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

THE REGIONAL OXIDANT MODEL (ROM) USER'S GUIDE
PART 1:
THE ROM PREPROCESSORS

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

NOTICE

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

FOREWORD

Within the Atmospheric Research and Exposure Assessment Laboratory, the Atmospheric Sciences Modeling Division (ASMD) conducts intramural and extramural research programs to develop predictive models on local, regional, and global scales for assessing changes in air quality and air pollutant exposures, as affected by changes in ecosystem management and regulatory decisions. The ASMD research activities are designed to:

- develop, evaluate, and validate air quality simulation and photochemical and meteorological/climatological models that describe air quality and atmospheric processes affecting the ultimate disposition of airborne pollutants on local, urban, and regional scales;
- perform and direct interagency research necessary to support ecological risk assessment by producing quantitative evaluations of changes in regional air quality due to global climate fluctuations;
- develop and apply fluid modeling techniques that describe atmospheric physical processes affecting buoyant and dense gas pollutant dispersion under unique meteorological situations, terrain features, and source configurations;
- implement modeling software design and systems within stated requirements of quality control and assurance to support atmospheric dispersion modeling, meteorological/climatological research, and predictive applications;
- for the user community, develop and provide evaluated improvements to air quality simulation models, meteorological models, pollutant exposure models, and related model input parameters;
- provide technical guidance on applying and evaluating air quality simulation models that are used to assess, develop, or revise air pollution control strategies for attainment/maintenance of ambient air quality standards;
- provide applications support with available models of monitoring and exposure field studies, other atmospheric research programs, and receptor-oriented research programs; and
- maintain communications with the EPA Research, Program, and Regional Offices, and with the international and national scientific community to incorporate and disseminate state-of-the-science developments pertaining to meteorological/climatological aspects of environmental quality and exposure assessment.

The Regional Oxidant Model (ROM) is a 3½-layer Eulerian grid model that calculates hourly concentrations of ozone and other chemical species for episodes extending up to about one month long. The physical processes that ROM simulates include photochemistry, spatially- and temporally-varying wind fields, nocturnal jets and temperature inversions, terrain effects, dry deposition, cumulus cloud effects, and emissions of anthropogenic and biogenic ozone precursors. Dispersion processes are based on atmospheric turbulence principles and K-theory closure.

There are several limitations inherent in the model. Among the most important of these are: (1) the ROM supports no pollutant wet removal processes; (2) the physics are standardized to represent summer meteorological conditions; (3) the ROM does not take into account any aqueous-phase chemistry; (4) cumulus cloud processes are such that when a cloud is created in a grid cell, it remains there for a full hour (i.e. cloud physics are not considered), and neither the cloud nor its pollutants are advected; (5) the ROM actually consists of three linked, two-dimensional models, and, as such, cannot be expected to model regions that contain high, complex, mountainous terrain; and (6) the ROM is currently configured to run on domains in eastern North America on the scale of 1000 km^2. This regional scale is germane since long-range transport of precursors will have a significant effect on local ozone concentrations. A major limitation imposed by the regional scale, multilayer, multispecies approach concerns the amount of CPU time required.

We advise users of ROM to interpret its results in terms of analysis of different emission control strategies on ozone concentration, rather than assuming the results to be an accurate snapshot of a specific pollution event.

ABSTRACT

The Regional Oxidant Model (ROM) determines hourly concentrations and fates of ozone and 34 other chemical species over a scale of 1000 km x 1000 km for ozone "episodes" of up to one month's duration.

The model structure, based on phenomenological concepts, consists of 3½ layers. The surfaces separating the layers respond to variations in space and time in the meteorological phenomena simulated in each layer. The model simulates many physical and chemical processes that affect the motion and distribution of chemical concentrations; among these are: horizontal transport, photochemistry, nighttime wind shear and nocturnal jet; cumulus cloud effects and mesoscale vertical motion; terrain and mesoscale eddy effects; subgrid scale chemistry processes, natural sources of hydrocarbons, NO_x and stratospheric ozone; and dry deposition.

The ROM is a complex model that requires users to have expertise in photochemical grid modeling. Meteorologists, engineers, and computer scientists familiar with this type of modeling will find this User's Guide relevant and helpful for running the ROM.

PREFACE

One of the research activities of the Atmospheric Sciences Modeling Division (ASMD) focuses on the development, evaluation, validation, and application of air quality simulation, photochemical, and meteorological models capable of describing air quality and atmospheric processes affecting the disposition of airborne pollutants, on scales ranging from local to global. Within the Division, the Modeling Systems Analysis Branch (MSAB) supports the other Branches in the ASMD by providing the computer programming and system analysis needed in the development of the models. The MSAB operates the Facility for Advanced Research Model Operation and Analysis (Research Modeling Facility) to provide proper expertise in the application and interpretation of advanced dispersion models for EPA regulatory applications, and establishes definitive scientific standards for model evaluation and policy analysis that are consistent with standards followed in the research and model development efforts.

The Regional Oxidant Model (ROM) simulates regional scale photo-oxidant motion and three-dimensional distribution using meteorological observations, turbulence theory, and chemical rates of reaction. Although ozone is the principal modeled species, 34 other chemical concentrations may be predicted.

Comments and suggestions regarding this publication should be directed to the address below:

Chief, Modeling Systems Analysis Branch
Atmospheric Sciences Modeling Division (MD-80)
U.S. Environmental Protection Agency
Research Triangle Park, NC 27711

GLOSSARY

DDD	Three-digit Julian day number
episode	A model period made up of several three-day scenarios
EST	Eastern Standard Time (GMT - 5 hours); all model times are in EST unless otherwise stated
GMT	Greenwich Mean Time
HH	Two-digit hour on a 24-hour clock, e.g., 23 = 11 p.m. EST
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
NCCIBM1	The EPA's IBM 3090 computer, used for running the ROM Core Model
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

This page is intentionally left blank.

TABLE OF CONTENTS

LIST OF TABLES

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

LIST OF FIGURES

This page is intentionally left blank.

