114	blocks
	52	files	477	blocks

B. Execution Time Requirements (Representative Values for a 72-Hour Scenario):

	VAX 8650
Charged CPU time (hh:mm:ss):	01:20:32
Buffered I/O count:	383
Direct I/O count:	45355
Peak working-set size:	3325
Peak virtual size:	6233

C. Space Requirements: Log and Print Files

P40GX1.LOG:	84 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.271 shows the input file and output file space requirements.

TABLE 2.271. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	MF123	5	1899	72 hours
	MF143	5	26234	72 hours
	MF144	5	26234	72 hours
	MF145	5	26234	72 hours
	IMET	Nonstd	203124	72 hours
			283725	
Output	B-MATRIX	Nonstd	754000	72 hours

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P40GV1B.LNK follows.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE DISK$VAXSET:[ROMCOM.ROMNET]P40GV1B.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P40GV1B.MAP"
$ EXEFILE = "DISK$VAXSET:[ROMEXE.ROMNET]P40GV1B.EXE"
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]P40G.OBJ          A1
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]BLKBMT.OBJ        A2
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]BMCOEF.OBJ        A3
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]OPIMET.OBJ        A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]POIMET.OBJ        A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDEMIS.OBJ        A6
```

```
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDIM.OBJ          A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]RDIMET.OBJ        A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]SPCEXP.OBJ        A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]TIMEBM.OBJ        A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]TRNSEM.OBJ        A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]VERMET.OBJ        A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]VRSPBM.OBJ        A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRBMAT.OBJ        A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.ROMNET]WRHDBM.OBJ        A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ         A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CELLO.OBJ         A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CLOCK1.OBJ        A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]CLOCK2.OBJ        A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR3.OBJ        A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]DATHR5.OBJ        A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]ERROUT.OBJ        A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FLSMRY.OBJ        A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]FSTAT1.OBJ        A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IBLKR.OBJ         A25
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]INDEX1.OBJ        A26
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ          A27
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE1.OBJ        A28
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ        A29
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JFILE6.OBJ        A30
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JHOUR1.OBJ        A31
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ         A32
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPDIR2.OBJ        A33
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]OPEN2.OBJ         A34
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]PROGID.OBJ        A35
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDDIR1.OBJ        A36
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD1.OBJ        A37
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFHD2.OBJ        A38
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDFT5.OBJ         A39
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]RDREC.OBJ         A40
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]TRNSLT.OBJ        A41
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRCHAR.OBJ        A42
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMOBJ.UTILIO]WRFILE.OBJ        A43
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[ROMEXE.ROMNET]P40GV1B.EXE -
  /MAP = MET1:[S01S]P40GV1B.MAP -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A41, -
  A42, -
  A43
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P40GV1B.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P40GX1.COM follows.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1503A]P40GX1.COM
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                 PFMFDIR
$ ASSIGN MET58:[ROMNET1.D85188]MF123A.DAT         MF123
$ ASSIGN MET58:[ROMNET1.D85188]MF143A.DAT         MF143
```

```
$ ASSIGN MET58:[ROMNET1.D85188]MF144A.DAT          MF144
$ ASSIGN MET58:[ROMNET1.D85188]MF145A.DAT          MF145
$ ASSIGN MET58:[BMATRIX]IMETFV1503A.DAT            INTRMET
$ ! ASSIGN OUTPUT FILES
$ ASSIGN -
NCCIBM::"""ADRNER1.NOBKUP.BMAT.FV1503A/NOTRANSLATE/SINGLE/SUPERSEDE""" -
BMAT_01
$ !SYNCHRONIZE/ENTRY=530
$ SHOW PROCESS / ACCOUNTING
$ !
$ TYPE USER2$DISK:[FOREMAN.FV1503A]P40GX1.CPR
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN USER2$DISK:[FOREMAN.FV1503A]P40GX1.CPR      SYS$INPUT
$ !
$ ASSIGN DISK$VAXSET:[ROMEXE.ROMNET]P40GV1B.EXE      PROCP40G
$ DIRECTORY / FULL PROCP40G
$ RUN PROCP40G
$ SHOW PROCESS / ACCOUNTING
$ SHOW STATUS
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P40GV1B

B. Subroutines

BMCOEF	RDIM	TIMEBM	VRSPBM
OPIMET	RDIMET	TRNSEM	WRBMAT
POIMET	SPCEXP	VERMET	WRHDBM
RDEMIS			

C. Functions

None

D. Block Data Files

BLKBMT

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	FLSMRY	OPEN2	RDFT5
CELL0	FSTAT1	PROGID	RDREC
CLOCK1	IBLKR	RDDIR1	TRNSLT
CLOCK2	JFILE1	RDFHD1	WRCHAR
DATHR5	OPDIR2	RDFHD2	WRFILE
ERROUT			

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

DATHR3	JFILE6
INDEX1	JHOUR1
IOCL	JUNIT
JFILE2	

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

HEADIM.EXT	BMLUNS.EXT	BM1VAR.EXT	BM2BUF.EXT
RTSHIM.EXT	BMSPEC.EXT	BM2VAR.EXT	TSHDIM.EXT
BMTPRM.EXT	IMFILE.EXT	ABGCHK.EXT	IMTIME.EXT
BMMFLS.EXT	EMISVR.EXT	BM1BUF.EXT	

DISK$VAXSET:[ROMLIB.UTILIO]

DATIME.EXT
PRID.EXT

DISK$VAXSET:[ROMLIB.CORE]

HEADBM.EXT
HDFMTS.EXT

2.35 PROCESSOR P41G (POINT-SOURCE EMISSIONS CONTROL STRATEGY)

I. PROCESSOR FUNCTION

This stage 0 processor applies a specific control strategy to NO_x and VOC species for point-source emissions.[15]

Control strategies are reductions to the base-case emissions files. The control strategies reduce the emissions of certain species by a designated percentage. The reductions can vary between 0% and 99%, but are usually between 5% and 99%.

Note that this processor is used only on raw point-source emissions data, and that all point sources are usually not affected by any one control strategy.

15. VOC species input to the ROM are: olefins, paraffins, toluene, xylene, formaldehyde, higher aldehydes, ethene, and isoprenes.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

We adopted a convention to define control strategy input files that appends a suffix of .CS*x* to the files. For example, .CS0 files indicate no control strategy (the base case), while .CS1 and .CS2 indicate control strategies numbers 1 and 2 respectively.

a. SEANPT.CS0

SEANPT.CS0 is the seasonal annual point-source emissions file containing 15 chemical species; it is produced by preprocessor PTEXTR. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 2.272.

```
      READ(UNITR, 1003, IOSTAT = IOSTR)
     &     XSTKID,
     &     NSPCIN, (CLASS(ISPC), ANEM(ISPC), ISPC = 1, NSPCIN),
     &     MXDYIN, (S_AFAC(IDAY), D_SFAC(IDAY),
     &             (H_DFAC(IHR, IDAY), IHR = 1, 24), IDAY = 1, MXDYIN)
1003  FORMAT(A20,
     &      I3, <NSPECS>(A5, E12.5),
     &      I2, <MXDAY>(F5.3, F6.4, 24F6.0))
```

b. NOXPTCTRL.CS*x*

This file contains the NO_x control strategy; it originates from the Office of Air Quality Planning and Standards, the U.S. Environmental Protection Agency, Research Triangle Park, NC. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 2.273.

```
        READ(CUNIT, 1001, IOSTAT = IOST)
     &                           CTRLST_NOX(ISTK), NOXSPF(ISTK)
1001    FORMAT(1X, A12, 11X, F6.2)
```

TABLE 2.272. SEANPT.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	XSTDID		Char*20	Stack ID
2	NSPCIN		Integer*4	Number of species
3	CLASS(1)		Char*5	AEROS code for VOC
4	ANEM(1)$_p$	tons·y^{-1}	Real*4	Annual emission rate, VOC
5	CLASS(2)		Char*5	AEROS code for CO
6	ANEM(2)$_p$	tons·y^{-1}	Real*4	Annual emission rate, CO
7	CLASS(3)		Char*5	AEROS code for olefins
8	ANEM(3)$_p$	tons·y^{-1}	Real*4	Annual emission rate, olefins
9	CLASS(4)		Char*5	AEROS code for paraffins
10	ANEM(4)$_p$	tons·y^{-1}	Real*4	Annual emission rate, paraffins
11	CLASS(5)		Char*5	AEROS code for toluene
12	ANEM(5)$_p$	tons·y^{-1}	Real*4	Annual emission rate, toluene
13	CLASS(6)		Char*5	AEROS code for xylene
14	ANEM(6)$_p$	tons·y^{-1}	Real*4	Annual emission rate, xylene
15	CLASS(7)		Char*5	AEROS code for formaldehyde
16	ANEM(7)$_p$	tons·y^{-1}	Real*4	Annual emission rate, formaldehyde
17	CLASS(8)		Char*5	AEROS code for aldehydes
18	ANEM(8)$_p$	tons·y^{-1}	Real*4	Annual emission rate, aldehydes
19	CLASS(9)		Char*5	AEROS code for ethene
20	ANEM(9)$_p$	tons·y^{-1}	Real*4	Annual emission rate, ethene
21	CLASS(10)		Char*5	AEROS code for nonr.
22	ANEM(10)$_p$	tons·y^{-1}	Real*4	Annual emission rate, nonr.
23	CLASS(11)		Char*5	AEROS code for isoprenes
24	ANEM(11)$_p$	tons·y^{-1}	Real*4	Annual emission rate, isoprenes
25	CLASS(12)		Char*5	AEROS code for methane
26	ANEM(12)$_p$	tons·y^{-1}	Real*4	Annual emission rate, methane
27	CLASS(13)		Char*5	AEROS code for NO
28	ANEM(13)$_p$	tons·y^{-1}	Real*4	Annual emission rate, NO
29	CLASS(14)		Char*5	AEROS code for NO_2
30	ANEM(14)$_p$	tons·y^{-1}	Real*4	Annual emission rate, NO_2
31	CLASS(15)		Char*5	AEROS code for methanol
32	ANEM(15)$_p$	tons·y^{-1}	Real*4	Annual emission rate, methanol
33	MXDAY		Integer*4	No. of day types per season
34	S_AFAC$_p$		Real*4	Seasonal allocation factor
35	D_SFAC$_p$		Real*4	Weekday allocation factor
36	H_DFAC(1)$_p$		Real*4	Hour 1 allocation factor
⋮	⋮		⋮	⋮
60	H_DFAC(24)$_p$		Real*4	Hour 24 allocation factor
61	S_AFAC$_p$		Real*4	Seasonal allocation factor
62	D_SFAC$_p$		Real*4	Saturday allocation factor
63	H_DFAC(1)$_p$		Real*4	Hour 1 allocation factor
⋮	⋮		⋮	⋮
87	H_DFAC(24)$_p$		Real*4	Hour 24 allocation factor
88	S_AFAC$_p$		Real*4	Seasonal allocation factor
89	D_SFAC$_p$		Real*4	Sunday allocation factor
90	H_DFAC(1)$_p$		Real*4	Hour 1 allocation factor
⋮	⋮		⋮	⋮
114	H_DFAC(24)$_p$		Real*4	Hour 24 allocation factor

Note: p = point source; nonr. = nonreactive hydrocarbons;. Units are English short tons. NO emissions are reported as weight of NO_2

TABLE 2.273. NOXPTCTRL.CSx PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CTRLST_NOX		Char*12	Stack ID
2	$NOXSPF_p$	%	Real*4	Percent emission reduction

Note: p = point source

c. VOCPTCTRL.CSx

This file contains the VOC control strategy; it originates from the Office of Air Quality Planning and Standards, the U.S. Environmental Protection Agency, Research Triangle Park, NC. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 2.274.

```
          READ(CUNIT, 1001, IOSTAT = IOST)
     &                           CTRLST_VOC(ISTK), VOCSPF(ISTK)
1001      FORMAT(1X, A12, 11X, F6.2)
```

TABLE 2.274. VOCPTCTRL.CSx PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CTRLST_VOC		Char*12	Stack ID
2	$VOCSPF_p$	%	Real*4	Percent emission reduction

Note: p = point source

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

P41G outputs one nonstandard file, SEANPT.CSx, which contains the same parameters as are in Table 2.272, with emissions values for NO_x and VOC modified by the applied control strategy. Note that x is never equal to 0 in the output files. The WRITE and FORMAT statements for this file are listed below.

```
      WRITE(UNITW, 1053, IOSTAT = IOSTW)
     &     XSTKID,
     &     NSPECS, (CLASS(ISPC), ANEM(ISPC), ISPC = 1, NSPECS),
     &     MXDAY, (S_AFAC(IDAY), D_SFAC(IDAY),
     &            (H_DFAC(IHR, IDAY), IHR = 1, 24), IDAY = 1, MXDAY)
 1053 FORMAT(A20,
     &      I3, <NSPECS>(A5, E12.5),
     &      I2, <MXDAY>(F5.3, F6.4, 24F6.4))
```

III. CONTROL CARDS

Three control cards are used to input control data in the format shown below. Table 2.275 defines the control card variables.

NOX_CS VOC_CS
DFLNOX
DFLVOC

TABLE 2.275. CONTROL CARD VARIABLES

Variable Name	Unit	Description
NOX_CS		NO_x flag: .TRUE. ⇒ control applied
VOC_CS		VOC flag: .TRUE. ⇒ control applied
DFLNOX	%	NO_x control strategy reduction
DFLVOC	%	VOC control strategy reduction

Example:

.TRUE. .TRUE.
30.0
30.0

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	22	blocks
FORTRAN INCLUDE files:	3	files	5	blocks
Object files:	2	files	14	blocks
Executable file:	1	file	19	blocks
	8	files	60	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:09:00
Buffered I/O count:	534
Direct I/O count:	4145
Peak working-set size:	663
Peak virtual size:	2587

C. Space Requirements: Log and Print Files

P41GX1.LOG:	11 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 2.276 shows the input file and output file space requirements.

TABLE 2.276. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SEANPT.CS0	Nonstd	26481	summer
	NOXPTCTRL.CSx	Nonstd	± 50	summer
	VOCPTCTRL.CSx	Nonstd	± 50	summer
			± 26581	
Output	SEANPT.CSx	Nonstd	26481	summer

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P41G.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER62$DISK:[XHS.P41G]P41G.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[S01S]P41G.MAP"
$ EXEFILE = "USER62$DISK:[XHS.P41G]P41G.EXE"
$ ASSIGN/USER_MODE  USER62$DISK:[XHS.P41G]P41G.OBJ                    A1
$ ASSIGN/USER_MODE  USER62$DISK:[XHS.P41G]RDPTCS.OBJ                  A2
$ ASSIGN/USER_MODE  DISK$VAXSET:[ROMOBJ.UTILIO]INDEX1.OBJ             A3
$ ASSIGN/USER_MODE  DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ               A4
$ ASSIGN/USER_MODE  DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ             A5
$ ASSIGN/USER_MODE  DISK$VAXSET:[ROMOBJ.UTILIO]JFILEX.OBJ             A6
$ ASSIGN/USER_MODE  DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ              A7
$ LINK -
  /EXECUTABLE = USER62$DISK:[XHS.P41G]P41G.EXE -
```

```
  /MAP = MET1:[S01S]P41G.MAP -
  A1, -
  A2, -
  A3, -
  A4, -
  A5, -
  A6, -
  A7
$ !PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[S01S]P41G.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

P41GX1.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ !
$ ! Run stream is in [VMQ.P41G]P41G.COM
$ !
$ ! Input files:
$ ASSIGN MET1:[EMISSIONS.NEONACTRL]NOXPTCTRL.CS4          PTCTLNOX
$ ASSIGN USER2$DISK:[VMQ.P41G]VOCPTCTRL2.CS1              PTCTLVOC
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]SEANPT.CSO             SEANLPT
$ !
$ ! Output file:
$ ASSIGN WORK_SCRATCH:SEANPT.CS1                          CTRLPT
$ !
$ TYPE USER62$DISK:[XHS.P41G]P41G.CPR
$ ASSIGN /USER USER62$DISK:[XHS.P41G]P41G.CPR             SYS$INPUT
$ !
$ RUN USER62$DISK:[XHS.P41G]P41G.EXE
$ !
$ DIRECTORY/FULL CONTROLPT
$ DIRECTORY/FULL SEASANLPT
```

(end of run stream)

VII. MAIN PROGRAM, SUBROUTINES, FUNCTIONS, AND BLOCK DATA REQUIRED

A. Main Program

P41G

B. Subroutines

RDPTCS

C. Functions

None

D. Block Data Files

None

VIII. I/O AND UTILITY LIBRARY SUBROUTINES AND FUNCTIONS REQUIRED

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO}

JFILEX

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

INDEX1
IOCL
JFILE2
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

NOXLIS.EXT
VOCLIS.EXT
CNTRLPT.EXT

This page is intentionally left blank.

SECTION 3

REGIONAL DEPENDENCIES AND THE VAX/VMS IMPLEMENTATION OF DOMAIN CHANGES IN THE REGIONAL OXIDANT MODEL'S PROCESSOR NETWORK AND ANALYSIS PROGRAMS

3.1 REGIONAL DEPENDENCIES

We offer a cautionary note to those of you reading this User's Guide who are thinking of implementing the ROM in a region other than the ones that we currently are using (NEROS, SEROS, and ROMNET). Regional dependencies are not simply what happens in the ROM FORTRAN code that enables the model to operate in a different domain, although the implementation of domain changes is part of the whole picture and is discussed in Section 3.2.

There are several issues that you must consider before you attempt to implement the ROM for your region. We list some of these below.

- The ROM will not function up to par in areas with sparse meteorological data, for example, the Great Plains, Great Basin, and Rocky Mountain states.
- High, mountainous terrain will have an unknown effect on the model, and is apt to produce erroneous results. We expect this inaccuracy because highly complex terrain requires a fully three-dimensional model; the ROM essentially consists of three interlinked two-dimensional models.
- A hot, dry climate (such as summer in the Northern and Southern Plains states, California, etc.) experiences the same general boundary layer meteorology as do the more humid climates for which the ROM was designed, but potentially on a very different temporal scale. For example, the growth of the boundary layer in the morning (after a night with a shallow surface-based temperature inversion) interacts with the other ROM layers and with emissions sources; by entraining air from layers above it to supply this growth, the chemical concentrations in the boundary layer will clearly be diluted. For example, we infer that the boundary layer meteorology is significantly different in Austin, Texas than in Burlington, Vermont. The question that you face (and must answer before you attempt to implement the ROM for your region) is exactly how the region interacts with the ROM layers -- and is the ROM adequately modeling these effects? We believe that it is likely that the model will require some scientific upgrades to handle these types of situations.

- If your region demands a larger grid size than ROMNET (64 x 52), are your hardware storage capacities adequate to handle the larger volume of data? Keep in mind that only one processor (P40G) is capable of splitting an output file between more than a single disk pack. With more data, data transfer to an IBM becomes a more tedious process -- and one inherently more prone to error.

3.2 THE VAX/VMS IMPLEMENTATION OF DOMAIN CHANGES

The Regional Oxidant Model's FORTRAN code is written to isolate the model's domain and grid dependencies in only three places: (1) the INCLUDE files REGPRM.EXT and REGION.EXT, (2) the utility I/O routine CELL0, and (3) in the names and header records of PF/MF files. Generally, REGPRM.EXT contains compile-time dimensioning and physical constants; routine CELL0 supplies geographically-based information such as grid dimensions and grid-cell sizes. PF/MF file headers contain the data on grid dimensions, a description of file parameters, and information that is used by the database software to access records in the file. There is some overlap among the information stored in these three places; e.g., the number of rows and columns in the grid is available in all three places.

3.2.1 REGPRM.EXT and REGION.EXT

Domain-dependent constants, particularly those required for dimensioning purposes in the ROM processor programs and subroutines, are to be found in the INCLUDE files REGPRM.EXT for the various domains. REGION.EXT is a similar INCLUDE file containing just the GRID1 name and grid dimensions, and is used by the ROM Core Model. The constants in REGPRM.EXT are the names of the associated grids, the time zone, the maximum number of upper-air and surface meteorology sites, the maximum number of levels in upper-air soundings (used for dimensioning records in PF/MF files), and other physical constants that depend on the spatial distribution of the meteorology inputs. There are also master grid name and PF/MF file name tables.

Prior to compiling programs for a given domain, you must assign the logical name 'REGPRMEXT1:' to the directory containing the REGPRM.EXT for that region. Other standard logical names for directories containing source code are as follows: 'UTILIOEXT1:' is the source code directory for the utility and I/O software (various processors and analysis programs use PARMS.EXT, FHEAD2.EXT, PI.EXT, and GRDDEF.EXT from this directory). 'P*nn*GEXT1:' is the source code directory for processor P*nn*G (where *nn* represents the processor number).

We show REGPRM.EXT for the ROMNET region below; we define and elaborate on several of the parameters following the listing.