SECTION 1

INTRODUCTION

Although air quality levels in the United States for most criteria pollutants have improved over the past decade, harmful levels of photochemical smog still persist in many urban areas. Concentrations of ozone, a major component of photochemical smog, often exceed the air pollution standard established by the Clean Air Act. Implementing VOC and NO_x emission control programs can bring some areas closer to the primary ozone standard, but the effectiveness of control strategies must be evaluated before they are actually applied. Because mathematical modeling is a very powerful tool, the U.S. Environmental Protection Agency (EPA) has developed the Regional Oxidant Model (ROM) to predict the ozone concentration changes that would result from specified emissions changes for a given region.

The initial development of a regional (≈1000km) air quality simulation model began in the late 1970's after the realization that photochemical smog often extended beyond individual urban areas to entire sections of the United States. Interstate transport of ozone (O_3) and its precursors was observed during field programs, especially in the northeast of the country. Since multiday chemical effects and long-range transport of ozone and its precursors was beyond the scope of the existing urban-scale photochemical models, the need for an appropriate simulation model to test the effectiveness of particular emission control strategies on regional and urban airshed ozone concentrations became clear. The Regional Oxidant Model (ROM) has been developed and enhanced over the past eight years at the EPA in response to this need. The first generation ROM (ROM1.0) became operational in 1984. The initial model formulation and algorithm testing is documented in a three-part volume titled *A Regional Scale (1000 km) Model of Photochemical Air Pollution*: Part 1 - *Theoretical Formulation*; Part 2 - *Input Processor Network Design*; Part 3 - *Tests of Numerical Algorithms*.[1]

The ROM is a sophisticated regional-scale model that predicts hourly ozone concentrations for episodes extending up to about a month in duration; each episode is modeled as a series of three-day executions. The model domain, divided into a set of grid cells approximately 19 km by 19 km each, encompasses an area on the order of 1000 km by 1000 km. For much of eastern North America, the ROM can model the regional variability of the chemical and physical processes that affect photochemically-produced ozone concentrations on a regional scale. The ROM models:

1. Lamb, 1983, Lamb, 1984a, and Lamb and Laniak, 1985, respectively.

- horizontal winds;
- the photochemistry of airborne chemical species (including the nighttime chemistry of products and precursors of photochemical reactions);
- nocturnal jets and nocturnal stable stratifications;
- cumulus cloud effects on vertical mass transport and photochemical reaction rates;
- mesoscale vertical motion and eddy effects;
- terrain effects on horizontal flows, and on the deposition and diffusion of chemical species;
- sub-grid-scale chemical processes (resulting from emissions smaller than the model's grid can resolve);
- emissions of biogenic and anthropogenic ozone precursors; and
- dry removal processes (including the dry deposition of ozone).

The ROM system is composed of a Core Model, which solves the sets of equations that describe the above processes, and a series of over 30 processors that prepare the input data needed by the core model. ROM 1.0, the first version of the ROM, emerged in 1984 and was used for a limited set of applications for the Northeast Corridor Regional Modeling Project. ROM 2.0 became operational in 1987; it was created from ROM 1.0 by changing or adding several features: (1) The Demerjian chemical mechanism was replaced with version 4.0 of the Carbon Bond Mechanism (CBM 4.0), which simulates some 70 reactions among 28 chemical species. (2) Biogenic hydrocarbon emissions were added to the list of emission types modeled; ROM 1.0 modeled only anthropogenic emissions. (3) The code was modified to allow atmospheric layer thicknesses to vary over space and time; in ROM1.0, these thicknesses remained constant. (4) The ability to simulate the effects of nocturnal jets and nighttime inversions was added.

Application requirements for EPA's Regional Ozone Modeling for Northeast Transport (ROMNET) project have prompted us to upgrade ROM2.0 to produce ROM2.1 (Young *et al.*, 1989). For the ROMNET application, we have expanded the model's domain in the northeastern U.S. from 60 by 42 cells to 64 by 52 cells in order to include major urban emitters in Ohio and Virginia. As a result, the design of ROM2.1 allows users to increase or reduce the numbers of columns and rows in the grid more easily than before. ROM2.1 also can be adapted more easily to other modeling domains in eastern North America. Some of the other modifications include an updated biogenic hydrocarbon processor; an improved wind fields processor; an upgraded Carbon Bond Mechanism (CBM 4.2) in the ROM's chemistry solver; expanded use of buoy data and the use of mobile-source emissions data; and changes that allow the ROM system to use computer resources more efficiently. We have also added features that should allow outside users to apply the ROM more easily.

The Regional Oxidant Model (ROM) User's Guide is a four-part document: Part 1 - The ROM Preprocessors (this volume); Part 2 - The ROM Processor Network (EPA-600/8-90/083b); and Part 3 - The Core Model (EPA-600/8-90/083c). This volume, Part 1, presents the steps necessary to prepare raw emissions and meteorology data that are used by the Processor network. In Section 2, we present a broad overview of the

ROM. Beginning with Section 3, we describe the various components of the preprocessing operations: Section 3 itself describes meteorology preprocessing; Section 4 introduces the anthropogenic emissions database, and Section 5 describes anthropogenic emissions preprocessing; Section 6 describes biogenic emissions preprocessing; and Section 7 describes the remaining miscellaneous preprocessing requirements.

In Part 2 of this User's Guide, Section 1 discusses the five *standard file types* that are used in the ROM and the processor network. Section 2 presents the ROM Processor Network. Section 3 discusses some aspects of regional dependencies, and describes modeling domain characteristics. Section 4 consists of references and a bibliography pertinent to the ROM. There are four appendices at the end of Part 2: Appendix A consists of a list of domain dimensions and cell sizes that are currently (or in the near future, will become) operational. Appendix B shows the input and output data files for all processors, and Appendix C illustrates the processor's subprogram hierarchy. Appendix D describes previous ROM2.1 emissions control strategy applications.

The ROM User's Guide, Part 3, describes the structure and execution of the Core Model.

The code of the ROM preprocessors, processor network, and Core Model is available on magnetic tape. The ROM User's Guide, Part 4 (EPA-600/8-90/083d), accompanies the code, and consists of a set of benchmark data sets as well as a tutorial to guide you through the code installation and benchmark tests.

Note that work continues to upgrade the ROM; while the revised code may be made available to interested users, no updates to this User's Guide are planned at present.

This page is intentionally left blank.