```
C *** REGPRM.EXT ***********************************************************
C
C  Regional parameter file.  This file is used by all ROM processors
C  to define all region specific parameters.
C
      CHARACTER*8  GRDNM1, GRDNM2, GRDNM3, GRDNM4, GRDNM5, GRDNM6
      PARAMETER   ( GRDNM1 =  'ROMNET1 ',
     &              GRDNM2 =  'ROMNET2 ',
     &              GRDNM3 =  'ROMNET3 ',
     &              GRDNM4 =  'ROMNET4 ',
     &              GRDNM5 =  'ROMNET5 ',
     &              GRDNM6 =  'ROMNET6 ' )
      INTEGER*4    NRWND, NSURF, TZONE, NVERTS, NSPECI, NSPECO,
     &             NCOLS1, NCOLS2, NCOLS3, NCOLS4, NCOLS5, NCOLS6,
     &             NROWS1, NROWS2, NROWS3, NROWS4, NROWS5, NROWS6,
     &             MXLAKE, NL50, PLEVS, PTOP, INFRAC
      REAL*4       WL50RW, WL50SF, SPFAC, TMPCRIT
      PARAMETER   ( NRWND  =  30,
     &              NSURF  =  250,
     &              TZONE  =  5,
     &              NVERTS =  100,
     &              NCOLS1 =  64,
     &              NCOLS2 =  234,
     &              NCOLS3 =  36,
     &              NCOLS4 =  11,
     &              NCOLS5 =  70,
     &              NCOLS6 =  7,
     &              NROWS1 =  52,
     &              NROWS2 =  132,
     &              NROWS3 =  30,
     &              NROWS4 =  8,
     &              NROWS5 =  58,
     &              NROWS6 =  6,
     &              MXLAKE =  20,
     &              WL50RW =  800,
     &              WL50SF =  250,
     &              NL50   =  100,
     &              NSPECI =  20,
     &              NSPECO =  13,
     &              PLEVS  =  23,
     &              PTOP   =  500,
     &              SPFAC  =  1.0,
     &              TMPCRIT = 3.0,
     &              INFRAC =  50 )
      CHARACTER*12 MF001, MF002, MF003, MF004, MF005, MF006, MF007,
     &             MF008, MF009, MF010, MF011, MF012, MF013, MF014,
     &             MF015, MF016, MF017, MF018, MF019, MF020, MF021,
     &             MF022, MF023, MF024, MF025, MF026, MF027, MF028,
     &             MF029, MF030, MF031, MF032, MF033, MF034, MF035,
     &             MF036, MF037, MF038, MF039, MF040, MF041, MF042,
     &             MF043, MF044, MF045, MF046, MF047, MF048, MF049,
     &             MF050, MF051, MF052, MF053, MF054, MF055, MF056,
     &             MF057, MF058, MF059, MF060, MF061, MF062, MF063,
     &             MF064, MF065, MF066, MF067, MF068, MF069, MF070,
     &             MF071, MF072, MF073, MF074, MF075, MF076, MF077,
     &             MF078, MF079, MF080, MF081, MF082, MF083, MF084,
```

```
&            MF085, MF086, MF087, MF088, MF089, MF090, MF091,
&            MF092, MF093, MF094, MF095, MF096, MF097, MF098,
&            MF099, MF100
 PARAMETER  ( MF001 = 'MF101', MF002 = 'MF102', MF003 = 'MF103',
&             MF004 = 'MF104', MF005 = 'MF105', MF006 = 'MF106',
&             MF007 = 'MF107', MF008 = 'MF108', MF009 = 'MF109',
&             MF010 = 'MF110', MF011 = 'MF111', MF012 = 'MF112',
&             MF013 = 'MF113', MF014 = 'MF114', MF015 = 'MF115',
&             MF016 = 'MF116', MF017 = 'MF117', MF018 = 'MF118',
&             MF019 = 'MF119', MF020 = 'MF120', MF021 = 'MF121',
&             MF022 = 'MF122', MF023 = 'MF123', MF024 = 'MF124',
&             MF025 = 'MF125', MF026 = 'MF126', MF027 = 'MF127',
&             MF028 = 'MF128', MF029 = 'MF129', MF030 = 'MF130',
&             MF031 = 'MF131', MF032 = 'MF132', MF033 = 'MF133',
&             MF034 = 'MF134', MF035 = 'MF135', MF036 = 'MF136',
&             MF037 = 'MF137', MF038 = 'MF138', MF039 = 'MF139',
&             MF040 = 'MF140', MF041 = 'MF141', MF042 = 'MF142',
&             MF043 = 'MF143', MF044 = 'MF144', MF045 = 'MF145',
&             MF046 = 'MF146', MF047 = 'MF147', MF048 = 'MF148',
&             MF049 = 'MF149', MF050 = 'MF150', MF051 = 'MF151' )
 PARAMETER  ( MF052 = 'MF152', MF053 = 'MF153', MF054 = 'MF154',
&             MF055 = 'MF155', MF056 = 'MF156', MF057 = 'MF157',
&             MF058 = 'MF158', MF059 = 'MF159', MF060 = 'MF160',
&             MF061 = 'MF161', MF062 = 'MF162', MF063 = 'MF163',
&             MF064 = 'MF164', MF065 = 'MF165', MF066 = 'MF166',
&             MF067 = 'MF167', MF068 = 'MF168', MF069 = 'MF169',
&             MF070 = 'MF170', MF071 = 'MF171', MF072 = 'MF172',
&             MF073 = 'MF173', MF074 = 'MF174', MF075 = 'MF175',
&             MF076 = 'MF176', MF077 = 'MF177', MF078 = 'MF178',
&             MF079 = 'MF179', MF080 = 'MF180', MF081 = 'MF181',
&             MF082 = 'MF182', MF083 = 'MF183', MF084 = 'MF184',
&             MF085 = 'MF185', MF086 = 'MF186', MF087 = 'MF187',
&             MF088 = 'MF188', MF089 = 'MF189', MF090 = 'MF190',
&             MF091 = 'MF191', MF092 = 'MF192', MF093 = 'MF193',
&             MF094 = 'MF194', MF095 = 'MF195', MF096 = 'MF196',
&             MF097 = 'MF197', MF098 = 'MF198', MF099 = 'MF199',
&             MF100 = 'MF200' )
 CHARACTER*12 PF001, PF002, PF003, PF004, PF005, PF006, PF007,
&             PF008, PF009, PF010, PF011, PF012, PF013, PF014,
&             PF015, PF016, PF017, PF018, PF019, PF020, PF021,
&             PF022, PF023, PF024, PF025, PF026, PF027, PF028,
&             PF029, PF030, PF031, PF032, PF033, PF034, PF035,
&             PF036, PF037, PF038, PF039, PF040, PF041, PF042,
&             PF043, PF044, PF045, PF046, PF047, PF048, PF049,
&             PF050, PF051, PF052, PF053, PF054, PF055, PF056,
&             PF057, PF058, PF059, PF060, PF061, PF062, PF063,
&             PF064, PF065, PF066, PF067, PF068, PF069, PF070,
&             PF071, PF072, PF073, PF074, PF075, PF076, PF077,
&             PF078, PF079, PF080, PF081, PF082, PF083, PF084,
&             PF085, PF086, PF087, PF088, PF089, PF090, PF091,
&             PF092, PF093, PF094, PF095, PF096, PF097, PF098,
&             PF099, PF100
```

```
      PARAMETER    ( PF001 = 'PF101', PF002 = 'PF102', PF003 = 'PF103',
     &                PF004 = 'PF104', PF005 = 'PF105', PF006 = 'PF106',
     &                PF007 = 'PF107', PF008 = 'PF108', PF009 = 'PF109',
     &                PF010 = 'PF110', PF011 = 'PF111', PF012 = 'PF112',
     &                PF013 = 'PF113', PF014 = 'PF114', PF015 = 'PF115',
     &                PF016 = 'PF116', PF017 = 'PF117', PF018 = 'PF118',
     &                PF019 = 'PF119', PF020 = 'PF120', PF021 = 'PF121',
     &                PF022 = 'PF122', PF023 = 'PF123', PF024 = 'PF124',
     &                PF025 = 'PF125', PF026 = 'PF126', PF027 = 'PF127',
     &                PF028 = 'PF128', PF029 = 'PF129', PF030 = 'PF130',
     &                PF031 = 'PF131', PF032 = 'PF132', PF033 = 'PF133',
     &                PF034 = 'PF134', PF035 = 'PF135', PF036 = 'PF136',
     &                PF037 = 'PF137', PF038 = 'PF138', PF039 = 'PF139',
     &                PF040 = 'PF140', PF041 = 'PF141', PF042 = 'PF142',
     &                PF043 = 'PF143', PF044 = 'PF144', PF045 = 'PF145',
     &                PF046 = 'PF146', PF047 = 'PF147', PF048 = 'PF148',
     &                PF049 = 'PF149', PF050 = 'PF150', PF051 = 'PF151' )
      PARAMETER    ( PF052 = 'PF152', PF053 = 'PF153', PF054 = 'PF154',
     &                PF055 = 'PF155', PF056 = 'PF156', PF057 = 'PF157',
     &                PF058 = 'PF158', PF059 = 'PF159', PF060 = 'PF160',
     &                PF061 = 'PF161', PF062 = 'PF162', PF063 = 'PF163',
     &                PF064 = 'PF164', PF065 = 'PF165', PF066 = 'PF166',
     &                PF067 = 'PF167', PF068 = 'PF168', PF069 = 'PF169',
     &                PF070 = 'PF170', PF071 = 'PF171', PF072 = 'PF172',
     &                PF073 = 'PF173', PF074 = 'PF174', PF075 = 'PF175',
     &                PF076 = 'PF176', PF077 = 'PF177', PF078 = 'PF178',
     &                PF079 = 'PF179', PF080 = 'PF180', PF081 = 'PF181',
     &                PF082 = 'PF182', PF083 = 'PF183', PF084 = 'PF184',
     &                PF085 = 'PF185', PF086 = 'PF186', PF087 = 'PF187',
     &                PF088 = 'PF188', PF089 = 'PF189', PF090 = 'PF190',
     &                PF091 = 'PF191', PF092 = 'PF192', PF093 = 'PF193',
     &                PF094 = 'PF194', PF095 = 'PF195', PF096 = 'PF196',
     &                PF097 = 'PF197', PF098 = 'PF198', PF099 = 'PF199',
     &                PF100 = 'PF200' )
C    Variable descriptions
C
C       NRWND     The maximum # of rawinsonde (upper air) sites
C       NSURF     The maximum # of surface stations used
C       NVERTS    The maximum # of vertical soundings
C       TZONE     The # of hours shifted from Greenwich Mean Time
C       GRDNMn    The name of grid n
C       NCOLSn    The # of columns in grid n
C       NROWSn    The # of rows in grid n
C       MXLAKE    The maximum number of lakes that a region may contain
C       WL50RW    50% filtered wavelength for upper air data
C       WL50SF    50% filtered wavelength for surface data
C       NL50      Number of heights for 50-meter interpolated data
C       SPFAC     Weight for wind speed in PF012 formula
C       TMPCRIT   Cutoff temperature for local inversion
C       INFRAC    Percentage of local inversion to trigger regional inversion
C       MFOpq     The logical MF file name for series mpq, m = 1, 2, . . .
C       PFOpq     The logical PF file name for series mpq, m = 1, 2, . . .
C
C ***************************************************************************
```

- GRDNMn are grid names. For ROMNET, grids 1, 2, and 5 are defined in P06G; grids 3, 4, and 6 are defined below:

  GRDNM3 is the acceptance window for PF106, generated by P03G. It consists of GRDNM1 ±6 degrees of longitude and ±4 degrees of latitude. The grid cells are approximately $^1/_2$ degree of longitude by $^3/_4$ degree of latitude.

  GRDNM4 is used for gridding divergences in P11G. It is the same as GRDNM1, but with grid cells of approximately 1 degree of longitude by $^2/_3$ degree of latitude.

  GRDNM6 is used for gridding cumulus clouds in P05G. It consists of GRDNM1 ±3 degrees of longitude and $\pm 2^1/_2$ degree of latitude. The grid cells are approximately $2^1/_2$ degrees of longitude by 2 degrees of latitude.

- NSPECI and NSPECO are obsolete and not used by ROM2.1.
- WL50RW and WL50SF are filtering radii used in the Barnes' gridding analyses.
- PTOP is the lowest atmospheric pressure (in millibars) for the vertical profiles used in the model; effectively, PTOP is the *highest* model top.
- SPFAC, TMPCRIT, and INFRAC are used in the surface-based inversion/no inversion decision made in P03G. The algorithm is as follows:

  [(current gridded surface temp. - sunset gridded surface temp.) + (SPFAC × gridded surface wind speed)].

  The result of the calculation is compared against TMPCRIT:

  if ≤TMPCRIT,the grid cell has an inversion;

  if >TMPCRIT, the grid cell does not have an inversion.

  The region is considered to have an inversion if ≥INFRAC percent of grid cells individually have inversions. (We decided on the 50% criterion after we examined satellite images.)
- PF1*pq* and MF1*pq* are the ROMNET-specific processor and model files, respectively.

3.2.2 CELL0

There are six different region-dependent grids used by the ROM processors. Let us term these GRID1, ..., GRID6 (where "GRID" must be replaced by a region name: NEROS, SEROS, ROMNET, SEROXA, or MIDOXA). There are also a number of extended-region grids that are used by analysis and graphics software. Complete grid definitions, contained in CELL0.FOR, can be found in Appendix B.

Note that if you choose to construct your own grid, you should select grid dimensions that yield approximately square grid cells. For example, our choice of $^1/_4$ degree longitude by $^1/_6$ degree latitude cells for the fundamental grid results in cells that are approximately 20 km on each side at 40° N.

- GRID1 is the fundamental grid on which calculations are based. It has cells with longitudinal and latitudinal dimensions of $^1/_4$ degree by $^1/_6$ degree respectively (approximately 20 km on each side). GRID1 is used by the majority of the ROM processors, as well as by the Core Model.

- GRID2 is a high-resolution grid that has cells dimensions of $^1/_{12}$ degree by $^1/_{12}$ degree. It covers the same geographical area as GRID1, but extends seven GRID1-sized cells outward from GRID1. GRID2 is used primarily for terrain computations in processors P06G and P07G.
- GRID3 surrounds GRID1 with a border stretching 6 degrees longitudinally and 4 degrees latitudinally. The grid cell dimensions are variable according to the region. This grid is used by processors P04G and P09G.
- GRID4 is a low-resolution (coarse) grid covering the same geographical area as GRID1. The grid cell dimensions are variable according to the region. The sole use of GRID4 is the internal divergence computations of processor P11G.
- GRID5 has the same cell-dimensions as GRID1, and surrounds it with a three-cell border in all directions. Processor P06G computes terrain and its derivatives on this grid; processor P07G uses this grid for its nocturnal inversion computations.
- GRID6 is a very-low-resolution grid that surrounds GRID1 with a border stretching 3 degrees longitudinally and 2.5 degrees latitudinally. The grid cell dimensions are variable according to the region. This grid is used internally by processor P05G for cumulus cloud cover computations.
- 'CONTUS ' is a grid covering the entire continental United States; it has the cell sizes of GRID1, and is used only for nationwide mapping routines.
- 'BIG ', 'RAWIND ', 'ROMNEX ', 'ROMNETEX', 'BUOYNET', and 'ULTRANET' are extended-region grids that are used primarily with map-drawing routines in preliminary meteorology quality assurance and analysis.

Calls to CELL0 with a CHARACTER*8 grid-name input argument return the latitudinal and longitudinal boundaries, the number of rows and columns, and the cell dimensions (in degrees) for that grid. ROM processor, graphics, and analysis programs and subroutines needing this geographical information obtain it at runtime from CELL0.

3.2.3 PF/MF Files

Currently, there are four PF/MF file series in use: Series 0 files (e.g., PF001) have a GRID1 with dimensions 60 by 42, and are for either the NEROS or the SEROS domains. Series 100 files (e.g., PF101) have a GRID1 with dimensions 64 by 52, and are for the ROMNET domain. Series 200 files (e.g., PF201) have a GRID1 with dimensions 90 by 58, and are for the SEROXA domain. Series 300 files (e.g., PF301) have a GRID1 with dimensions 76 by 71, and are for the MIDOXA domain.

This version of the ROM requires that both the database directory and the file name encode the file series (0, 100, 200, ...) correctly. This creates domain dependencies in the file-name tables of the processor and analysis software source codes. In version 2.1 of the ROM, there is a master name-table in each region's REGPRM.EXT that defines the character string parameters PF001 through

PF099 and MF001 through MF099, and contains the correct file names for that region. Version 2.2 of the ROM will permit hard-coding the file names in FORTRAN source code independently of series number -- e.g.,

```
CHARACTER*12  WINPROF / 'PF001        ' /
```

The database software will then adjust names at run-time to match the series indicator of the database directory used. There is a utility program, FSERIES, which can be used with the ROM2.1 database software to achieve a similar result, albeit not so transparently.

3.2.4 Mapping Routines

There are several map-drawing routines used by the analysis and graphics software: they are GEOPOL, AMAP, LMAP, PMAP, and QMAP. Each routine has two versions that are indicated by the suffixes "_ASCII" and "_BIN" (binary). The routines take a grid name as their first argument. The routines concatenate 'MAP' or 'BIN' onto the end of the grid name (so that, for example, 'ROMNET1 ' becomes 'ROMNET1BIN') in order to generate a VMS logical name for the appropriate map data file. You must ASSIGN map data files to appropriate logical names before executing these analysis programs (e.g., ROMNET1MAP). The "_ASCII" versions expect list-directed data files; input consists of lines, each of which has values of latitude, longitude, and pen status that constitute map drawing instructions for HP-plotters. The "_BIN" versions expect unformatted FORTRAN binary files with record length 3; each record has the same information as provided for the "_ASCII" versions. Note that the binary versions have substantial performance and storage-space advantages, but are not portable across machine architectures.

SECTION 4

REFERENCES AND BIBLIOGRAPHY

The ROM has been described at length in Lamb 1983 and 1984a, and Lamb and Laniak, 1985.

Baldocchi, D., *et al.*, 1984. Solar radiation within an oak-hickory forest: an evaluation of the extinction coefficients for several radiation components during fully-leafed and leafless periods. *Agricultural and Forest Meteorology* **32**: 307-322.

Barnes, S.L. (1973). Mesoscale objective map analysis using weighted time series observations. *NOAA Tech. Memo. ERL NSSL-62.* Norman, OK. 60 pp.

Carter, W.P.L., M.C. Dodd, W.D. Long, and R. Atkinson, 1985. *Outdoor Chamber Study to Test Multi-Day Effects - Volume I: Results and Discussion.* EPA/600/3-84/115. U.S. Environmental Protection Agency, Research Triangle Park, NC.

Demerjian, K.L., K.L. Schere, and J.T. Peterson, 1980. Theoretical estimates of actinic (spherically integrated) flux and photolytic rate constants of atmospheric species in the lower troposphere. In *Advances in Environmental Science and Technology* **10**. J.N. Pitts *et al.*, eds., John Wiley and Sons, New York, NY, pp. 369-459.

Demmy, J.L., W.M. Tax, T.E. Warn, 1989. *Area Source Documentation for the 1985 National Acid Precipitation Assessment Program Inventory.* EPA/600/S8-88/106, Air and Energy Engineering Research Laboratory, Research Triangle Park, NC.

Dodge, M.C., 1989. A comparison of three photochemical oxidant mechanisms. *J. Geophys. Res.* 94:5121-5136.

Domaratz, M.A., C.A. Hallam, W.E. Schmidt, H.W. Calkins, 1983. *Digital Line Graphs From 1:2,000,000-Scale Maps.* Geological Survey Circular 895-D. U.S. Geological Survey, Reston, VA. 38 pp.

EPA, 1990. *Regional Ozone Modeling for Northeast Transport (ROMNET): Final Project Report.* Possiel, N., L. Milich, and B. Goodrich, eds. EPA/450/4-90/002a and EPA/450/4-90/002b. Office of Air Quality Planning and Standards, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Gates, D. and L. Papian, 1971. *Atlas of Energy Budgets on Plant Leaves.* Academic Press, New York, New York, pp. 1-16.

Gay, D., 1987. *A national inventory of biogenic hydrocarbon emissions based upon a simple forest canopy model.* M.S. Thesis, Washington State University, Pullman, WA. 73 pp.

Gery, M.W., G.Z. Whitten, and J.P. Killus, 1988. *Development and testing of the CBM-IV for urban and regional modeling.* EPA/600/3-88/012, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Gery, M.W., G.Z. Whitten, J.P. Killus, and M.C. Dodge, 1989. A photochemical kinetics mechanism for urban and regional scale computer models. *J. Geophys. Res.* **94**:D10, 12295-12956.

Homolya, J.B., D.A. Toothman, and J.D. Mobley, 1985. *Development of the 1985 NAPAP (National Acid Precipitation Assessment Program) Emissions Inventory.* EPA/600/D-85/151, Air and Energy Engineering Research Laboratory, Research Triangle Park, NC.

Killus, J.P, and G.Z. Whitten, 1984. *Technical Discussion Relating to the Use of the Carbon Bond Mechanism in OZIPM/EKMA.* EPA/450/4-84/009, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Lamb, B., A. Guenther, D. Gay, and H. Westberg, 1987. A national inventory of biogenic hydrocarbon emissions. *Atmospheric Environment*, **21**(8):1695-1705.

Lamb, B., D. Gay, H. Westberg, and E. Allwine, 1990. *Development of a National Inventory for Natural Hydrocarbon Emissions.* Invited paper presented at the NAPAP 1990 International Conference on "Acidic Deposition: State of Science and Technology," February 11 16 , 1990, Hilton Head Island, SC. National Acidic Precipitation Program, Washington, DC.

Lamb, R.G., 1983. *A regional scale (1000 km) model of photochemical air pollution. Part I - Theoretical formulation.* EPA/600/3-83/035, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Lamb, R.G., 1984a. *A regional scale (1000 km) model of photochemical air pollution. Part II - Input processor network design.* EPA/600/3-84/085, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Lamb, R.G., 1984b. Air pollution models as descriptors of cause-effect relationships. *Atmos. Environ.* 18(3):591-606.

Lamb, R.G., 1985. *Application of the first generation Regional Oxidant Model to an assessment of the effects of proposed 1987 emissions reductions on episodic ozone levels in the northeastern United States.* Internal Report, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Lamb, R.G., 1986. *Numerical simulations of photochemical air pollution in the northeastern United States: ROM 1.0 Applications.* EPA/600/3-86/038, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Lamb, R.G., 1987. *Design and applications of the Regional Oxidant Model (ROM).* In: Conference Proceedings of the North American Oxidant Symposium, February 25-27 1987, Quebec, Canada. Environment Canada, 153 - 189.

Lamb, R.G., 1988. *Simulated effects of hydrocarbon emissions controls on seasonal ozone levels in the northeastern United States: A preliminary study.* EPA/600/3-88/017, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Lamb, R.G., and G.F. Laniak, 1985. *A regional scale (1000 km) model of photochemical air pollution. Part III - Tests of the numerical algorithms.* EPA/600/3-85/037, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Lamb, R.G., and S.K. Hati, 1987. The representation of atmospheric motion in models of regional-scale air pollution. *J. Clim. Appl. Meteorol.* 26(7):837-846.

Matthews, E., 1984. *Vegetation, Land Use and Seasonal Albedo Data Sets: Documentation of Archived Data Tape.* NASA Technical Memorandum 86107, National Aeronautics and Space Administration, Goddard Space Flight Center, Institute for Space Studies. New York, NY.

Modica, L.G., D.R. Dulleba, R.A. Walters, and J.E. Langstaff, 1989. *Flexible Regional Emissions Data System (FREDS) Documentation for the 1985 NAPAP Emissions Inventory.* EPA/600/S9-89/047, Air and Energy Engineering Research Laboratory, Research Triangle Park, NC.

Novak, J.H., and J.A. Reagan, 1986. *A comparison of natural and man-made hydrocarbon emission inventories necessary for regional acid deposition and oxidant modeling,* paper 86-30. In: Proceedings of the 79th Annual APCA Meeting. APCA, Pittsburgh, PA.

Olson, R.J., 1980. *Geoecology: a County-Level Environmental Data Base for the Conterminous United States.* Publication No. 1537, Oak Ridge National Laboratory, Environmental Sciences Division, Oak Ridge, TN.

Page, S.H., 1980. *National Land Use and Land Cover Inventory.* Lockheed Engineering and Management Services Co., Inc., Remote Sensing Laboratory. Las Vegas, NV.

Pierce, T.E., and K.L. Schere, 1988. *Regional ozone modeling: An investigation of hydrocarbon emissions from treatment, storage, and disposal facilities on ambient levels of ozone.* Internal Report, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Pierce, T.E., B.K. Lamb, and A.R. Van Meter, 1990a. *Development of a Biogenic Emissions Inventory System for Regional Scale Air Pollution Models.* Paper number 90-94.3 presented at the 83rd Air and Waste Management Association Annual Meeting, Pittsburgh, PA, June 24 - 29, 1990.

Pierce, T.E., L.L. Schere, D.C. Doll, and W.E. Heilman, 1990b. *Evaluation of the Regional Oxidant Model (Version 2.1) using ambient and diagnostic simulations.* Draft Report, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Possiel, N.C., J.A. Tikvart, J.H. Novak, K.L. Schere, and E.L. Meyer, 1989. *Evaluation of ozone control strategies in the northeastern region of the United States.* In: Proceedings of the Third U.S.-Dutch International Symposium on Atmospheric Ozone Research and Its Policy Implications, May 9-13 1988, Nijmegen, The Netherlands. Elsevier, The Netherlands, pp. 623-632.

Saeger, M., J. Langstaff, R. Walters, L. Modica, D. Zimmerman, D. Fratt, D. Dulleva, R. Ryan, J. Demmy, Y. Tax, D. Sprague, D. Mudgett, and A Werner, 1989. *The 1985 NAPAP Emissions Inventory Version 2: Development of the Annual Data and Modeler's Tapes.* EPA/600/7-89/012a. Air and Engineering Research Laboratory, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Schere, K.L., 1986. *EPA Regional Oxidant Model: ROM 1.0 evaluation for 3-4 August 1979.* EPA/600/3-86/032, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Schere, K.L., 1986. *Evaluating O_3 predictions from a test application of the EPA Regional Oxidant Model.* In: Proceedings of the Fifth Joint Conference on Applications of Air Pollution Meteorology, November 18-21 1986, Chapel Hill, NC. AMS, Boston, MA, pp. 61-64.

Schere, K.L., and A.J. Fabrick, 1985. *EPA Regional Oxidant Model: description and evaluation plan.* EPA/600/3-85/067, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Schere, K.L., and J.H. Novak, 1986. *Regional oxidant modeling of the northeast U.S.* In: C. DeWispelaere *et al.*, eds. Air Pollution Modeling and Its Application, V - Vol. 10. Plenum Press, NY, pp. 45-59.

Schere, K.L., and N.P. Possiel, 1984. *U.S. EPA Regional Oxidant Model - background and overview,* paper no. 84-47.2. In: Proceedings of the 77th Annual APCA Meeting, June 25-29 1984, San Francisco, CA. APCA, Pittsburgh, PA.

Schere, K.L., and R.A. Wayland, 1989. *Development and evaluation of the Regional Oxidant Model for the northeastern United States.* In: Proceedings of the Third U.S.-Dutch International Symposium on Atmospheric Ozone Research and Its Policy Implications, May 9-13 1988, Nijmegen, The Netherlands. Elsevier, The Netherlands, pp. 613-622.

Schere, K.L., and R.A. Wayland, 1989. *EPA Regional Oxidant Model (ROM 2.0): Evaluation on 1980 NEROS databases.* EPA/600/3-89/057, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Sellars, F. M. *et al.*, 1985. *National Acid Precipitation Assessment Program Emission Inventory Allocation Factors.* EPA/600/7-85/035. U.S. Environmental Protection Agency, Research Triangle Park, NC.

Tingey, D.T., 1981. *The effect of environmental factors on the emission of biogenic hydrocarbons from live oak and slash pine.* In: J.J. Bufalini and R.R. Arnts (eds.), Atmospheric Biogenic Hydrocarbons, Vol. 1, Emissions. Ann Arbor Science, Ann Arbor, Michigan.

Walters, R.A., L.G. Modica, and D.B. Fratt, 1990. *The 1985 NAPAP Emissions Inventory: Overview of Allocation Factors.* EPA/600/S7-89/010A, Air and Energy Engineering Research Laboratory, Research Triangle Park, NC.

Whitten, G.Z., and M.W. Gery, 1986. *Development of CBM-X mechanisms for urban and regional AQSMs.* EPA/600/3-86/012, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Whitten, G.Z., J.P. Killus, and R.G. Johnson, 1985. *Modeling of Auto Exhaust Smog Chamber Data for EKMA Development.* EPA/600/3-85/025. U.S. Environmental Protection Agency, Research Triangle Park, NC.

Winer, A.M., *et al.*, 1983. *Investigation of the Role of Natural Hydrocarbons in Photochemical Smog Formation in California.* Final Report A0/056-32, California Air Resources Board, Statewide Air Pollution Research Center, University of California, Riverside, CA.

Young, J.O., M. Aissa, T. Boehm, C. Coats, J. Eichinger, D. Grimes, S. Hallyburton, W. Heilman, D. Olerud, S. Roselle, A. Van Meter, R. Wayland, and T. Pierce, 1989. Development of the Regional Oxidant Model, Version 2.1. Project Report, T.E. Pierce, Project Officer, Atmospheric Sciences Modeling Division, U.S. Environmental Protection Agency, Research Triangle Park, NC.

Zimmerman, P.R., 1979. *Determination of Emission Rates of Hydrocarbons from Indigenous Species of Vegetation in the Tampa/St. Petersburg, Florida Area.* EPA/904/9-77/028, U.S. Environmental Protection Agency.

INDEX

This page is intentionally left blank.

APPENDIX A

GRID DOMAIN DEFINITIONS

The NEROS, SEROS, and ROMNET domains are shown in Part 1, Figure 2.4.

A.1 NEROS GRID DOMAINS

The following are base values used for calculating the corners (NE and SW) of expanded or shrunken grid domains:

```
SWLON  =  84.0 degrees
SWLAT  =  38.0 degrees
NELON  =  69.0 degrees
NELAT  =  45.0 degrees

  GRDNN  =  GRDNA              !  NEROS1
  NCOLL  =  60                 !  Number of columns
  NROWW  =  42                 !  Number of rows
  SWLON  =  84.0               !  Southwest corner longitude
  SWLAT  =  38.0               !  Southwest corner latitude
  NELON  =  69.0               !  Northeast corner longitude
  NELAT  =  45.0               !  Northeast corner latitude
  DLONN  =  - 1/4              !  (NELON - SWLON) / NCOLL
  DLATT  =  1/6                !  (NELAT - SWLAT) / NROWW

  GRDNN  =  GRDNA              !  NEROS2
  NCOLL  =  222                !  Number of columns
  NROWW  =  112                !  Number of rows
  SWLON  =  85.75              !  84.0 + 7.0 * (3.0 * (5.0 / 60.0))
  SWLAT  =  36.833333          !  38.0 - 7.0 * (2.0 * (5.0 / 60.0))
  NELON  =  67.25              !  69.0 - 7.0 * (3.0 * (5.0 / 60.0))
  NELAT  =  46.1667            !  45.0 + 7.0 * (2.0 * (5.0 / 60.0))
  DLONN  =  - 1/12             !  (NELON - SWLON) / NCOLL
  DLATT  =  1/12               !  (NELAT - SWLAT) / NROWW

  GRDNN  =  GRDNA              !  NEROS3
  NCOLL  =  36                 !  Number of columns
  NROWW  =  30                 !  Number of rows
  SWLON  =  90.0               !  84.0 + 6.0
  SWLAT  =  34.0               !  34.0 - 4.0
  NELON  =  63.0               !  69.0 - 6.0
  NELAT  =  49.0               !  45.0 + 4.0
  DLONN  =  - 3/4              !  (NELON - SWLON) / NCOLL
  DLATT  =  1/2                !  (NELAT - SWLAT) / NROWW

  GRDNN  =  GRDNA              !  NEROS4
  NCOLL  =  11                 !  Number of columns
  NROWW  =  8                  !  Number of rows
  SWLON  =  84.0               !  Southwest corner longitude
  SWLAT  =  38.0               !  Southwest corner latitude
  NELON  =  69.0               !  Northeast corner longitude
  NELAT  =  45.0               !  Northeast corner latitude
  DLONN  =  - 1.3636364        !  (NELON - SWLON) / NCOLL
  DLATT  =  0.875              !  (NELAT - SWLAT) / NROWW
```

```
GRDNN  =  GRDNA              !  NEROS5
NCOLL  =  66                 !  Number of columns
NROWW  =  48                 !  Number of rows
SWLON  =  84.75              !  84.0 + 3.0 * (3.0 * (5.0 / 60.0))
SWLAT  =  37.50              !  38.0 - 3.0 * (2.0 * (5.0 / 60.0))
NELON  =  68.25              !  69.0 - 3.0 * (3.0 * (5.0 / 60.0))
NELAT  =  45.5               !  45.0 + 3.0 * (2.0 * (5.0 / 60.0))
DLONN  =  - 1/4              !  (NELON - SWLON) / NCOLL
DLATT  =  1/6                !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA              !  NEROS6
NCOLL  =  7                  !  Number of columns
NROWW  =  6                  !  Number of rows
DLONN  =  -3.0               !  (NELON - SWLON) / NCOLL
DLATT  =  2.0                !  (NELAT - SWLAT) / NROWW
SWLON  =  87.0               !  84.0 + 3.0
SWLAT  =  35.5               !  38.0 - 2.5
NELON  =  66.0               !  69.0 - 3.0
NELAT  =  47.5               !  45.0 + 2.5
```

A.2 SEROS GRID DOMAINS

The following are base values used for calculating the corners (NE and SW) of expanded or shrunken grid domains:

```
SWLON  =  97.0 degrees
SWLAT  =  28.0 degrees
NELON  =  82.0 degrees
NELAT  =  35.0 degrees
```

```
GRDNN  =  GRDNA              !  SEROS1
NCOLL  =  60                 !  Number of columns
NROWW  =  42                 !  Number of rows
SWLON  =  97.0               !  Southwest corner longitude
SWLAT  =  28.0               !  Southwest corner latitude
NELON  =  82.0               !  Northeast corner longitude
NELAT  =  35.0               !  Northeast corner latitude
DLONN  =  - 1/4              !  (NELON - SWLON) / NCOLL
DLATT  =  1/6                !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA              !  SEROS2
NCOLL  =  222                !  Number of columns
NROWW  =  112                !  Number of rows
SWLON  =  100.50             !  97.0 + 7.0 * (3.0 * (5.0 / 60.0))
SWLAT  =  26.833333          !  28.0 - 7.0 * (2.0 * (5.0 / 60.0))
NELON  =  80.25              !  82.0 - 7.0 * (3.0 * (5.0 / 60.0))
NELAT  =  36.166667          !  35.0 + 7.0 * (2.0 * (5.0 / 60.0))
DLONN  =  - 1/12             !  (NELON - SWLON) / NCOLL
DLATT  =  1/12               !  (NELAT - SWLAT) / NROWW
```

```
GRDNN  =  GRDNA              !  SEROS3
NCOLL  =  36                 !  Number of columns
NROWW  =  30                 !  Number of rows
SWLON  =  103.0              !  97.0 + 6.0
SWLAT  =  24.0               !  28.0 - 4.0
NELON  =  76.0               !  82.0 - 6.0
NELAT  =  39.0               !  35.0 + 4.0
DLONN  =  - 3/4              !  (NELON - SWLON) / NCOLL
DLATT  =  1/2                !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA              !  SEROS4
NCOLL  =  11                 !  Number of columns
NROWW  =  8                  !  Number of rows
SWLON  =  97.0               !  Southwest corner longitude
SWLAT  =  28.0               !  Southwest corner latitude
NELON  =  82.0               !  Northeast corner longitude
NELAT  =  35.0               !  Northeast corner latitude
DLONN  =  - 1.3636364        !  (NELON - SWLON) / NCOLL
DLATT  =  0.875              !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA              !  SEROS5
NCOLL  =  66                 !  Number of columns
NROWW  =  48                 !  Number of rows
SWLON  =  97.75              !  97.0 + 3.0 * (3.0 * (5.0 / 60.0))
SWLAT  =  27.50              !  28.0 - 3.0 * (2.0 * (5.0 / 60.0))
NELON  =  81.25              !  82.0 - 3.0 * (3.0 * (5.0 / 60.0))
NELAT  =  35.50              !  35.0 + 3.0 * (2.0 * (5.0 / 60.0))
DLONN  =  - 1/4              !  (NELON - SWLON) / NCOLL
DLATT  =  1/6                !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA              !  SEROS6
NCOLL  =  7                  !  Number of columns
NROWW  =  6                  !  Number of rows
DLONN  =  -3.0               !  (NELON - SWLON) / NCOLL
DLATT  =  2.0                !  (NELAT - SWLAT) / NROWW
SWLON  =  100.0              !  97.0 + 3.0
SWLAT  =  25.5               !  28.0 - 2.5
NELON  =  79.0               !  82.0 - 3.0
NELAT  =  37.5               !  35.0 + 2.5
```

A.3 ROMNET GRID DOMAINS

The following are base values used for calculating the corners (NE and SW) of expanded or shrunken grid domains:

```
SWLON  =  85.0        degrees
SWLAT  =  36.3333333  degrees
NELON  =  69.0        degrees
NELAT  =  45.0        degrees
```