SECTION 2

AN OVERVIEW OF THE REGIONAL OXIDANT MODEL (ROM)

2.1 A GENERAL DESCRIPTION OF THE MODELING SYSTEM

The system under study by the ROM is the set of chemical and physical processes resulting in the transport and transformation of ozone (O_3) on spatial scales of 1000 km. Ozone is produced by chemical reactions in the presence of sunlight among anthropogenic, biogenic, and geogenic emissions of nitrogen oxides and hydrocarbons. The system is three-dimensional and complex. The relationship of O_3 production to emitted precursor species is nonlinear.

2.2 GENERAL MODEL CHARACTERISTICS

The ROM was designed to simulate most of the important chemical and physical processes that are responsible for the photochemical production of O_3 over a domain of 1000 km and for episodes of typically around 15 days in duration. These processes include (1) horizontal transport, (2) atmospheric chemistry and subgrid-scale chemical processes, (3) nighttime wind shear and turbulence associated with the low-level nocturnal jet, (4) the effects of cumulus clouds on vertical mass transport and photochemical reaction rates, (5) mesoscale vertical motions induced by terrain and the large-scale flow, (6) terrain effects on advection, diffusion, and deposition, (7) emissions of natural and anthropogenic ozone precursors, and (8) dry deposition. The processes are mathematically simulated in a three-dimensional Eulerian model with $3^1/_2$ vertical layers, including the boundary layer and the capping inversion or cloud layer.[1] Horizontal grid resolution is $^1/_4°$ longitude by $^1/_6°$ of latitude, or about 18.5 km × 18.5 km. Current model domains include the northeastern United States and the southeastern U.S./Gulf Coast area.

2.2.1 Physical Processes within Layers 0, 1, 2 and 3

The meteorological data are used to objectively model regional winds and diffusion. The top three model layers are prognostic (predictive) and are free to locally expand and contract in response to changes in the physical processes occurring within them. During an entire simulation period, horizontal advection and diffusion and gas-phase chemistry are modeled in the upper three layers. The bottom layer, layer 0, is a shallow diagnostic surface layer designed to approximate the subgrid-scale effects on chemical reaction rates from a spatially heterogeneous emissions distribution. ROM pre-

1. Layer 0, the "half" layer, has a layer thickness that is always $^1/_{10}$ of the thickness of layer 1; the thicknesses of layers 1, 2, and 3 vary.

dictions from layer 1 are used as surrogates for surface concentrations. The time scale of output concentrations is 30 minutes, although typically 1- and 8-hour daytime averages are used for analysis in the public policy sector. Figure 2.1 shows the ROM layers during the day and at night, and describes some of their features.

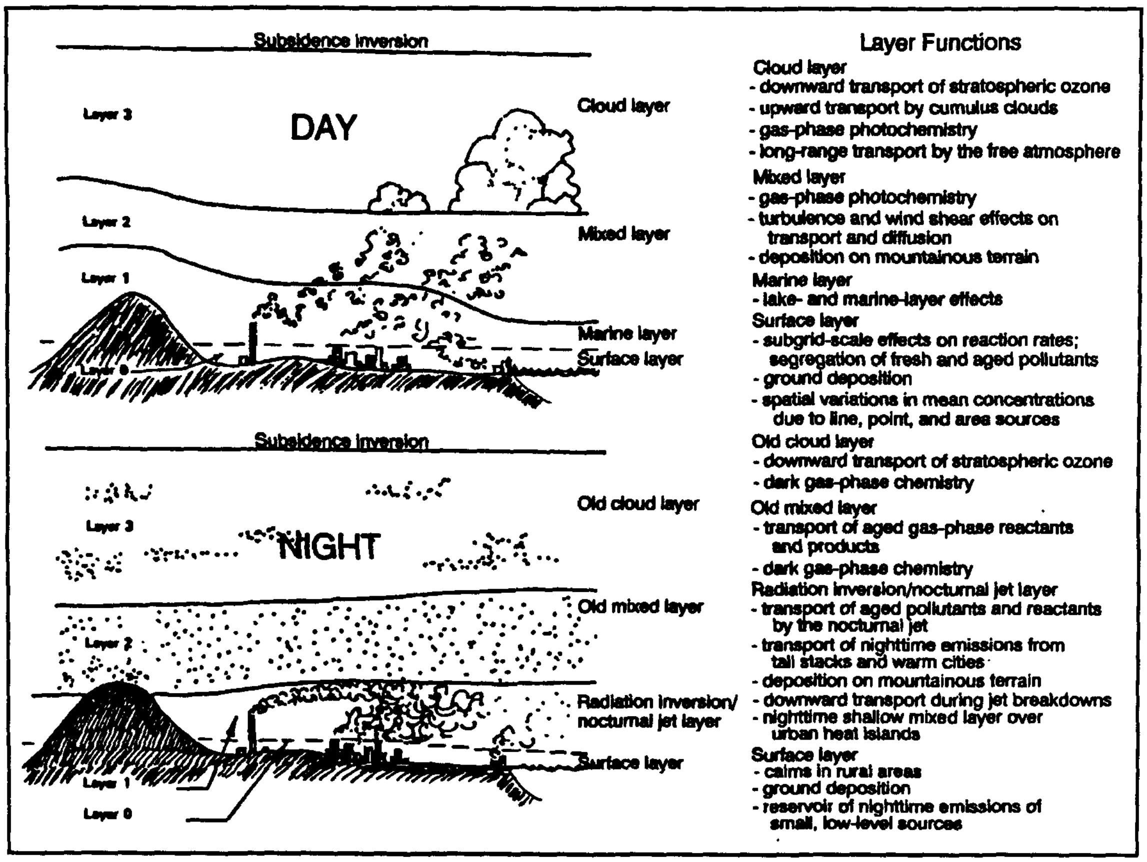

Figure 2.1. The three-and-a-half layers of the ROM.

Layers 1 and 2 model the depth of the well-mixed layer during the day. Some special features of layer 1 include the modeling of (1) the substantial wind shear that can exist in the lowest few hundred meters above ground in local areas where strong winds exist and the surface heat flux is weak, (2) the thermal internal boundary layer that often exists over large lakes or near sea coasts, and (3) deposition onto terrain features that protrude above the layer. At night, layer 2 represents what remains of the daytime mixed layer. As stable layers form near the ground and suppress turbulent vertical mixing, a nocturnal jet forms above the stable layer and can transport aged pollutant products and reactants considerable distances. At night, emissions from tall stacks and warm cities are injected directly into layers 1 and 2. Surface emissions are specified as a mass flux through the bottom of layer 1.

During the day, the top model layer, layer 3, represents the synoptic-scale subsidence inversion characteristic of high ozone-concentration periods; the base of layer 3 is typically 1 to 2 km above the ground. Relatively clean tropospheric air is assumed to exist above layer 3 at all times. If cumulus clouds are present, an upward flux of O_3 and precursor species is injected into the layer by penetrative convection. At night, O_3 and the remnants of other photochemical reaction products may remain in this layer and be transported long distances downwind. These processes are modeled in layer 3.

When cumulus clouds are present in a layer 3 cell, the upward vertical mass flux from the surface is partially diverted from injection into layer 1 to injection directly into the cumulus cloud of layer 3. In the atmosphere, strong thermal vertical updrafts, primarily originating near the surface in the lowest portion of the mixed layer, feed growing fair weather cumulus clouds with vertical air currents that extend in one steady upward motion from the ground to well above the top of the mixed layer. These types of clouds are termed "fair weather cumulus" since atmospheric conditions are such that they do not grow to the extent that precipitation forms. The dynamic effects of this transport process and daytime cloud evolution can have significant effects on the chemical fate of pollutants. For example, fresh emissions from the surface layer can be injected into a warm thermal and rise, essentially unmixed, to the top of the mixing layer where they enter the base of a growing cumulus cloud. Within the cloud, the chemical processes of ambient pollutant species are suddenly altered by the presence of liquid water and the attendant attenuation of sunlight. The presence of fair weather cumulus clouds implies that the atmosphere above the earth's boundary layer is too stably stratified for thermals to penetrate higher. In this case the air comprising the tops of these clouds returns to the mixed layer and is heated on its descent since it is being compressed by increasing atmospheric pressures. Ultimately, the air again arrives at the surface level where new emissions can be injected into it and ground deposition may occur, and the process may begin again. The time required for one complete cycle is typically 30 to 50 minutes with perhaps one-tenth of the time spent in the cloud stage.

Within the ROM system, a submodel parameterizes the above cloud flux process and its impact on mass fluxes among all the model's layers. In the current implementation of the chemical kinetics, liquid-phase chemistry is not modeled, and thus part of the effects from the cloud flux processes are not accounted for in the simulations. Future versions of the chemical kinetics may include liquid-phase reactions. The magnitude of the mass flux proceeding directly from the surface layer to the cloud layer is modeled as being proportional to the observed amount of cumulus cloud coverage and inversely proportional to the observed depth of the clouds.

Horizontal transport within the ROM system is governed by hourly wind fields that are interpolated from periodic wind observations made from upper-air soundings and surface measurements. During the nighttime simulation period, the lowest few hundred meters of the atmosphere above the ground may become stable as a radiation inversion forms. Wind speeds increase just above the top of this layer, forming the nocturnal jet. This jet is capable of carrying O_3, other reaction products, and emissions injected aloft considerable distances downwind. This phenomenon is potentially significant in modeling regional-scale air quality and is implicitly treated by the model, where the definition of layer 1 attempts to account for it.

Because standard weather observations do not have the spatial or temporal resolution necessary to determine with confidence the wind fields in layer 1, a submodel within the ROM system was developed to simulate the nighttime flow regime in layer 1 only. This prognostic flow submodel is activated only when a surface inversion is present over most of the model domain. At all other times the flow in layer 1 is determined from interpolation of observed winds. The nighttime flow regime within layer 1 is influenced by buoyancy, terrain, warm cities, pressure-gradient forcing, and frictional forces, all of which are accounted for in the model's flow formulation. Solution of the wind submodel equations produces estimates of the wind components as well as the depth of the inversion layer for all grid cells in layer 1.

2.2.2 ROM Chemistry

The chemical kinetic mechanism embedded in the current version of the ROM is the Carbon Bond IV (CB-IV) set of reactions (Gery *et al.*, 1989). This mechanism simulates the significant reaction pathways responsible for gas-phase production and destruction of the constituents of photochemical smog on regional scales. The mechanism consists of 82 reactions encompassing 33 individual species. The ROM's chemical solution scheme makes no *a priori* assumptions concerning local steady states. Therefore, all species are advected, diffused, and chemically reacted in the model simulations.

The CB-IV contains a standard set of reactions for atmospheric inorganic chemical species, including O_3, NO, NO_2, CO, and other intermediate and radical species. Organic chemistry is partitioned along reactivity lines based on the carbon structures of the organic molecules. Nine individual categories of organics are represented to account for the chemistry of the hundreds of organic molecules existing in the ambient atmosphere: ETH, an explicit representation of ethene; FORM, an explicit representation of formaldehyde; OLE, a double-bonded lumped structure including two carbons (e.g., olefins); PAR, a single-bond, single-carbon structure (i.e., paraffins); ALD2, the oxygenated two-carbon structure of the higher aldehydes; TOL, the aromatic structure of molecules with only one functional group (e.g., toluene); XYL, the structure of molecules with multifunctional aromatic rings (e.g., xylene); ISOP, the five-carbon isoprene molecule; and NONR, a single-carbon organic structure not significantly participating in the reaction sequence. MTHL, methanol, is also included in the mechanism; it is used only in future-year scenarios, when we expect that methanol-powered automobiles will be common.

Three classes of biogenic hydrocarbons are included in a separate natural area source emissions inventory used by the ROM: (1) isoprene, a molecule principally emitted by deciduous trees, is treated by the ISOP species in CB-IV; (2) monoterpenes, a class of natural hydrocarbons emitted principally by coniferous trees, is not treated explicitly in CB-IV. The surrogate monoterpene molecule, α-pinene, which consists of 10 carbons, is apportioned to the existing CB-IV categories as 0.5 OLE, 1.5 ALD2, and 6 PAR; (3) unidentified hydrocarbons (gas chromatography analysis did not identify specific hydrocarbon compounds) are tentatively treated as 50% terpenes, 45% PAR, and 5% NONR (Pierce *et al.*, 1990a). These unidentified compounds can comprise as much as 40% of the biogenic hydrocarbons.