```
GRDNN  =  GRDNA               !  ROMNET1
NCOLL  =  64                  !  Number of columns
NROWW  =  52                  !  Number of rows
SWLON  =  85.0                !  Southwest corner longitude
SWLAT  =  36.3333333          !  Southwest corner latitude
NELON  =  69.0                !  Northeast corner longitude
NELAT  =  45.0                !  Northeast corner latitude
DLONN  =  - 1/4               !  (NELON - SWLON) / NCOLL
DLATT  =  1/6                 !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA               !  ROMNET2
NCOLL  =  234                 !  Number of columns
NROWW  =  132                 !  Number of rows
SWLON  =  86.75               !  85.0 + 7.0 * (3.0 * (5.0 / 60.0))
SWLAT  =  35.1666667          !  36.3 - 7.0 * (2.0 * (5.0 / 60.0))
NELON  =  67.25               !  69.0 - 7.0 * (3.0 * (5.0 / 60.0))
NELAT  =  46.1666667          !  45.0 + 7.0 * (2.0 * (5.0 / 60.0))
DLONN  =  - 1/12              !  (NELON - SWLON) / NCOLL
DLATT  =  1/12                !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA               !  ROMNET3
NCOLL  =  36                  !  Number of columns
NROWW  =  30                  !  Number of rows
SWLON  =  91.0                !  85.0 + 6.0
SWLAT  =  32.33333333         !  36.333333 - 4.0
NELON  =  63.0                !  69.0 - 6.0
NELAT  =  49.0                !  45.0 + 4.0
DLONN  =  - 0.7777778         !  (NELON - SWLON) / NCOLL
DLATT  =  0.5555556           !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA               !  ROMNET4
NCOLL  =  16                  !  Number of columns
NROWW  =  13                  !  Number of rows
SWLON  =  85.0                !  Southwest corner longitude
SWLAT  =  36.3333333          !  Southwest corner latitude
NELON  =  69.0                !  Northeast corner longitude
NELAT  =  45.0                !  Northeast corner latitude
DLONN  =  - 1.0               !  (NELON - SWLON) / NCOLL
DLATT  =  0.6666667           !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA               !  ROMNET5
NCOLL  =  70                  !  Number of columns
NROWW  =  58                  !  Number of rows
SWLON  =  85.75               !  85.0 + 3.0 * (3.0 * (5.0 / 60.0))
SWLAT  =  35.8333333          !  36.3 - 3.0 * (2.0 * (5.0 / 60.0))
NELON  =  68.25               !  69.0 - 3.0 * (3.0 * (5.0 / 60.0))
NELAT  =  45.5                !  45.0 + 3.0 * (2.0 * (5.0 / 60.0))
DLONN  =  - 1/4               !  (NELON - SWLON) / NCOLL
DLATT  =  1/6                 !  (NELAT - SWLAT) / NROWW
```

```
GRDNN  =  GRDNA          ! ROMNET6
NCOLL  =   8             ! Number of columns
NROWW  =   7             ! Number of rows
DLONN  =  -2.75          ! (NELON - SWLON) / NCOLL
DLATT  =  1.95238        ! (NELAT - SWLAT) / NROWW
SWLON  =  88.0           ! 85.0   + 3.0
SWLAT  =  33.83333       ! 36.333 - 2.5
NELON  =  66.0           ! 69.0   - 3.0
NELAT  =  47.5           ! 45.0   + 2.5
```

A.4 SEROXA GRID DOMAINS

The following are base values used for calculating the corners (NE and SW) of expanded or shrunken grid domains:

```
SWLON  =  98.75     degrees
SWLAT  =  27.666667 degrees
NELON  =  76.25     degrees
NELAT  =  37.333333 degrees
```

```
GRDNN  =  GRDNA          ! SEROXA1
NCOLL  =  90             ! Number of columns
NROWW  =  58             ! Number of rows
SWLON  =  98.75          ! Southwest corner longitude
SWLAT  =  27.666667      ! Southwest corner latitude
NELON  =  76.25          ! Northeast corner longitude
NELAT  =  37.333333      ! Northeast corner latitude
DLONN  =  - 1/4          ! (NELON - SWLON) / NCOLL
DLATT  =  1/6            ! (NELAT - SWLAT) / NROWW
```

```
GRDNN  =  GRDNA          ! SEROXA2
NCOLL  =  284            ! Number of columns
NROWW  =  130            ! Number of rows
SWLON  =  98.75          ! 98.75 + 7.0 * (3.0 * (5.0 / 60.0))
SWLAT  =  25.50          ! 27.666667 - 7.0 * (2.0 * (5.0 / 60.0))
NELON  =  74.50          ! 76.25 - 7.0 * (3.0 * (5.0 / 60.0))
NELAT  =  38.50          ! 37.333333 + 7.0 * (2.0 * (5.0 / 60.0))
DLONN  =  - 1/12         ! (NELON - SWLON) / NCOLL
DLATT  =  1/12           ! (NELAT - SWLAT) / NROWW
```

```
GRDNN  =  GRDNA          ! SEROXA3
NCOLL  =  36             ! Number of columns
NROWW  =  30             ! Number of rows
SWLON  =  104.75         ! 98.75     + 6.0
SWLAT  =  23.666667      ! 27.666667 - 4.0
NELON  =  71/4           ! 76.25     - 6.0
NELAT  =  41.333333      ! 37.333333 + 4.0
DLONN  =  - 0.9583333    ! (NELON - SWLON) / NCOLL
DLATT  =  0.5888889      ! (NELAT - SWLAT) / NROWW
```

```
GRDNN  =  GRDNA              !  SEROXA4
NCOLL  =  23                 !  Number of columns
NROWW  =  15                 !  Number of rows
SWLON  =  98.75              !  Southwest corner longitude
SWLAT  =  27.666667          !  Southwest corner latitude
NELON  =  76.25              !  Northeast corner longitude
NELAT  =  37.333333          !  Northeast corner latitude
DLONN  =  - 0.9782609        !  (NELON - SWLON) / NCOLL
DLATT  =  0.7111111          !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA              !  SEROXA5
NCOLL  =  96                 !  Number of columns
NROWW  =  64                 !  Number of rows
SWLON  =  99.50              !  98.75 + 3.0 * (3.0 * (5.0 / 60.0))
SWLAT  =  27.166667          !  27.666667 - 3.0 * (2.0 * (5.0 / 60.0))
NELON  =  75.50              !  76.25 - 3.0 * (3.0 * (5.0 / 60.0))
NELAT  =  37.833333          !  37.333333 + 3.0 * (2.0 * (5.0 / 60.0))
DLONN  =  - 1/4              !  (NELON - SWLON) / NCOLL
DLATT  =  1/6                !  (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA              !  SEROXA6
NCOLL  =  10                 !  Number of columns
NROWW  =  8                  !  Number of rows
DLONN  =  -2.85              !  (NELON - SWLON) / NCOLL
DLATT  =  1.833333           !  (NELAT - SWLAT) / NROWW
SWLON  =  101.75             !  98.75     + 3.0
SWLAT  =  25.16667           !  27.666667 - 2.5
NELON  =  73.25              !  76.25     - 3.0
NELAT  =  39.83333           !  37.333333 + 2.5
```

A.5 MWROXA GRID DOMAINS

The following are base values used for calculating the corners (NE and SW) of expanded or shrunken grid domains:

```
SWLON  =  97.0      degrees
SWLAT  =  35.0      degrees
NELON  =  78.0      degrees
NELAT  =  46.83333  degrees
```

```
GRDNN  =  GRDNA              !  MWROXA1
NCOLL  =  76                 !  Number of columns
NROWW  =  71                 !  Number of rows
SWLON  =  97.0               !  Southwest corner longitude
SWLAT  =  35.0               !  Southwest corner latitude
NELON  =  78.0               !  Northeast corner longitude
NELAT  =  46.83333           !  Northeast corner latitude
DLONN  =  - 1/4              !  (NELON - SWLON) / NCOLL
DLATT  =  1/6                !  (NELAT - SWLAT) / NROWW
```

```
GRDNN  =  GRDNA          ! MWROXA2
NCOLL  =  242            ! Number of columns
NROWW  =  156            ! Number of rows
SWLON  =  98.75          ! 97.0 + 7.0 * (3.0 * (5.0 / 60.0))
SWLAT  =  33.8333333     ! 35.0 - 7.0 * (2.0 * (5.0 / 60.0))
NELON  =  76.25          ! 78.0 - 7.0 * (3.0 * (5.0 / 60.0))
NELAT  =  48.00          ! 46.83333 + 7.0 * (2.0 * (5.0 / 60.0))
DLONN  =  - 1/12         ! (NELON - SWLON) / NCOLL
DLATT  =  1/12           ! (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA          ! MWROXA3
NCOLL  =  36             ! Number of columns
NROWW  =  30             ! Number of rows
SWLON  =  103.0          ! 97.0 + 6.0
SWLAT  =  31.0           ! 35.0 - 4.0
NELON  =  72.0           ! 78.0 - 6.0
NELAT  =  50.833333      ! 46.83333 + 4.0
DLONN  =  - 0.8611111    ! (NELON - SWLON) / NCOLL
DLATT  =  0.6611111      ! (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA          ! MWROXA4
NCOLL  =  19             ! Number of columns
NROWW  =  18             ! Number of rows
SWLON  =  97.0           ! Southwest corner longitude
SWLAT  =  35.0           ! Southwest corner latitude
NELON  =  78.0           ! Northeast corner longitude
NELAT  =  46.83333       ! Northeast corner latitude
DLONN  =  - 1.0          ! (NELON - SWLON) / NCOLL
DLATT  =  0.6572389      ! (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA          ! MWROXA5
NCOLL  =  82             ! Number of columns
NROWW  =  77             ! Number of rows
SWLON  =  97.75          ! 97.0     + 3.0 * (3.0 * (5.0 / 60.0))
SWLAT  =  34.50          ! 35.0     - 3.0 * (2.0 * (5.0 / 60.0))
NELON  =  77.25          ! 78.0     - 3.0 * (3.0 * (5.0 / 60.0))
NELAT  =  46.33333       ! 46.83333 + 3.0 * (2.0 * (5.0 / 60.0))
DLONN  =  - 1/4          ! (NELON - SWLON) / NCOLL
DLATT  =  1/6            ! (NELAT - SWLAT) / NROWW

GRDNN   =  GRDNA         ! MWROXA6
NCOLL   =  9             ! Number of columns
NROWW   =  9             ! Number of rows
DLONN   =  -2.7777778    ! (NELON - SWLON) / NCOLL
DLATT   =  1.8703703     ! (NELAT - SWLAT) / NROWW
SWLON   = 100.0          ! 97.0     + 3.0
SWLAT   =  32.5          ! 35.0     - 2.5
NELON   =  75.0          ! 78.0     - 3.0
NELAT   =  49.333333     ! 46.83333 + 2.5
```

A.6 CONTUS GRID DOMAIN

This domain is used only for mapping routines that require the entire continental U.S.

```
200   GRDNN  =  GRDNA
      NCOLL  =  240                ! Number of columns
      NROWW  =  156                ! Number of rows
      DLONN  =  - 1/4              ! (NELON - SWLON) / NCOLL
      DLATT  =  1/6                ! (NELAT - SWLAT) / NROWW
      SWLON  =  125.0              ! Southwest corner longitude
      SWLAT  =  25.0               ! Southwest corner latitude
      NELON  =  65.0               ! Northeast corner longitude
      NELAT  =  51.0               ! Northeast corner latitude
```

A.7 QUALITY ASSURANCE GRID DOMAINS

These domains are used for map computations, and have 1-degree-square cells.

```
      GRDNN  =  GRDNA              ! BIG
      NCOLL  =  40                 ! Number of columns
      NROWW  =  22                 ! Number of rows
      SWLON  =  100.00             ! 97.0 + 3.0 * (3.0 * (5.0 / 60.0))
      SWLAT  =   28.00             ! 28.0 - 3.0 * (2.0 * (5.0 / 60.0))
      NELON  =   60.00             ! 82.0 - 3.0 * (3.0 * (5.0 / 60.0))
      NELAT  =   50.00             ! 35.0 + 3.0 * (2.0 * (5.0 / 60.0))
      DLONN  =  - 1.00             ! (NELON - SWLON) / NCOLL
      DLATT  =  1.00               ! (NELAT - SWLAT) / NROWW

      GRDNN  =  GRDNA              ! BUOYNET
      NCOLL  =  35                 ! Number of columns
      NROWW  =  30                 ! Number of rows
      SWLON  =  99.0               ! Southwest corner longitude
      SWLAT  =  20.0               ! Southwest corner latitude
      NELON  =  64.0               ! Northeast corner longitude
      NELAT  =  50.0               ! Northeast corner latitude
      DLONN  =  -1.0               ! (NELON - SWLON) / NCOLL
      DLATT  =  1.0                ! (NELAT - SWLAT) / NROWW

      GRDNN  =  GRDNA              ! RAWIND
      NCOLL  =  120                ! Number of columns
      NROWW  =  60                 ! Number of rows
      DLONN  =  -1/4               ! (NELON - SWLON) / NCOLL
      DLATT  =   1/4               ! (NELAT - SWLAT) / NROWW
      SWLON  =  90.0               ! Southwest corner longitude
      SWLAT  =  32.0               ! Southwest corner latitude
      NELON  =  60.0               ! Northeast corner longitude
      NELAT  =  47.0               ! Northeast corner latitude

      GRDNN  =  GRDNA              ! ROMNETEX
      NCOLL  =  60                 ! Number of columns
      NROWW  =  72                 ! Number of rows
      SWLON  =  80.0               ! Southwest corner longitude
      SWLAT  =  36.0               ! Southwest corner latitude
      NELON  =  65.0               ! Northeast corner longitude
      NELAT  =  48.0               ! Northeast corner latitude
      DLONN  =  - 1/4              ! (NELON - SWLON) / NCOLL
      DLATT  =  1/6                ! (NELAT - SWLAT) / NROWW
```

```
GRDNN  =  GRDNA              ! ROMNEX
NCOLL  =  72                 ! Number of columns
NROWW  =  66                 ! Number of rows
SWLON  =  86.0               ! Southwest corner longitude
SWLAT  =  35.0               ! Southwest corner latitude
NELON  =  68.0               ! Northeast corner longitude
NELAT  =  46.0               ! Northeast corner latitude
DLONN  =  - 1/4              ! (NELON - SWLON) / NCOLL
DLATT  =  1/6                ! (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA              ! RWNDQA
NCOLL  =  45                 ! Number of columns
NROWW  =  30                 ! Number of rows
SWLON  =  105.0              ! Southwest corner longitude
SWLAT  =  25.0               ! Southwest corner latitude
NELON  =  60.0               ! Northeast corner longitude
NELAT  =  55.0               ! Northeast corner latitude
DLONN  =  - 1.0              ! (NELON - SWLON) / NCOLL
DLATT  =  1.0                ! (NELAT - SWLAT) / NROWW