2.2.3 System Components

The components of the ROM are shown in Figure 2.2. The raw input data to the ROM (refer to Section 2.3) are manipulated by the preprocessors and a hierarchical network of 35 processors that range in function from simple reformatting of emissions data to generating the complex wind fields that drive the atmospheric transport. These processors are interconnected by their requirements for and production of data. The ultimate product of the processor network is a collection of data files that can be categorized into two types of *standard* input files: processor files (PF) and model files (MF). PF's contain partially processed data required as input to higher level processors. MF's contain the parameter fields that are transformed into the variables required by the model algorithms; however, they also provide input to a number of higher level processors. The processor network ultimately produces several large data files for the core model; these contain initial-condition and boundary-condition concentrations as well as the data used to model all physical and chemical processes affecting species concentrations in a given episode. These files combined contain tens of millions of Core Model input values; simulating one day, for example, requires nearly 100 billion computations with these millions of values.

The processors are organized into nine distinct hierarchical stages, numbered 0 - 8. Stage 0 processors produce output files such as the gridded land use data. Stage 1 processors interface directly with the preprocessed input data sets which have undergone extensive quality control. Subsequent stages transform the input data into the gridded fields of temporally and spatially varying parameter values needed by the highest stages of the processing network. Processors at any stage can interface directly with the B-Matrix compiler, described below, by production of model input files. This multistage organization is important to the network because it clearly delineates the sequence of program execution. Processors at the same stage may execute simultaneously. A processor at any given stage,

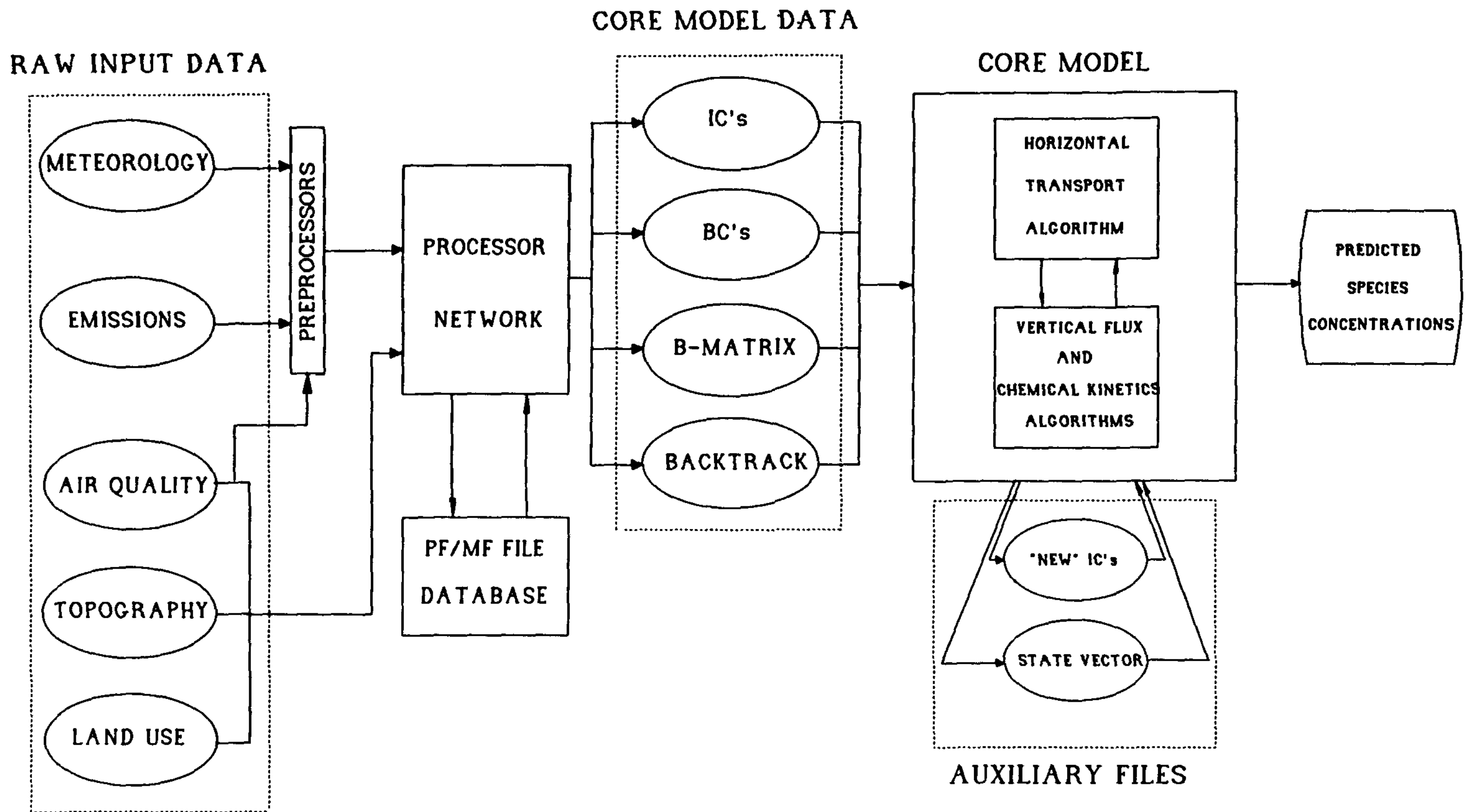

Figure 2.2. Components of the ROM.

however, must wait until all processors from lower stages along its input data paths have been completed. Formal definition of all data/processor relationships and automation of processor executions are essential to ensure consistency and validity of MF's.

Figure 2.3 shows the ROM2.1 processor network. It consists of three interrelated parts: the initial-condition and boundary-condition (IC/BC) processors; the meteorology processors, which process topography and land use data in addition to meteorology data; and the emissions processors. The network transforms the raw data input files into the four core model input files shown on the right: ICON (initial-condition concentration data), BCON (boundary-condition concentration data), BTRK [diffusivity and backtrack (advection transport) information], and BMAT (parameterization for vertical fluxes, meteorological parameters necessary for chemistry rate constant adjustments, and parameterized emissions source strengths). Table 2.1 summarizes the functions of all the processors in the ROM 2.1 network. Table 2.2 reorganizes Table 2.1 to summarize the role of particular processors within functional categories.