GRDNN  =  GRDNA              ! ULTRANET
NCOLL  =  100                ! Number of columns
NROWW  =  84                 ! Number of rows
SWLON  =  90.0               ! Southwest corner longitude
SWLAT  =  34.0               ! Southwest corner latitude
NELON  =  65.0               ! Northeast corner longitude
NELAT  =  48.0               ! Northeast corner latitude
DLONN  =  - 1/4              ! (NELON - SWLON) / NCOLL
DLATT  =  1/6                ! (NELAT - SWLAT) / NROWW
```

APPENDIX B

PROCESSOR INPUT AND OUTPUT DATA FILES

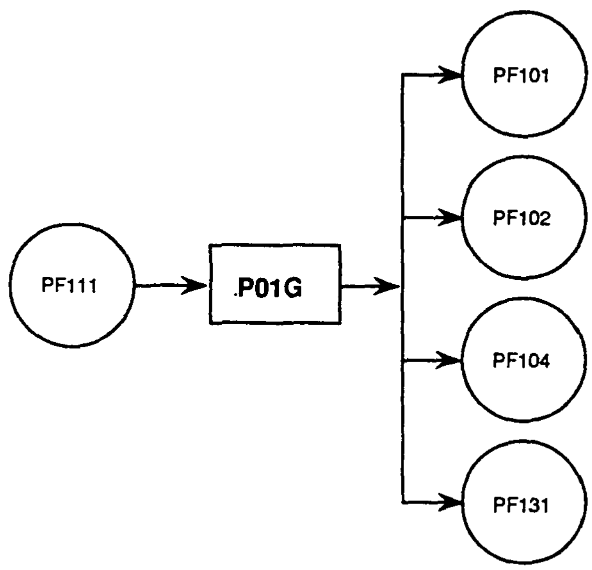


Figure B-1. Processor P01G

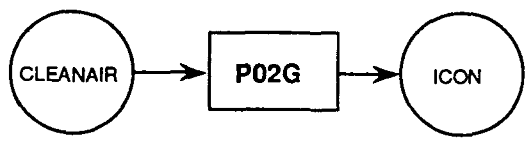


Figure B-2. Processor P02G

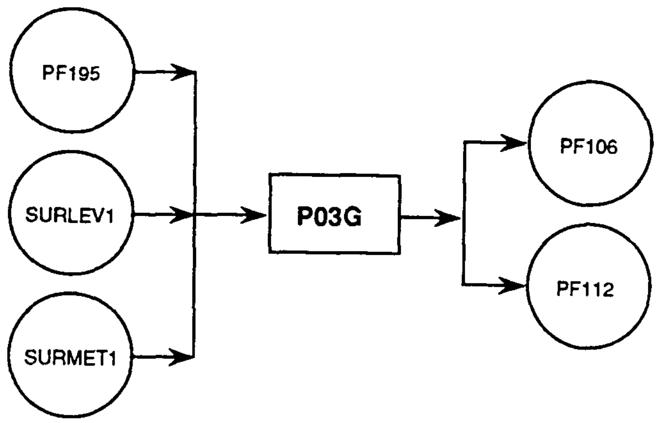


Figure B-3. Processor P03G

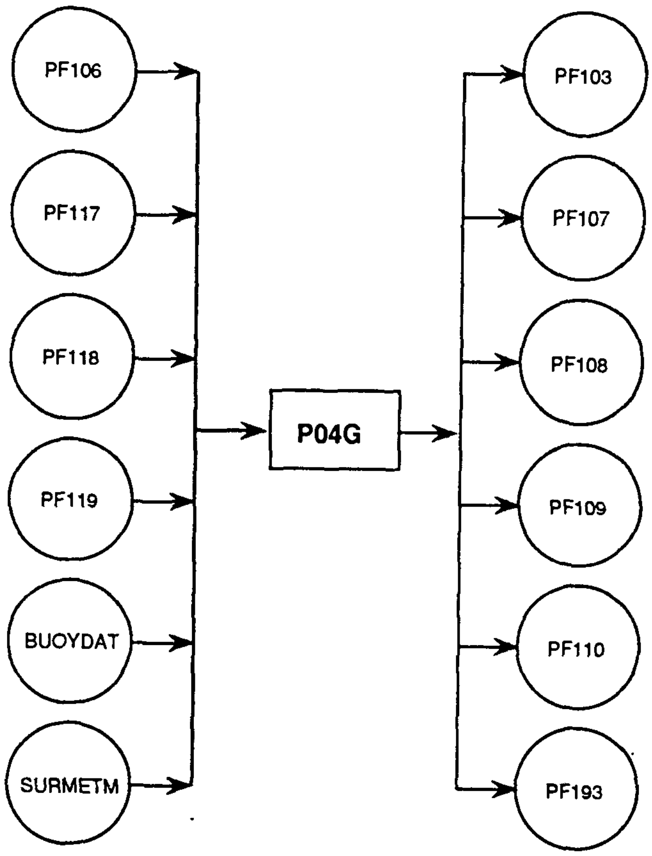


Figure B-4. Processor P04G

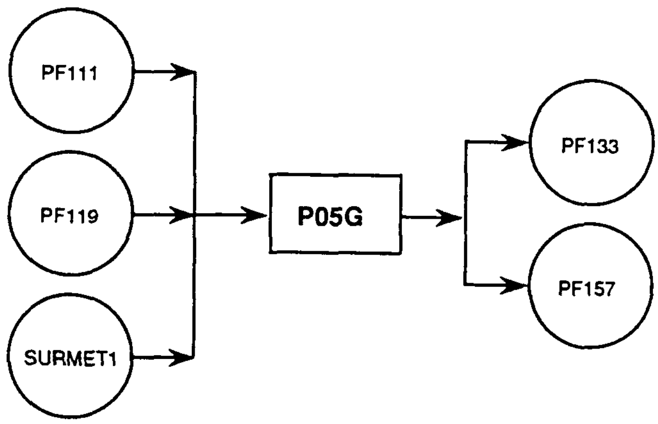


Figure B-5. Processor P05G

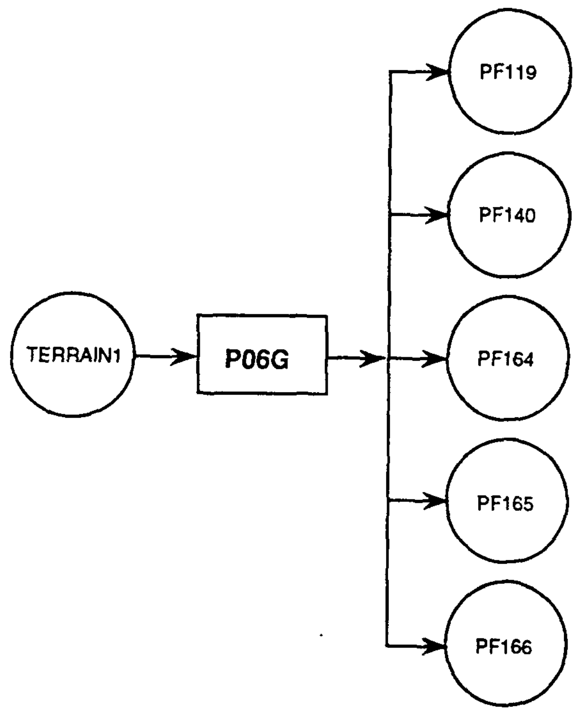


Figure B-6. Processor P06G

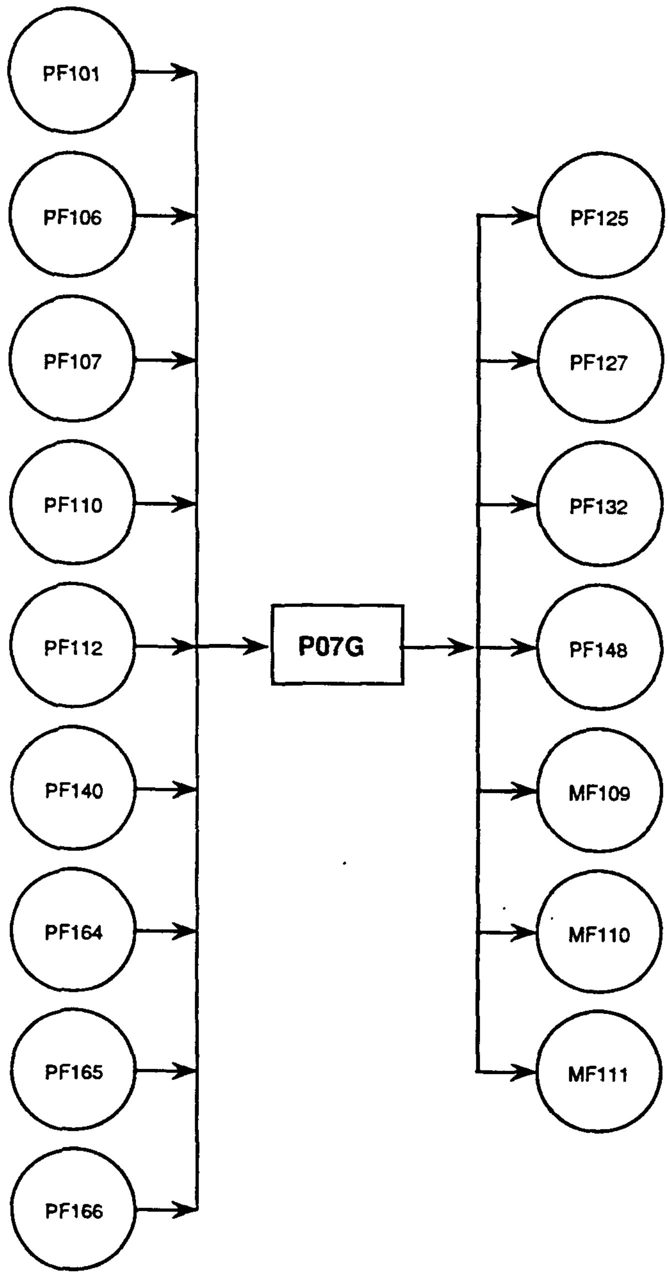


Figure B-7. Processor P07G

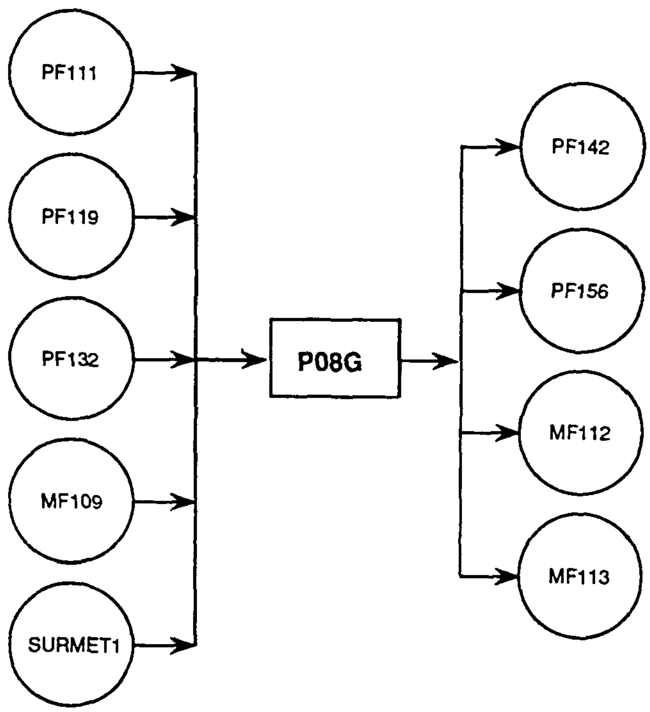


Figure B-8. Processor P08G

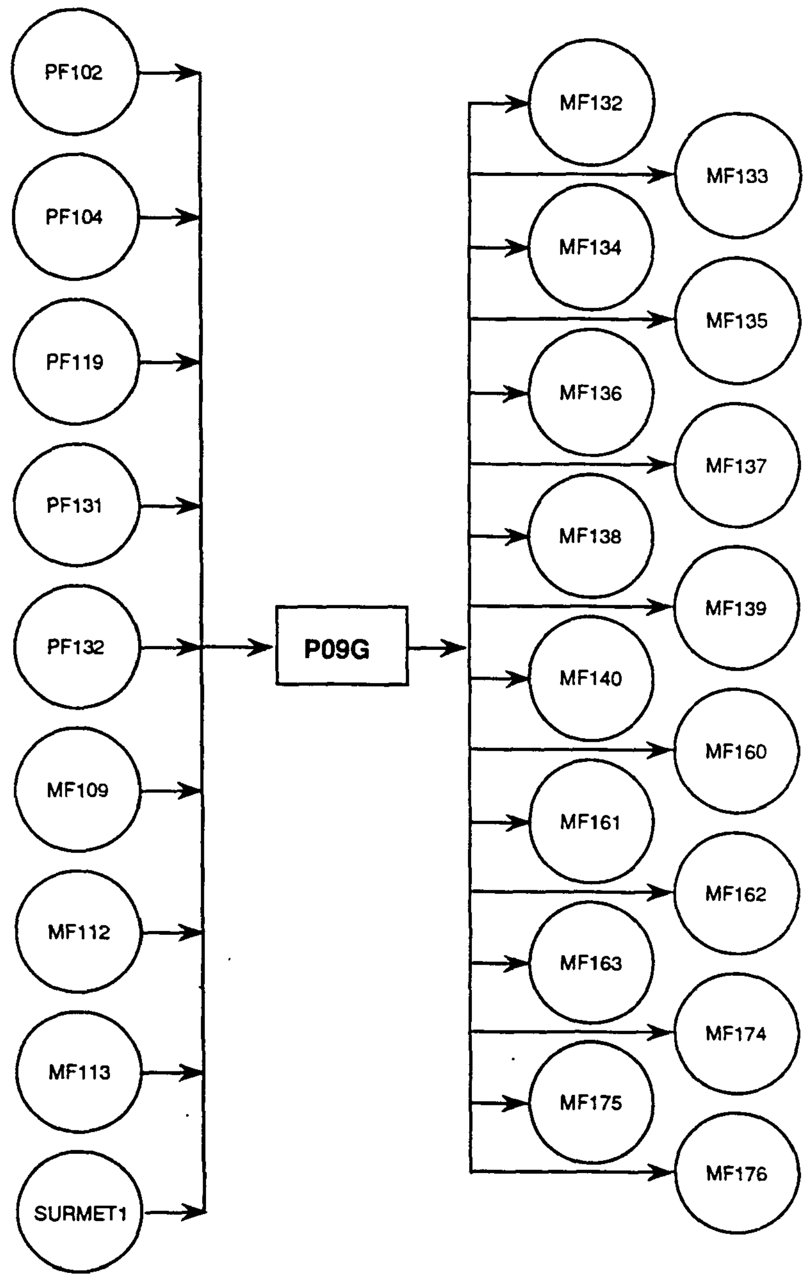


Figure B-9. Processor P09G

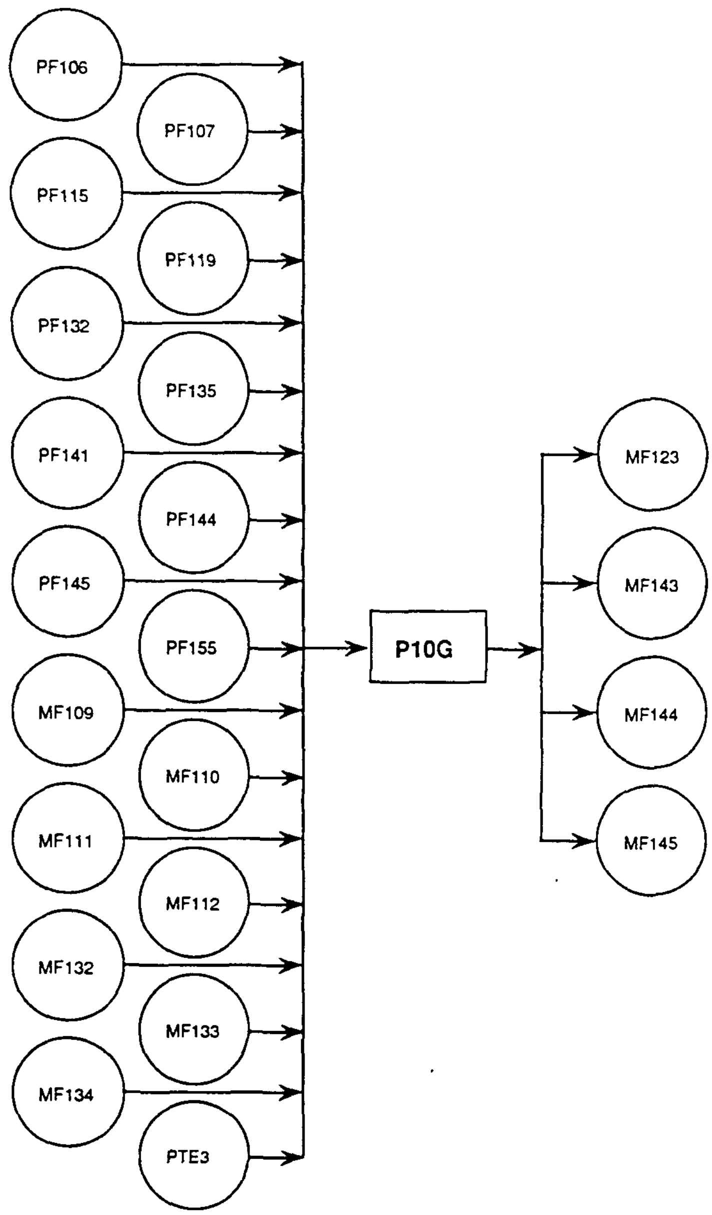


Figure B-10. Processor P10G

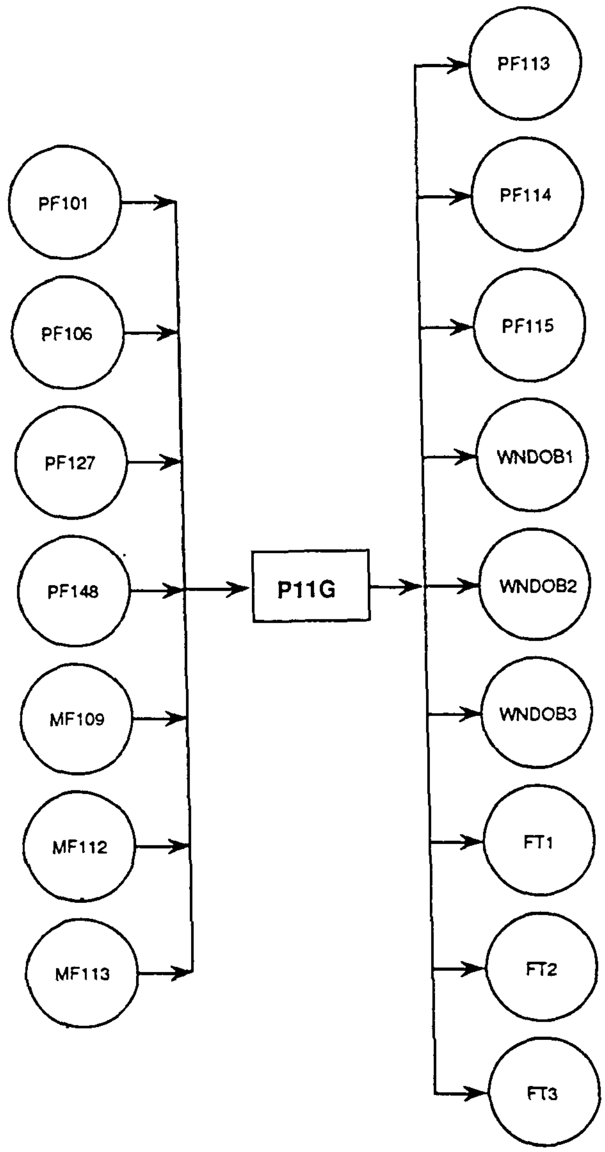


Figure B-11. Processor P11G

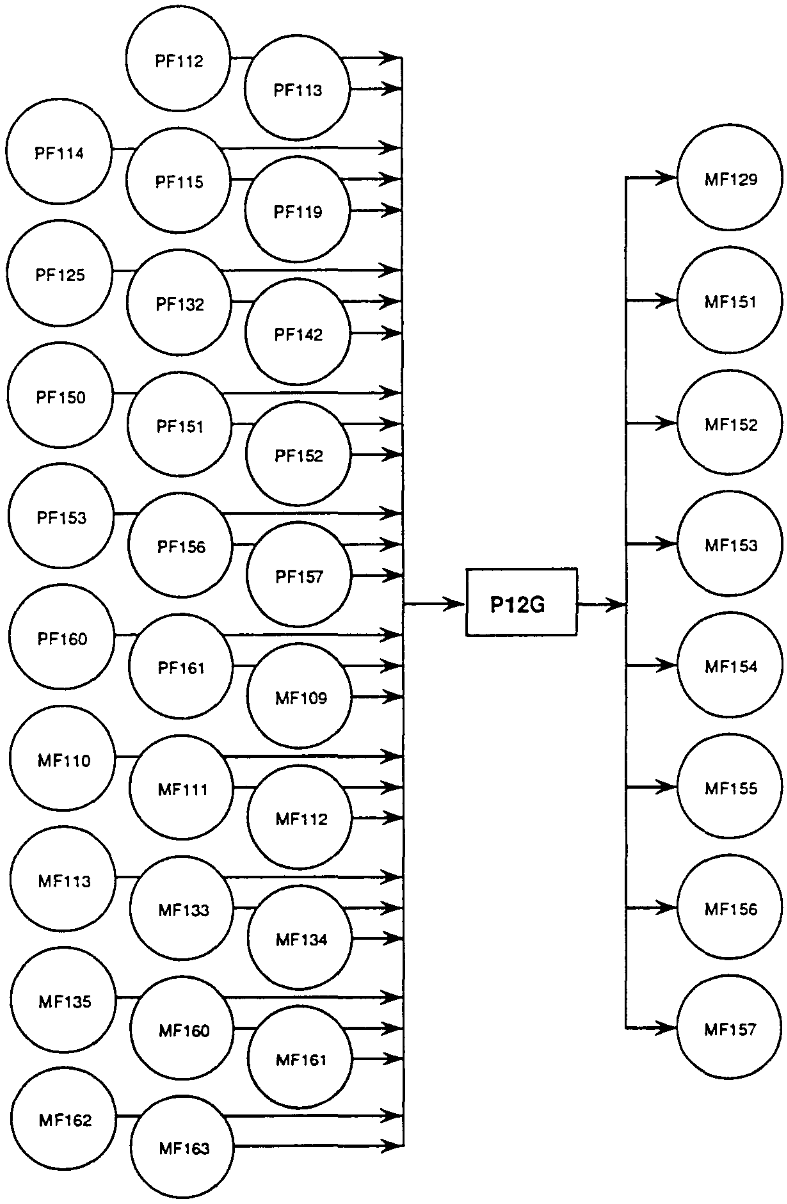


Figure B-12. Processor P12G

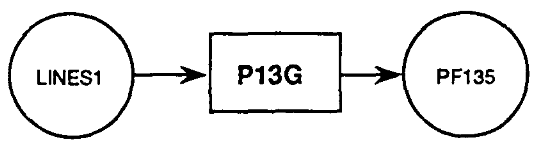


Figure B-13. Processor P13G

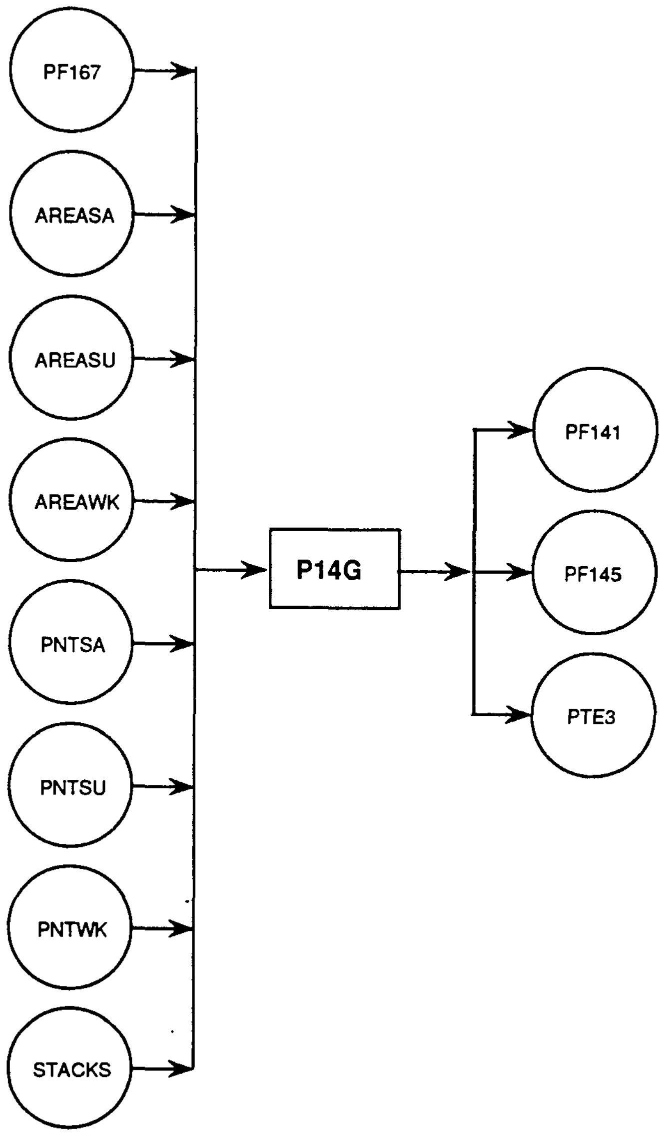


Figure B-14. Processor P14G

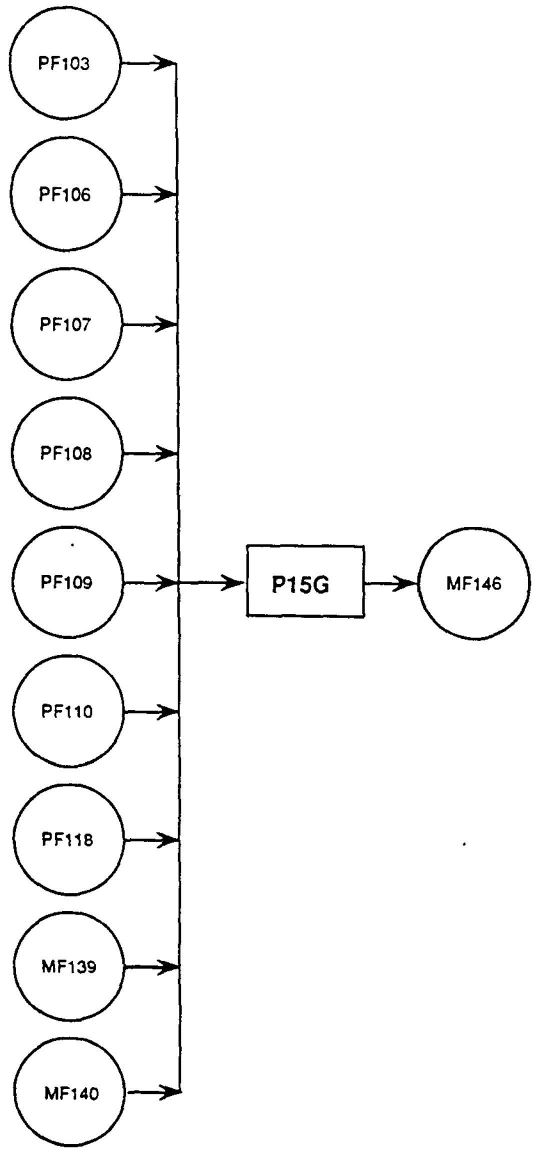


Figure B-15. Processor P15G

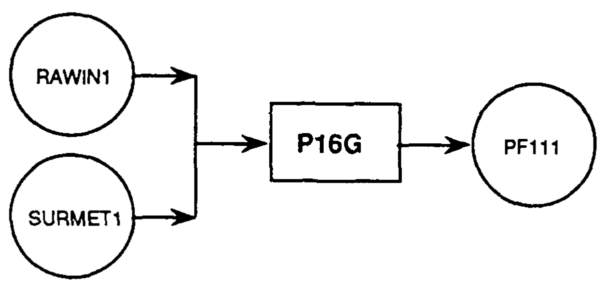


Figure B-16. Processor P16G

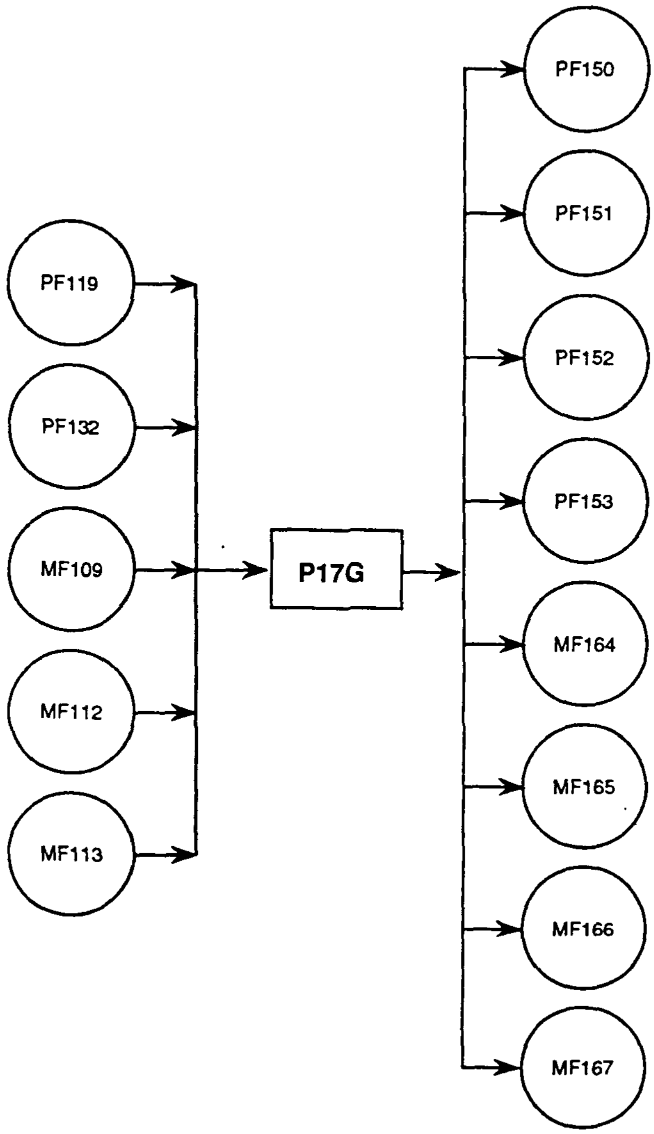


Figure B-17. Processor P17G

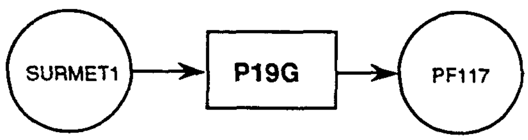