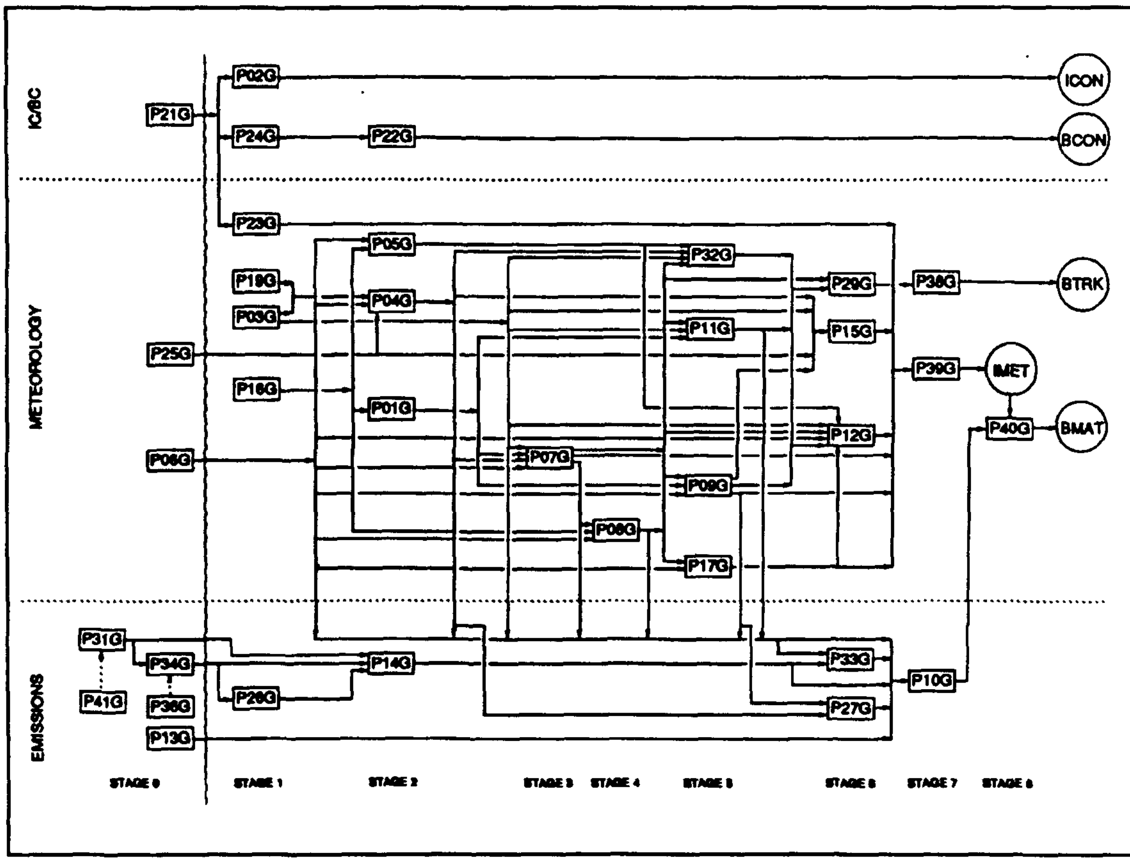

Figure 2.3. Structure and final output files of the ROM 2.1 processor network.

TABLE 2.1. FUNCTIONAL DESCRIPTIONS OF THE ROM2.1 INPUT PROCESSORS

Processor	Stage	Function in ROM2.1
P01G	2	Interpolates profiles of upper-air meteorological parameters at intervals of 50 m from hourly rawinsonde vertical profiles
P02G	1	Writes to the file ICON the gridded initial-condition concentrations for each layer and species simulated by the core model, using P21G's clean-air concentrations as initial-condition concentrations
P03G	1	Prepares surface meteorology data (e.g., mixing ratio, virtual temperature, and wind speed) for use in higher-stage processors
P04G	2	Computes gridded surface roughness, and hourly gridded Monin-Obukhov length, surface heat flux, friction velocity, surface temperature, surface relative humidity, and surface wind speed
P05G	2	Uses surface observations to compute hourly gridded values for the fraction of sky covered by cumulus clouds, and also calculates cumulus cloud-top heights
P06G	0	Computes the smoothed terrain elevation for each 10′ lat. by 15′ long. ROM domain grid cell, and also for a larger domain that extends three grid cells beyond the ROM domain. In addition, it computes average terrain elevations in a finer-resolution domain (cells 5′ lat. by 5′ long.) for the terrain penetration calculation. Finally, it computes the north-south and east-west components of the terrain elevation gradient (slope)
P07G	3	Computes hourly gridded wind fields in the cold layer, hourly gridded terrain penetration fractions, hourly gridded cold layer growth rates, and hourly gridded thicknesses for layers 0 and 1
P08G	4	Computes hourly gridded cell thicknesses for layers 2 and 3, and various parameters used to specify volume fluxes between these two layers
P09G	5	Computes hourly gridded atmospheric density, temperature, cloud cover, solar zenith angle, and water vapor concentration
P10G	7	Computes hourly gridded emissions source functions in layers 0, 1, and 2 for combined anthropogenic and biogenic sources, and also computes the plume volume fraction in layer 0
P11G	5	Computes hourly gridded horizontal winds for each layer, using rawinsonde vertical profiles and surface-station wind observations

(continued)

TABLE 2.1. (CONTINUED)

Processor	Stage	Function in ROM2.1
P12G	6	Computes hourly gridded volume fluxes through all model layer boundaries, and cumulus cloud vertical flux parameters
P13G	0	Computes the total length of highway line-emissions sources within each grid cell
P14G	2	Prepares files containing hourly emissions values and stack descriptions for all major point sources, and combined hourly gridded emissions values for minor point sources, area sources, and mobile sources
P15G	6	Computes hourly gridded effective deposition velocities for a set of representative species
P16G	1	Interpolates between rawinsonde observations to produce hourly upper-air profiles at 25-mb resolution
P17G	5	Computes hourly gridded elevations (above MSL) for the tops of layers 0, 1, 2, and 3, and local time derivatives of these elevations
P19G	1	Computes hourly gridded values of fractional sky coverage at the terrain surface for all cloud types combined
P21G	0	Computes daytime and nighttime tropospheric background (clean-air) concentrations in each layer for each chemical species
P22G	2	Computes and writes to the file BCON the gridded boundary-condition concentrations for each species, layer, and advection time step simulated by the core model, for the north, south, east, and west boundaries
P23G	1	Computes hourly gridded upper-boundary-condition concentrations (C-infinity) for a set of representative species
P24G	1	Equilibrates background concentrations of all modeled chemical species with averaged observed ozone concentrations on the north, south, east, and west boundaries, for both daytime and nighttime conditions in each layer
P25G	0	Computes the fraction of each grid cell in each land use category recognized by the model