Figure B-18. Processor P19G

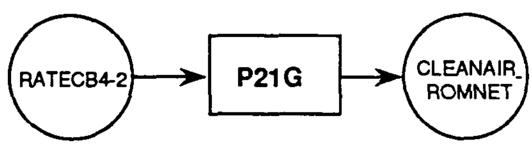


Figure B-19. Processor P21G

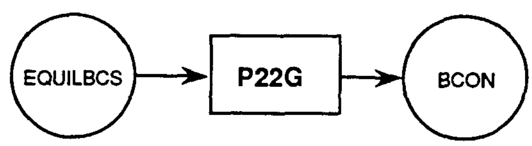


Figure B-20. Processor P22G

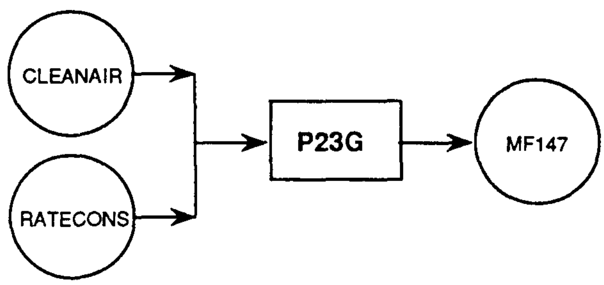


Figure B-21. Processor P23G

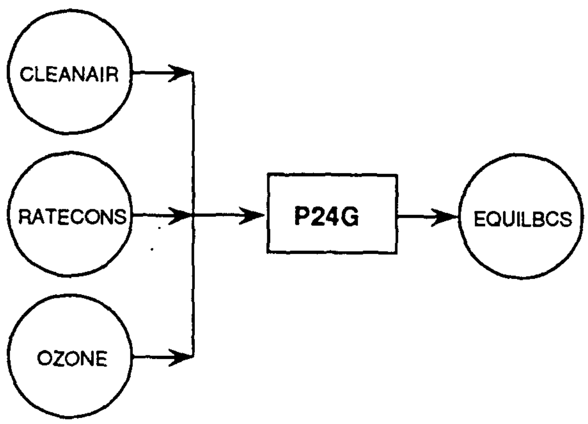


Figure B-22. Processor P24G

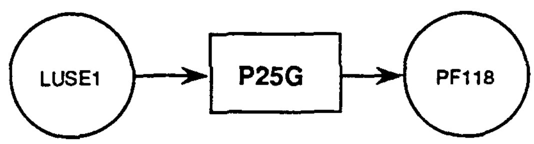


Figure B-23. Processor P25G

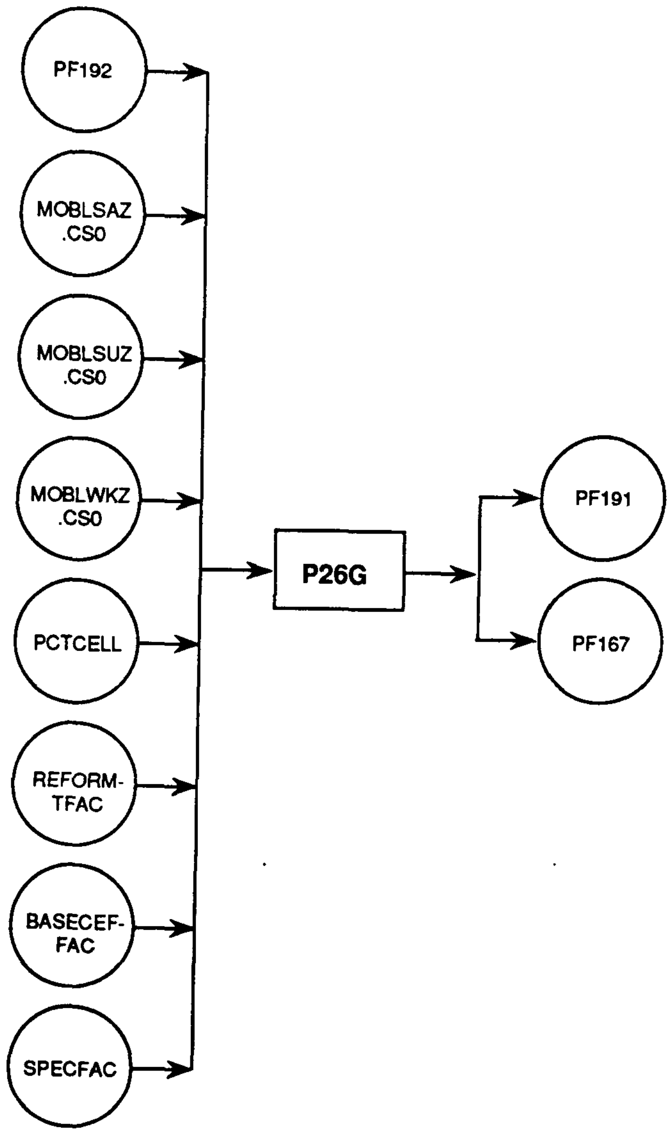


Figure B-24. Processor P26G

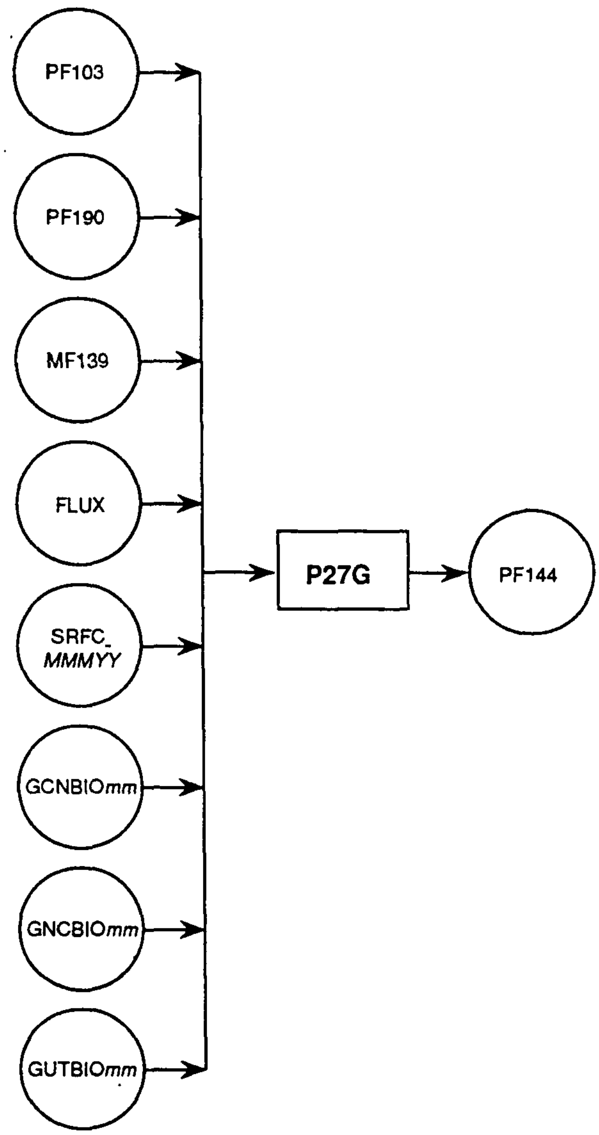


Figure B-25. Processor P27G

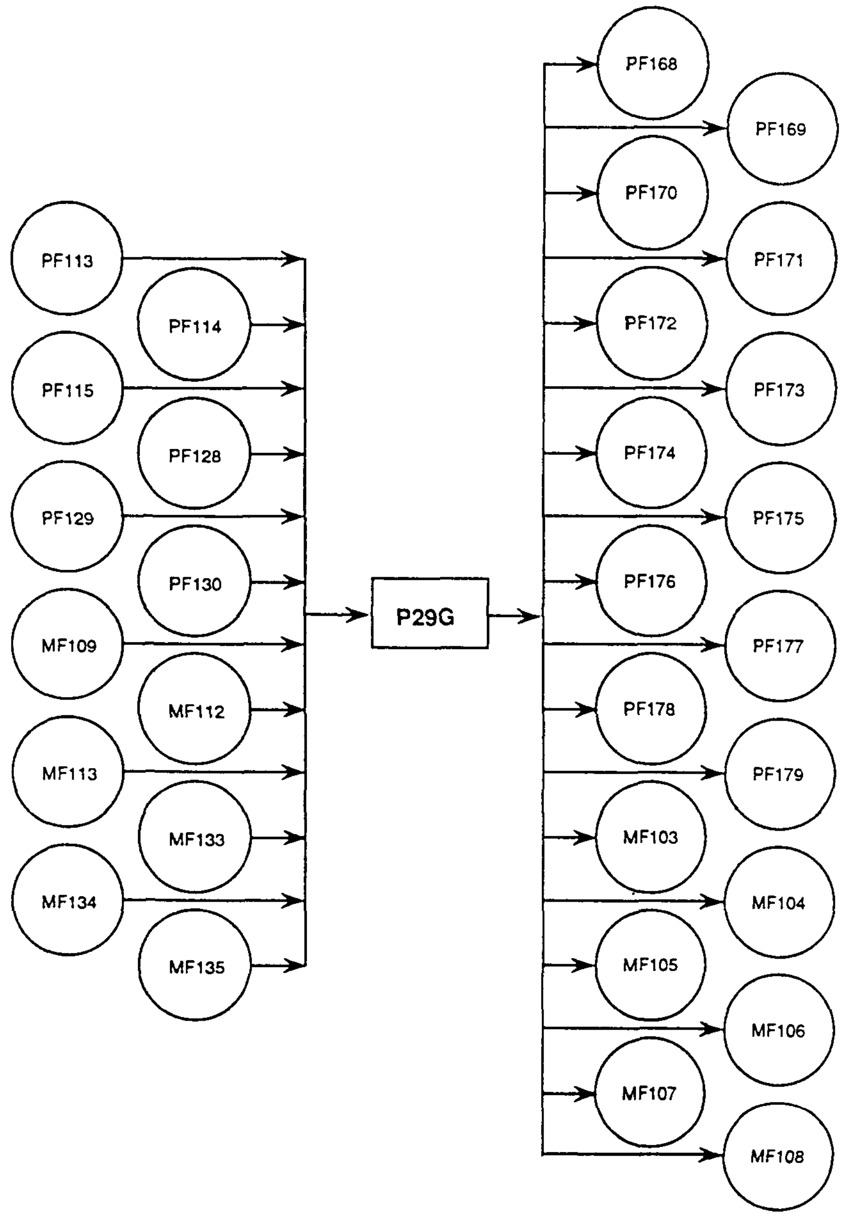


Figure B-26. Processor P29G

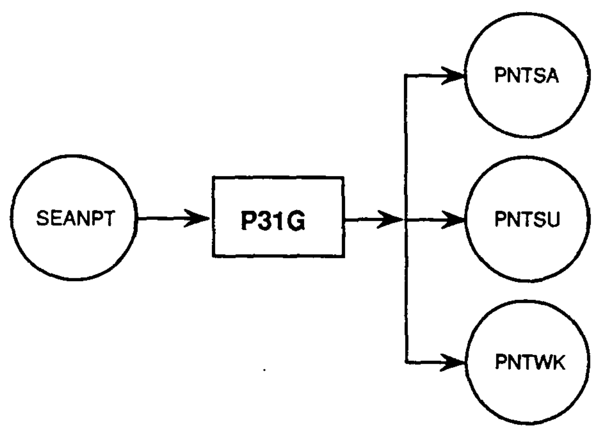


Figure B-27. Processor P31G

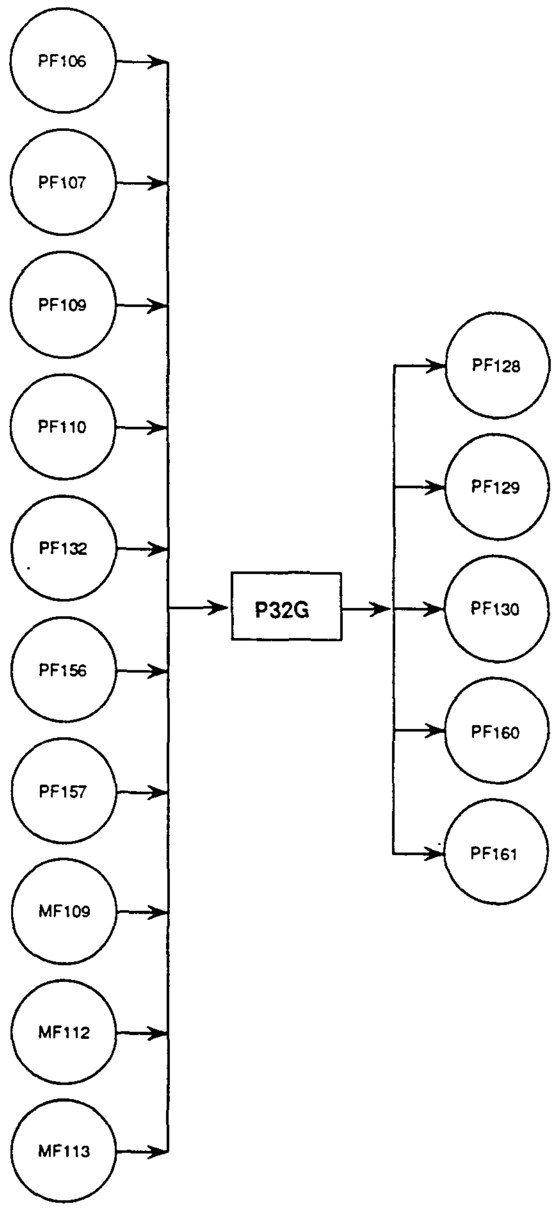


Figure B-28. Processor P32G

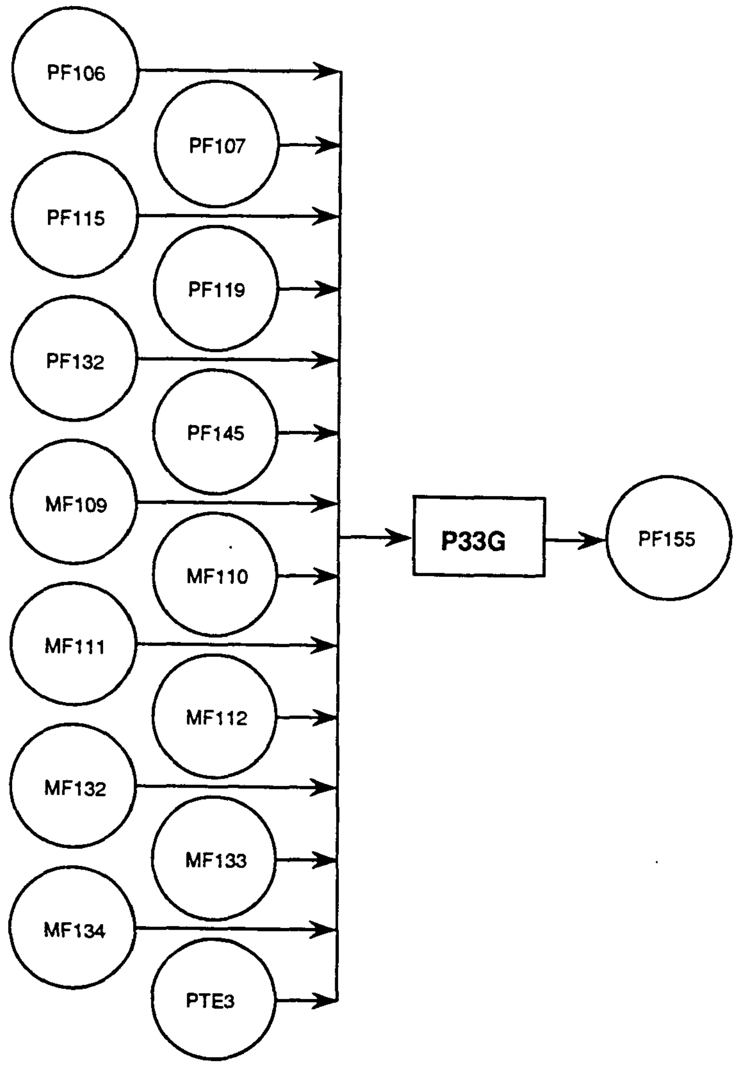


Figure B-29. Processor P33G

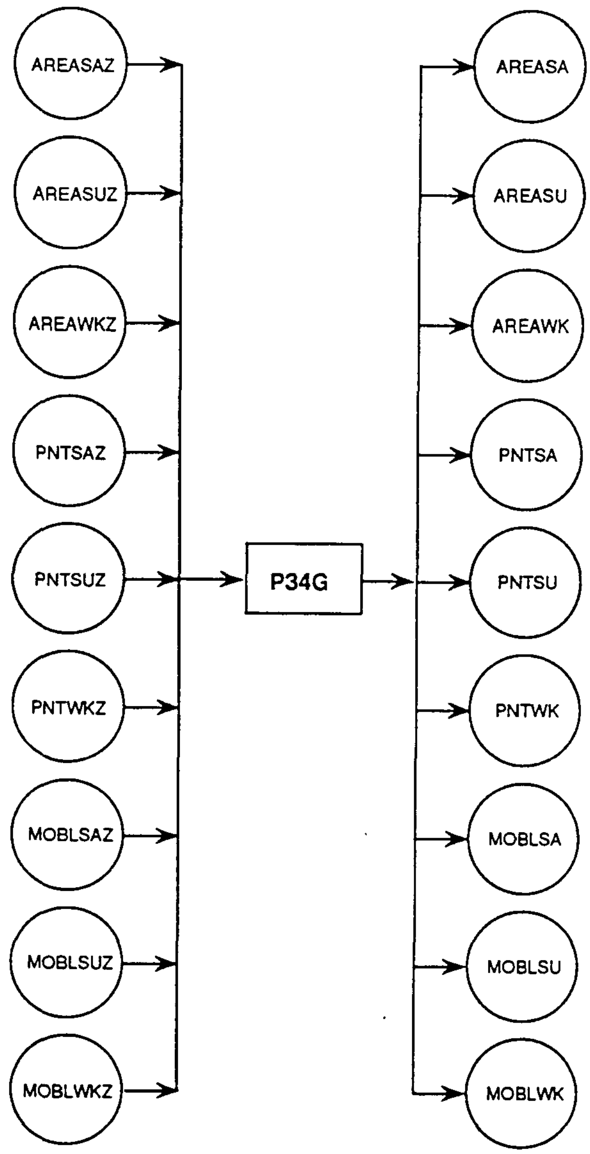


Figure B-30. Processor P34G

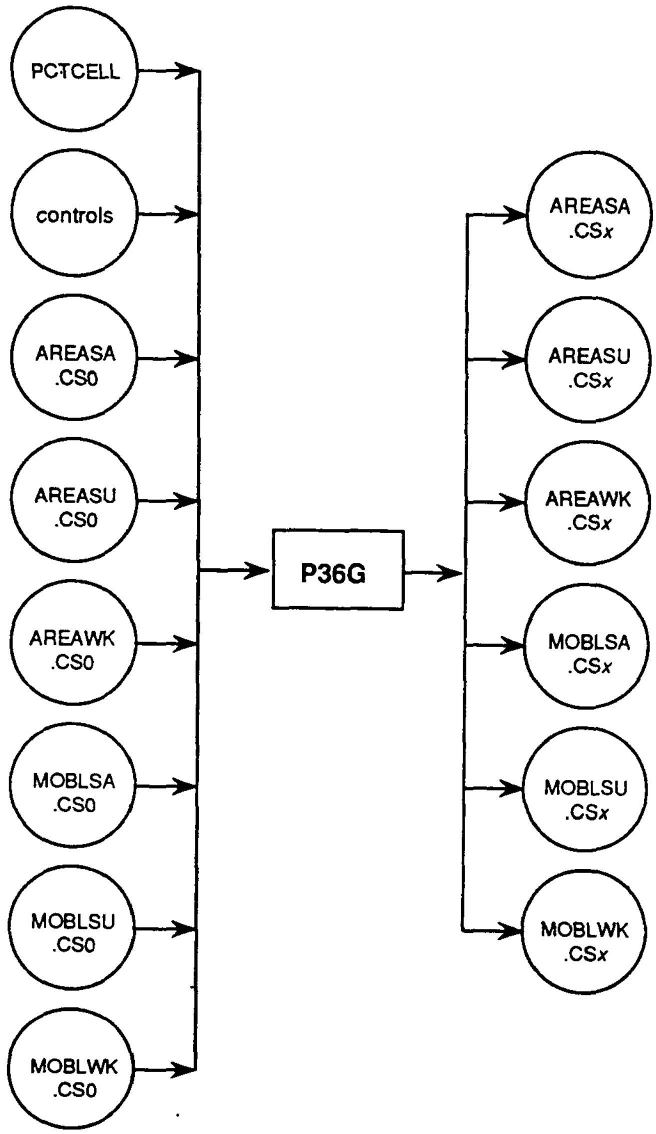


Figure B-31. Processor P36G

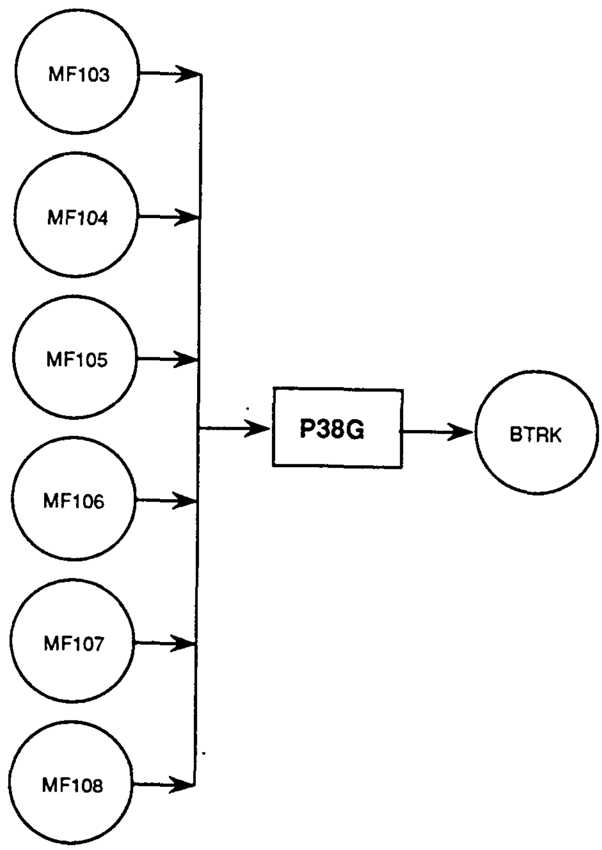


Figure B-32. Processor P38G

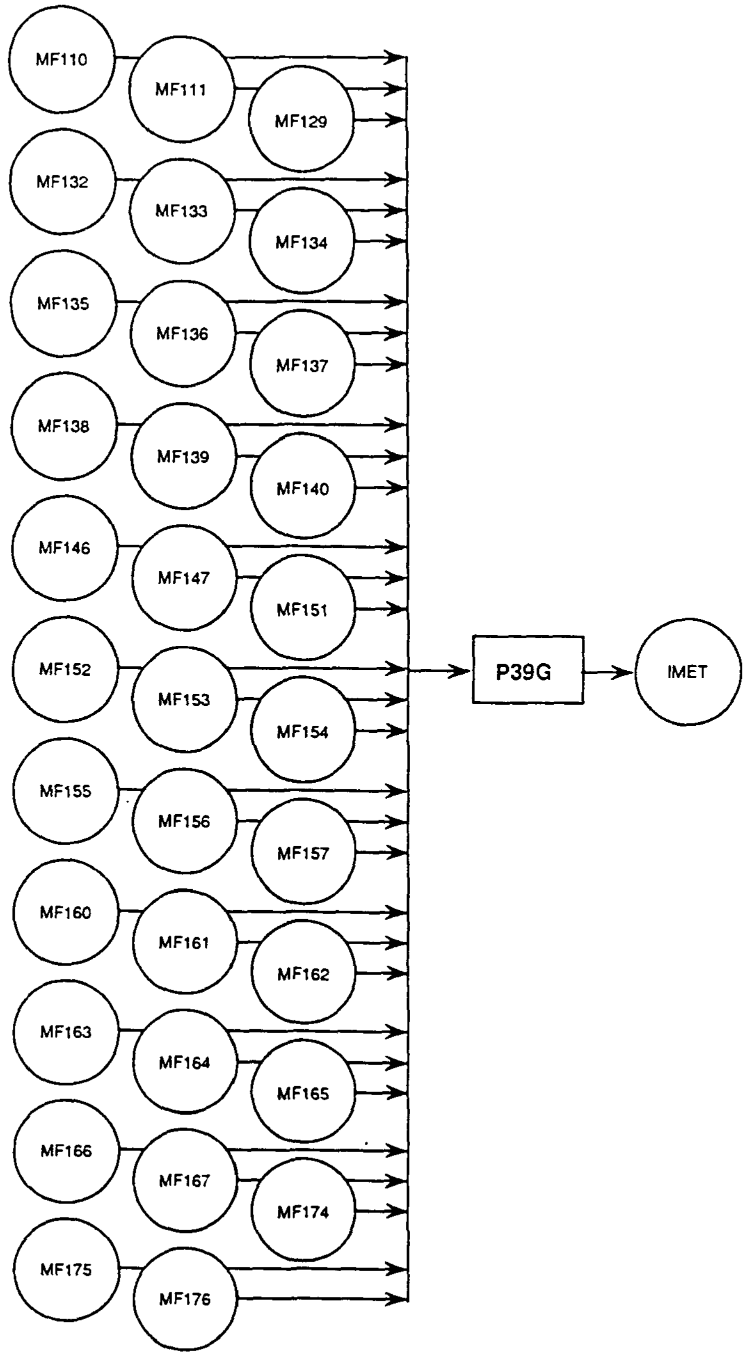


Figure B-33. Processor P39G

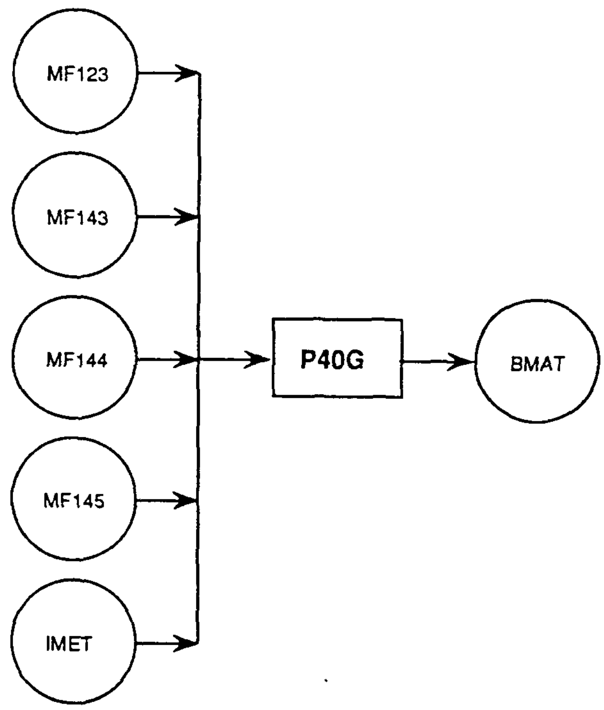


Figure B-34. Processor P40G

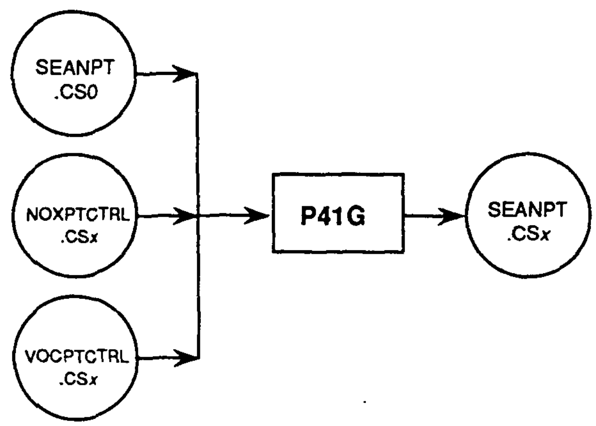


Figure B-35. Processor P41G

APPENDIX C

PROCESSOR SUBPROGRAM HIERARCHY

* ⇒ Subprogram is called zero or more times.
No * ⇒ subprogram is called once only.

Italicized subprograms are utility subprograms; nonitalicized subprograms are processor-specific.

P01G

PROGID
OPDIR2
OPEN2
FLSMRY
NEXTHR
RDFT1
RLEVEL
PARAMS
PSHIFT
* *WRFT1*

Figure C-1. Processor P01G

- P02G
  - *PROGID*
  - *JFILE2*
  - *CELL0*
  - *DATTIM*
  - *CLOCK1*
  - * WRIC
    - OPWRIC
      - *JFILE5*
      - * *TRNSLT*
      - * *WRCHAR*
    - * *TRNSLT*
    - * *WRFILE*

Figure C-2. Processor P02G