(continued)

TABLE 2.1. (CONCLUDED)

Processor	Stage	Function in ROM2.1
P26G	1	Computes hourly gridded mobile-source VOC, NO_x, and CO emissions parameters, adjusted for daily average temperature
P27G	6	Prepares hourly gridded biogenic emission rates for isoprene, paraffin, olefin, high molecular weight aldehydes (RCHO, R > H), nonreactive hydrocarbons, NO, and NO_2.
P29G	6	Computes hourly gridded 30-min backtrack (advection) cell locations and horizontal diffusivities for each layer simulated by the core model
P31G	0	Allocates annual point-source emissions data between a weekday-emissions file, a Saturday-emissions file, and a Sunday-emissions file
P32G	5	Calculates hourly gridded horizontal eddy diffusivities for layers 1, 2, and 3, and also produces parameter fields needed to compute interfacial volume fluxes across layer boundaries
P33G	6	Generates hourly gridded locations and strengths of constant-source emitters for a tracer emissions species
P34G	0	Converts all point-, area-, and mobile-source data files from GMT to LST
P36G	0	Applies NO_x and VOC emission controls at the county level for area- and mobile-source emissions data
P38G	7	Reads the backtrack and diffusivity hourly gridded MF files and computes the BTRK file parameters for each advection time step simulated by the core model
P39G	7	Reads all meteorology hourly gridded MF files *except* the backtrack and diffusivity files read by P38G and computes the intermediate meteorology (IMET) file parameters for each advection time step simulated by the core model
P40G	8	Reads the intermediate meteorology (IMET) file and the emissions sources hourly gridded MF files and computes the BMAT parameters for each advection time step simulated by the core model
P41G	0	Applies NO_x and VOC emission controls to point-source emissions data, at a state, county, point, or individual-boiler level

TABLE 2.2. ROM2.1 INPUT PROCESSORS BY FUNCTIONAL CATEGORY

Stage	Meteorology processors
1	Prepares surface meteorology data (e.g., mixing ratio, virtual temperature, and wind speed) for use in higher-stage processors; processor P03G
1	Interpolates between rawinsonde observations to produce hourly upper-air profiles at 25-mb resolution; processor P16G
1	Computes hourly gridded values of fractional sky coverage at the terrain surface for all cloud types combined; processor P19G
2	Interpolates profiles of upper-air meteorological parameters at intervals of 50 m from hourly rawinsonde vertical profiles; processor P01G
2	Computes gridded surface roughness, and hourly gridded Monin-Obukhov length, surface heat flux, friction velocity, surface temperature, surface relative humidity, and surface wind speed; processor P04G
2	Uses surface observations to compute hourly gridded values for the fraction of sky covered by cumulus clouds, and also calculates cumulus cloud-top heights; processor P05G
3	Computes hourly gridded wind fields in the cold layer, hourly gridded terrain penetration fractions, hourly gridded cold layer growth rates, and hourly gridded thicknesses for layers 0 and 1; processor P07G
4	Computes hourly gridded cell thicknesses for layers 2 and 3, and various parameters used to specify volume fluxes between these two layers; processor P08G
5	Computes hourly gridded atmospheric density, temperature, cloud cover, solar zenith angle, and water vapor concentration; processor P09G
5	Computes hourly gridded horizontal winds for each layer, using rawinsonde vertical profiles and surface-station wind observations; processor P11G
5	Computes hourly gridded elevations (above MSL) for the tops of layers 0, 1, 2, and 3, and local time derivatives of these elevations; processor P17G
5	Calculates hourly gridded horizontal eddy diffusivities for layers 1, 2, and 3, and also produces parameter fields needed to compute interfacial volume fluxes across layer boundaries; processor P32G
6	Computes hourly gridded volume fluxes through all model layer boundaries, and cumulus cloud vertical flux parameters; processor P12G
6	Computes hourly gridded effective deposition velocities for a set of representative species; processor P15G
6	Computes hourly gridded 30-min backtrack (advection) cell locations and horizontal diffusivities for each layer simulated by the core model; processor P29G
7	Reads the backtrack and diffusivity hourly gridded MF files and computes the BTRK file parameters for each advection time step simulated by the core model; processor P38G
7	Reads all meteorology hourly gridded MF files *except* the backtrack and diffusivity files read by P38G and computes the intermediate meteorology (IMET) file parameters for each advection time step simulated by the core model; processor P39G
8	Reads the intermediate meteorology (IMET) file and the emissions sources hourly gridded MF files and computes the BMAT parameters for each advection time step simulated by the core model; processor P40G

(continued)

TABLE 2.2 (CONTINUED)

Stage	Emissions processors
0	Computes the total length of highway line-emissions sources within each grid cell; processor P13G
0	Allocates annual point-source emissions data between a weekday-emissions file, a Saturday-emissions file, and a Sunday-emissions file; processor P31G
0	Converts all point-, area-, and mobile-source data files from GMT to LST; processor P34G
0	Applies NO_x and VOC emission controls at the county level for area- and mobile-source emissions data; processor P36G
0	Applies NO_x and VOC emission controls to point-source emissions data, at a state, county, point, or individual-boiler level; processor P41G
1	Computes hourly gridded mobile-source VOC, NO_x, and CO emissions parameters, adjusted for daily average temperature; processor P26G
2	Prepares files containing hourly emissions values and stack descriptions for all major point sources, and combined hourly gridded emissions values for minor point sources, area sources, and mobile sources; processor P14G
6	Prepares hourly gridded biogenic emission rates for isoprene, paraffin, olefin, high molecular weight aldehydes (RCHO, R > H), nonreactive hydrocarbons, NO, and NO_2; processor P27G
6	Generates hourly gridded locations and strengths of constant-source emitters for a tracer emissions species; processor P33G
7	Computes hourly gridded emissions source functions in layers 0, 1, and 2 for combined anthropogenic and biogenic sources, and also computes the plume volume fraction in layer 0; processor P10G