```
P03G
    PROGID
    OPDIR2
    OPEN2
  * FLSMRY
    CLRHDR
    RDFT4
    GETYN
    CELL0
    HASH
        JFILE2
      * CHAIN2
  * RDWNDO
  * STINFO
      * CHAIN2
  * WRFT3
  * BGRID1
  * WRFT5
  * NEXTHR
```

Figure C-3. Processor P03G

```
P04G
    PROGID
    OPDIR2
    OPEN2
  * FLSMRY
    JFILE3
    RDFT4
    ZONEIT
    SETUP
    WRFT4
    CELL0
    AVGWAT
        * RDWNDM
            * RDSRFM
                JFILE2
                IOERR
                JULIAN
                GMTLCL
        * BGRID1
    GREG
    RDFT3
    BUOYPR
          RDWNDO
        * STAWAT
        * UNSWAT
            * PSIM
          FILLIN
    WEIGHT
  * WRFT5
    GRBUOY
        * BGRID3
    GRIDGN
    GRBRNS
    RDFT5
    TSCALE
    CELL2
    SOLELV
    CRTELV
        * RADLWV
        * CLDFN
    ISLAND
  * STABLE
    QFLUX
          GAMOVS
    UNSTBL
    NEXTHR
```

Figure C-4. Processor P04G

P05G

- *PROGID*
- *OPDIR2*
- *OPEN2*
- * *FLSMRY*
- * *CELL0*
- *RDFT4*
- SFCMET
  - * *RDWNDO*
  - ABTERM
- *RDFT1*
- TDPGRD
  - * *BGRID1*
- QFORM
- LCLVAL
- VPROFL
- TSHIFT
  - * ABTERM
- BOYANT
  - * ABTERM
- QFIELD
  - BGRID1
  - AGRID1
  - TSMUTH
- TMPRAL
- TOPSET
- * *WRFT5*

Figure C-5. Processor P05G

P06G

- *PROGID*
- *OPDIR2*
- *OPEN2*
- * *FLSMRY*
- * LAKE
    - *JFILE2*
- *JFILE2*
- * *CELL0*
- * *CELL1*
- * *WRFT4*

Figure C-6. Processor P06G

P07G

- *PROGID*
- SETUP7
  - *OPDIR2*
  - *OPEN2*
  - *FLSMRY*
  - • *RDFHD2*
  - *CELL0*
- FLOINIT
- CORFOR
- • *CLRHDR*
- • *RDFT4*
- • CPULOG
  - *CPUTOT*
- • *RDFT5*
- • *DATHR3*
- • H1H0Z
  - • *WRFT5*
- • SIGMAT
  - • *WRFT5*
  - • MINMX2
- • FLOMOD
  - *NEXTHR*
  - BASEVA
    - SLPRES
      - *RDFT3*
      - SMOOTH
    - DPCD
      - GRADX6
      - RADRAG
        - *RDFT1*
    - COLDGR
      - EXTEND
      - *WRFT5*
    - MINMAX
  - BARS
  - • CLOCK
  - • PRIMES
  - • FRAME
  - • SETBCS
  - • SOLVER
  - • SOLVUV
  - • SOLVDH
  - MINMAX
  - *NEXTHR*
  - *WRFT5*
- • *NEXTHR*
- • FLORESET
- *FLSMRY*

Figure C-7. Processor P07G

P08G

- *PROGID*
- *OPDIR2*
- *OPEN2*
- *JFILE2*
- * *FLSMRY*
- * *CELL0*
- *GETYN*
- *RDFT4*
- ALBALF
- * *RDFT5*
- *RDFT1*
- RADPFL
- PROFIL
  - * ABTERM
- CLDRAD
  - * *RDSRF1*
  - * ABTERM
  - ARCEST
- FLXRAD
  - *GREG*
  - CRTEL8
  - QFLUX8
- PBLMIX
- CLDMOD
  - * ABTERM
  - RTIMOD
    - MBAR5
    - INTERP
    - * SPLINE
    - QUDPTS
    - SPLSOL
    - SPLDER
  - * PTOZ
    - ABTERM
- CLDPAR
- * *WRFT5*
- DELTAT

Figure C-8. Processor P08G

```
P09G
  |-- PROGID
  |-- OPDIR2
  |-- OPEN2
  |-- * FLSMRY
  |-- ZEN
  |     |-- CELL0
  |     |-- * RDWNDO
  |     |-- GREG
  |     |-- * SOLAR
  |     |-- BGRID1
  |     |-- * WRFT5
  |-- DEN
        |-- CELL0
        |-- RDFT4
        |-- * RDFT1
        |-- * BGRID3
        |-- RDFT5
        |-- * WRFT5
```

Figure C-9. Processor P09G

P10G

- *PROGID*
- *OPDIR2*
- *OPEN2*
- * *FLSMRY*
- *CELL0*
- * *RDFHD2*
- *RDFT4*
- * RDSTAK
- * SETCLK
- * *RDFT3*
- * *RDFT5*
- * *ADPFMF*
  - *RDFT5*
- * *CELL1*
- * SATEMP
- * *CELLAR*
- * *WRFT5*

Figure C-10. Processor P10G

- FILES
  - *OPDIR2*
  - *OPEN2*
  - *FLSMRY*
  - *JFILE3*
  - • *JUNIT*
  - *DATHR6*
  - • *CELL0*
  - *RDFHD2*
- ALLOC
  - • *GETMEM*
- READIN11
  - • *CELL2*
- P20
  - DVMCLC
  - DSLMAT
- P18
  - • *DATHR3*
  - • *RDFT1*
  - • *NEXTHR*
  - • *RDFT3*
  - • GCALC
- • *RDFT1*
- • *RDFT3*
- • *RDFT5*
- • DVGGRD
  - *BGRID2*
  - • DIVRG
  - • *WRFT5*
  - • *BGRID3*
  - LAVRG
- • RDHTS
  - • *CELL1*
- • UV11
- • *WRFT5*
- • SETUP11
- • GENCLC
- • DVGCLC
- • WRTOBS
  - WRPLOT
- • MEAN
  - WEIGHT11
- • EXPCLC
- • CMPAUX
- • SLMAT
- • XVCCLC
- • INITPT
  - POW
- • CPCONF
- • ALTSOL
  - PROJEC
  - POW
- • CONVRG
  - • TSTFN
    - POW
  - • CPGRAD
  - • HESS
  - • PROJEC
  - • LINMIN
    - • TSTFN
      - POW
    - • CUBMIN
  - • REFRSH
    - PROJEC
    - TSTFN
      - POW
- • FINLPT
  - POW
  - WRPLOT
    - *RDREC*
    - *WRREC*
  - *WRFT5*
- • REPORT
  - CHECK
    - RDPLOT
      - *RDREC*
  - TSTFN
    - POW
- • *NEXTHR*

Figure C-11. Processor P11G

```
P12G
    PROGID
    OPDIR2
    OPEN2
  * FLSMRY
    CELLO
    RDFT4
  * RDFT5
    VOLFLX
    CUSET
    CUFLUX
        * ERF
    ALLFLX
  * WRFT5
    NEXTHR
```

Figure C-12. Processor P12G

- P13G
  - *PROGID*
  - *OPDIR2*
  - *OPEN2*
  - * *FLSMRY*
  - *JFILE2*
  - * SEGSUB
    - *CELL0*
    - * *CELL1*
    - * RBCLIP
  - *WRFT4*

Figure C-13. Processor P13G

```
P14G
    PROGID
    OPDIR2
    OPEN2
  * FLSMRY
  * RDFHD2
    CELL0
    CELL1
    DATHR4
    FSKIP1
    JFILE2
  * WRFT5
    WRFT4
  * ARESEL
        JFILE2
      * KDAY
  * SWAPEM
  * SWATSTK
  * STKMRG
      * PONSEL
            JFILE4
            KDAY
        STKHSH
          * CHAIN2
            JFILE2
      * CHAIN2
  * RDPFMF
        RDFT5
```

Figure C-14. Processor P14G

P15G

- *PROGID*
- *OPDIR2*
- *OPEN2*
- * *FLSMRY*
- *CELL0*
- *RDFHD2*
- *RDFT4*
- * *RDFT5*
- * RST
  - *RDFT4*
  - * *RDFT5*
  - *RDFT3*
  - *NEXTHR*
- * *WRFT5*
- * *NEXTHR*

Figure C-15. Processor P15G

P16G

- *PROGID*
- *OPDIR2*
- *OPEN2*
- * *FLSMRY*
- * *JFILE2*
- *GETYN*
- * ENDPTS
  - * GM2MOD
  - RADTIM
  - WHICH1
  - MBAR25
  - APPROX
    - * FILLER
  - WNDFIX
    - * SMUTHR
- * SFCHRS
  - GM2MOD
  - *POINTS*
- * SFCLEV
- * RWND25
- * COUPLE
- * *WRFT1*

Figure C-16. Processor P16G

```
P17G
────────────┐
            │   PROGID
            │   OPDIR2
            │   OPEN2
            │ * FLSMRY
            │   CELLO
            │   RDPRM2
            │   ──────────────┐
            │                 │ * DATHR3
            │                 │ * RDFT5
            │   RDFT4
            │ * WRFT5
```

Figure C-17. Processor P17G

P19G

	PROGID
	OPDIR2
	OPEN2
*	*FLSMRY*
	GETYN
	CELL0
*	*RDWNDO*
*	*IOERR*
	BGRID1
	WRFT5

Figure C-18. Processor P19G

```
P21G
        PROGID
      * JFILE2
        ADATE
      * GPRIME
                * TOTAL
                * PQCOEF
                * RATED
                * PQ1
```

Figure C-19. Processor P21G

```
P22G
  |-- PROGID
  |-- JFILE2
  |-- CELL0
  |-- DATTIM
  |-- CLOCK1
  |-- * WRBC
  |       |-- OPWRBC
  |       |     |-- JFILE5
  |       |     |-- * TRNSLT
  |       |     |-- * WRCHAR
  |       |-- TRNSLT
  |       |-- * WRFILE
  |-- * TRNSLT
  |-- * CLOCK2
```

Figure C-20. Processor P22G

P23G

- PROGID
- OPDIR2
- OPEN2
- * FLSMRY
- * JFILE2
- CELL0
- * GPRIME
  - * TOTAL
  - * PQCOEF
  - * RATED
  - * PQ1
- * WRFT5

Figure C-21. Processor P23G

```
P24G
  ─────────┐
           │  PROGID
           │* JFILE2
           │* GPRIME
              ─────────┐
                       │* TOTAL
                       │* PQCOEF
                       │* RATED
                       │* PQ1
```

Figure C-22. Processor P24G

P25G

CELLO
PROGID
OPDIR2
OPEN2
* *FLSMRY*
IOERR
CELL1
WRFT4

Figure C-23. Processor P25G

```
P26G
    PROGID
    OPDIR2
    OPEN2
  * FLSMRY
    RDTFAC
          JUNIT
        * OFFNDX
  * CTREFF
          JFILE2
          INDEX1
    RDFT5
  * RDTFAC
  * WRFT5
    PROCMB
        * JFILE2
          CELO
        * INDEX1
        * KDAY
        * WRFT5
```

Figure C-24. Processor P26G

```
P27G
  |-- PROGID
  |-- OPDIR2
  |-- OPEN2
  |-- FLSMRY
  |-- GETYN
  |-- SOLENG
  |     |-- PROGID
  |     |-- CELL2
  |     |-- FLXEIN
  |     |     |-- JFILE2
  |     |-- * RDFT5
  |     |-- CLDATN
  |     |     |-- JFILE2
  |     |     |-- SOLELV
  |     |     |-- BGRID1
  |     |-- PRNTGRD
  |     |-- * WRFT5
  |-- * JFILEX
  |-- JFILE2
  |-- * CORRECT
  |     |-- * RDFT5
  |-- * WRFT5
```

Figure C-25. Processor P27G

```
P29G
        PROGID
        CELL0
      * RDCONV
              * OPDIR2
              * OPEN2
              * FLSMRY
              * DATHR
              * DATHR3
              * AREAC
              * DAY
              * RDFT5
              * CONWIN
              * CONDIF
                      * CELL2
              * DIFLAM
              * DIFLPHI
              * VOLCAL
              * LNRHVO
              * DLNLAM
              * DLNPHI
      * POLY
      * BQ3LAM
      * BQ3PHI
      * WRFT5
        FLSMRY
```

Figure C-26. Processor P29G

- P31G
  - *PROGID*
  - *JFILE4*
  - * SRTREC
    - *GETMSG*
    - * RDSAPT
      - *JFILE2*

Figure C-27. Processor P31G

```
P32G
            PROGID
            OPDIR2
            OPEN2
          * FLSMRY
            GETMEM
            RDFHD2
            RDFT3
          * RDFT5
            DIFTRB
                        BGRID1
          * WRFT5
            NEXTHR
```

Figure C-28. Processor P32G

P33G

- *PROGID*
- *OPDIR2*
- *OPEN2*
- * *FLSMRY*
- *CELL0*
- *RDFT4*
- * RDSTAK
  - *JFILE2*
- SETCLK
- *RDFT3*
- * *RDFT5*
- * *CELL1*
- SATEMP
- * *CELLAR*
- * SORTAR
- SORTAI
- *WRFT4*

Figure C-29. Processor P33G

P34G

PROGID
JUNIT
JFILE4
* *ARSHFT*
JFILE2
* *MBSHFT*
JFILE2
* *PTSHFT*
JFILE4

Figure C-30. Processor P34G

```
P36G
    RDPCTC
    ARPROC
        RDARCS
        ARCTRL
      * AGRID
            RDATE
    MBPROC
        RDMBCS
        MBCTRL
      * AGRID
        RDATE
```

Figure C-31. Processor P36G

```
P38G
    PROGID
    OPDIR2
    OPEN2
    FLSMRY
    JFILE6
    CELLO
  * DATHR5
  * TIMEBT
        * CLOCK1
        * CLOCK2
    WRHDBT
        * TRNSLT
        * WRCHAR
        * ERROUT
  * WRBTRK
        * TRNSLT
        * WRFILE
  * INTRPD
        * RDDIFF
              * DATHR3
              * RDFT5
  * RDWIND
        * RDFT5
```

Figure C-32. Processor P38G

```
P39G
    PROGID
    OPDIR2
    OPEN2
    FLSMRY
    JFILE2
    JFILE3
    VRSPIM
        * RDFHD2
        * INDEX1
    CELL0
  * DATHR5
  * TIMEIM
        * CLOCK1
        * CLOCK2
    WRHDIM
        * WRCHAR
        * ERROUT
  * WRIMET
        * WRFILE
  * INTRPM
        * RDMET
            * DATHR3
            * RDFT5
  * INTRPS
        * RDSPC
            * DATHR3
            * RDFT5
  * LISTIM
```

Figure C-33. Processor P39G

P40G

- *PROGID*
- *OPDIR2*
- *OPEN2*
- *FLSMRY*
- VRSPBM
  - • *RDFHD2*
  - • *INDEX1*
- *CELLO*
- • TIMEBM
  - • *CLOCK1*
  - • *CLOCK2*
- • RDIMET
  - OPIMET
    - RDIM
  - POIMET
    - • *CLOCK1*
    - RDIM
  - • RDIM
- VERMET
  - • *INDEX1*
- SPCEXP
- • *DATHR5*
- • WRBMAT
  - • *JFILE6*
  - • WRHDBM
    - • *TRNSLT*
    - • *WRCHAR*
    - • ERROUT
  - • *TRNSLT*
  - • *WRFILE*
- • TRNSEM
  - • *RDEMIS*
    - • *DATHR3*
    - • *RDFT5*
- • BMCOEF

Figure C-34. Processor P40G

```
P41G
 ├─ RDPTCS
 │    └─ JFILE2
 ├─ JFILE2
 ├─ JFILEX
 └─ * INDEX1
```

Figure C-35. Processor P41G

APPENDIX D

EMISSION CONTROL STRATEGIES

D.1 ROM2.0: SCENARIOS FOR NEROS

In addition to base cases, individual emission control strategies for earlier versions of the ROM than ROM2.1 are defined below. Some have been implemented alone, and some have been used in combination, such as VOC2 + NOX3.

VOC0	Base emissions projected to 1987 including the effects of existing controls. Largest changes in VOC emissions; some smaller changes to NO_x emissions.
VOC1	Base emissions from 1980 reduced to include the effects of existing VOC controls in 1982 State Implementation Plans. Only VOC emissions are affected.
VOC2	VOC1 plus additional reductions expected by 1995 due to the Federal Motor Vehicle Control Program (FMVCP).
VOC3	VOC2 plus additional reductions in the Northeast Corridor, Pittsburgh, Cleveland, and Detroit.
VOC4	VOC3 including a higher level of reductions, up to 90% control in the Northeast Corridor.
NOX1	NO_x controls on utility boiler emissions from 1980 resulting in a 39% cut in utility emissions (11% cut in region-wide NO_x).
NOX2	Utility/industrial boiler plus FMVCP NO_x controls only in Detroit (27% resulting cut in NO_x emissions) and the Northeast Corridor (22% resulting cut). Overall 10% cut in region-wide NO_x.
NOX3	Utility/industrial boiler plus FMVCP NO_x controls applied regionally. (This resulted in a 22% cut in NO_x emissions for the Northeast Corridor, and a 27% cut in region-wide NO_x for the northeast United States domain.)
NOX4	NOX3 except for a 70% cut in point source NO_x emissions in the Northeast Corridor.

D.2 ROM2.1: SCENARIOS FOR ROMNET

D.2.1 1985 Base Case

This scenario includes summer season weekday, Saturday, and Sunday anthropogenic emissions from the 1985 NAPAP Version 2.0 inventories for Volatile Organic Compounds (VOC), Oxides of Nitrogen (NO_x), and Carbon Monoxide (CO). Mobile source vehicle emissions were calculated with MOBILE4 (a model that is a part of the FREDS [Part 1, Section 4]) and were adjusted by grid for daily average temperature and temperature range. (Running loss, standing evaporative, and excess evaporative mobile emissions were included.) Stationary source VOC emissions from six categories (gasoline marketing, bulk terminals/plants, gasoline vessels, architectural coatings, miscellaneous solvent use, and TSDF's [Treatment, Storage, and Disposal Facilities]) were adjusted to reflect average daily temperatures or activity levels typical of the high ozone episodes being modeled in ROMNET. Existing controls were accounted for in these inventories. Also included in this scenario were natural emissions of biogenic VOC from vegetation biomass and geogenic NO_x from grassland soils. Natural emissions were generated using the Biogenic Emissions Inventory System (BEIS). Both anthropogenic and natural emissions were speciated using the Carbon Bond Mechanism Version 4.2

D.2.2 2005 Future Base Case

In this scenario, 1985 Base Case anthropogenic emissions were projected to the year 2005 using emissions-category specific growth factors. Non-utility stationary source growth factors were obtained from the Bureau of Economic Analysis (BEA). Utility growth rates were obtained using the Advanced Utility Simulation Model (AUSM) with the EPA 1987 Interim Base Case economic scenario. Adjustments to stationary source growth rates were made by individual States. Mobile source vehicle miles travelled (VMT) growth rates were taken from Federal Highway Administration projections. Included in this scenario were existing Federal and State emissions controls (1985 State-specific control efficiency, penetration, and rule effectiveness levels), and a Reid Vapor Pressure (RVP) of 9.0 psi. In addition, NO_x emissions were controlled on a within-State basis down to 1985 emissions levels. Controls were applied in the Canadian portion of the domain. Natural emissions for this scenario were identical to those in the 1985 Base Case.

D.2.3 CS01 - 2005 Maximum Technology Strategy

In this scenario, controls are applied on a source-category specific basis to emissions of VOC in the 2005 Future Base Case. Emissions of anthropogenic NO_x and natural emissions are the same as in the 2005 Future Base Case scenario. The VOC controls represent the highest efficiencies possible from known control technologies. Controls are applied to 31 stationary area source categories and 39 major groups of point source categories. Both penetration and rule effectiveness of controls are assumed to be 100%. Controls on mobile sources include an "enhanced" inspection/maintenance program and "on-board" controls for refueling emissions. In this scenario, the above controls are applied in all areas of the modeling domain except Canada, where emissions are at the 2005 Future Base Case levels.

D.2.4 CS02 - 2005 Northeast Corridor Maximum Technology Strategy

In this scenario, the maximum technology controls included in the CS01 scenario are applied only to counties in the Northeast Corridor. Emissions in all other areas of the region are identical to those in the 2005 Future Base Case.

D.2.5 CSO3 - 2005 Nonattainment Maximum Technology Strategy

In this scenario, the maximum technology controls included in the CS01 scenario are applied to counties in the Northeast Corridor and to all counties in nonattainment areas in the region.

D.2.6 CS04 - Rule Effectiveness Sensitivity Test

In other ROMNET strategies, we have assumed that controls are 100% effective. In actuality, the level of emissions reduction from a control device or procedure is less effective than the control efficiency; this is due to various routine operating constraints. This scenario adjusts the control efficiencies in one of the other ROMNET strategies; this reduces the effectiveness of controls and obtains a more realistic assessment of the benefits of the strategy. CS04 will be run by altering efficiencies from another ROMNET strategy that resulted in ozone concentrations of less than 0.12 ppm across the domain.

D.2.7 CSO5 - Proposed Clean Air Act Legislation Strategy

This strategy includes controls on stationary and mobile sources of VOC and NO_x (including controls for Acid Rain) as prescribed in H.R.3030, modified to include the California Phase I tail pipe standards in the 10/04/89 Waxman-Dingell compromise proposal. In addition to the application of specific control measures, VOC emissions from all source categories were reduced in an across-the-board manner in individual urban areas to meet the 3% per year emissions reduction target calculated between 1990 and the attainment date. Canadian emissions are the same as in the 2005 Future Base Case. Natural emissions for this scenario are identical to those in the 1985 Base Case.

D.2.8 Biogenics Sensitivity Tests

These scenarios are designed to test the sensitivity of ozone predictions to uncertainty in biogenic emissions for base case and selected post-control scenarios. Included are ROM applications for the 1985 Base Case, CS01, and CS05 scenarios with biogenic emissions altered to reflect the possible range of uncertainty (± a factor of 3). Each of the three anthropogenic scenarios are run with (1) biogenic emissions reduced by a factor of 3 ("Low" biogenics) and (2) biogenic emissions increased by a factor of 3 ("High" biogenics). In all scenarios, biogenic emissions are altered in an across-the-board manner from the "best available science" estimates generated by the Biogenic Emissions Inventory System (BEIS). The following are the specific scenarios modeled:

1985L: 1985 anthropogenics - Low biogenics
1985H: 1985 anthropogenics - High biogenics
CSO6:CS05 anthropogenics - Low biogenics

CS07:CS01 anthropogenics - Low biogenics
CS08:CS05 anthropogenics - High biogenics
CS09:CS01 anthropogenics - High biogenics

D.2.9 New 1985 Base Case

This scenarios is an update to the 1985 Base Case to reflect MOBILE4 emissions factors. Point- and area-source emissions, Canadian emissions and natural emissions remain unchanged from the previous 1985 Base Case.

D.2.10 New 2005 Future Base Case

This scenario contains several modifications to the previous 2005 Base Case. These include:

- removal of utility and industrial combustion NO_x reductions previously used to achieve a "no growth" scenario for NO_x;
- a 5% reduction in utility NO_x emissions to reflect improved efficiencies in the generation and use of electricity;
- revised mobile emissions to reflect MOBILE4 and new Federal evaporative test procedures; and
- corrections and revisions to several methodologies.

D.2.11 CS10 - Maximum Technology NO_x Controls and Enhanced Maximum Technology VOC Controls

This strategy was built from the new 2005 Future Base Case and includes:

- mobile source tail pipe standards in Senate Bill 1630 and on-board controls to passenger vehicles;
- banning residential wood stove use and open burning;
- controls on off-highway vehicles;
- reduction in RVP to 7.0 from 9.0 psi;
- new controls on previously uncontrolled VOC point sources;
- more stringent controls on three point-source categories (aircraft coating, SOCMI [Synthetic Organic Chemical Manufacturing Industry] reactors and SOCMI distillation processes);
- Selective Catalytic Reduction (SCR) controls to all utilities;
- Low-NO_x Burners (LNB) (or equivalent) controls to all industrial and institutional point sources; and
- all other stationary-source VOC controls as specified in CS01.

D.2.12 CS11 - Maximum Technology NO_x Controls

This strategy is designed to examine the effects of NO_x controls only. NO_x emissions are the same as in CS10. VOC and CO emissions are at the new 2005 levels.

D.2.13 CS12 - Enhanced Maximum Technology VOC Controls

This strategy is designed to examine the affects of the new enhanced VOC control measures. VOC and CO emissions are the same as CS10. NO_x emissions are at the new 2005 levels.

D.2.14 CS13 - Point-Source Maximum Technology NO_x Controls

This strategy is designed to examine the effects of controlling major point-sources of NO_x emissions (utilities, industrial and institutional boilers) with enhanced maximum technology VOC controls measures. NO_x point-source emissions and all VOC and CO emissions are the same as CS10. Mobile- and area- NO_x emissions are at the new 2005 levels.

D.2.15 CS14 - Mobile-Source Maximum Technology NO_x Controls

This strategy is designed to examine the effects of mobile-source NO_x controls with enhanced maximum technology VOC control measures. NO_x mobile emissions and all VOC and CO emissions are the same as CS10. Point- and area- NO_x emissions are at the new 2005 levels.