(continued)

TABLE 2.2 (CONCLUDED)

Stage	Initial/boundary conditions processors
0	Computes daytime and nighttime tropospheric background (clean-air) concentrations in each layer for each chemical species; processor P21G
1	Writes to the file ICON the gridded initial-condition concentrations for each layer and species simulated by the core model, using P21G's clean-air concentrations as initial-condition concentrations; processor P02G
1	Computes hourly gridded upper-boundary-condition concentrations (C-infinity) for a set of representative species; processor P23G
1	Equilibrates background concentrations of all modeled chemical species with averaged observed ozone concentrations on the north, south, east, and west boundaries, for both daytime and nighttime conditions in each layer; processor P24G
2	Computes and writes to the file BCON the gridded boundary-condition concentrations for each species, layer, and advection time step simulated by the core model, for the north, south, east, and west boundaries; processor P22G
Stage	**Land use processor**
0	Computes the fraction of each grid cell in each land use category recognized by the model; processor P25G
Stage	**Terrain processor**
0	Computes the smoothed terrain elevation for each 10′ latitude by 15′ longitude ROM domain grid cell, and also for a larger domain that extends three grid cells beyond the ROM domain. In addition, it computes average terrain elevations in a finer-resolution domain (cells 5′ latitude by 5′ longitude) for the terrain penetration calculation. Finally, it computes the north-south and east-west components of the terrain elevation gradient (slope); processor P06G

The program that serves as the interface between the model input files and the algorithms describing the governing processes in the Core Model is called the B-Matrix Compiler (BMC) because it functions similarly to a computer language compiler that transforms high-level language commands into a machine or algorithm-specific representation. The BMC mathematically combines physical parameters such as layer thicknesses, air densities, etc., into the complex coefficients required for solution of the governing equations. These coefficients can no longer be equated with physical quantities; they are purely mathematical entities related specifically to the form of the finite difference algorithms used by the ROM.

The Core Model of the ROM system consists of a set of algorithms that solves the coupled set of finite difference equations describing the governing processes in each layer of the model. These governing equations are expressed in a form that allows the chemical kinetics, advection, and vertical flux to be treated independently. The chemistry module exchanges information with algorithms of the governing equations via two vectors: (1) a vector that contains the net production rate of each species, and (2) a vector that contains the net destruction rate. Such design simplifications enhance the flexibility of the model and are not limited to the interchanges of the chemical mechanism; they apply to all theoretical formulations of the physical and meteorological processes (i.e., to all the processors).

Currently, we are running the ROM Core Model on the EPA's IBM® 3090 computer, and the ROM processor network on the EPA's VAX™ cluster.[2] Running the ROM requires significant CPU resources; a typical three-day simulation (72 hours, 144 model time steps) requires about 9.5 hours of CPU on the IBM 3090 plus about 12 hours on the VAX cluster.

2.2.4 ROM Limitations

There are several limitations inherent in the model. Among the most important of these are: (1) the model is designed to represent only summer meteorological conditions; (2) the ROM does not take into account any aqueous-phase chemistry; (3) cumulus cloud processes are such that when a cloud is created in a grid cell, it remains there for a full hour (i.e. cloud physics are not considered), and the cloud is not advected; (4) the ROM actually consists of three two-dimensional models that are linked,[3] and, as such, cannot be expected to model regions that contain high, complex, mountainous terrain such as the Rocky Mountains or Sierra Nevada; (5) the ROM, with its current 18.5 km × 18.5 km grid resolution, is not designed to provide detailed information at local scales that are significantly influenced by local source distributions; and (8) the ROM is currently configured to run only on domains in eastern North America on the scale of 1000 km. This regional scale is germane since long-range transport of ozone precursors will have a significant effect on local ozone concentrations.

We advise users of ROM to interpret its results in terms of analysis of different emission control strategies on ozone concentration, rather than assuming the results to be an accurate snapshot of a specific pollution event.

2. IBM is a registered trademark of the International Business Machines Corporation. VAX is a trademark of the Digital Equipment Corporation.
3. The three models represent layer 3, layer 2, and a combined layer 1 and 0.

2.3 INPUT REQUIREMENTS

The ROM system requires five types of "raw" data inputs: air quality, meteorology, emissions, land use, and topography.

Air quality data consist of hourly ozone observations obtained from the U.S. Environmental Protection Agency's National Air Data Branch. These hourly observations are used to specify the initial and upwind-boundary ozone concentrations required by the ROM. Initial conditions are derived from the mean tropospheric background concentrations listed in Table 2.3 (Killus and Whitten, 1984), and computed using the temperature-dependent rate constants calculated by the ROM system. Background concentrations are used since ROM simulations begin on relatively-low ozone days, several days prior to observations of ozone levels > 120 ppb. The model is initialized with clean tropospheric conditions for all species two to four days before the start of the period of interest (an ozone *episode* of typically about 15 days in duration). Thus, the initial condition field has effectively been transported out of the model domain in advance of the portion of the episode of greatest interest (i.e., high ozone days).

TABLE 2.3. 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}

The twice-daily (daytime and nighttime) gridded equilibrated concentrations for the 35 species used in the ROM, for the north, south, east, and west boundaries of each model layer are derived as follows: each boundary is assigned a single value for ozone, based on average ambient measurements at rural monitoring sites. The remaining 34 species are then equilibrated to this ozone concentration, generating the set of concentration values for that boundary.

Meteorology input data consist of the regular hourly surface and upper-air observations from U.S. National Weather Service and Environment Canada. Surface-weather station reports include wind speed and direction, air temperature and dew point temperature, atmospheric pressure, and cloud amounts and heights. Twice-daily sounding data from the upper-air observation network are included in the meteorology database, and consist of atmospheric pressure, wind speed and direction, and air temperature and dew point temperature. Additional meteorology data are obtained from the National Climatic Data

Center and consist of buoy and Coastal Marine Automated Station data; parameters typically reported are wind speed and direction, and air and sea temperatures. Meteorology data are used to specify the meteorological parameters in ROM simulations.

Emissions input data consist of annual point-source emissions, with stack parameters and seasonal, day-type factors; area-source emissions for typical summertime Saturdays, Sundays, and a "generic" weekday; and mobile-source emissions. Most recently, anthropogenic emissions data have been provided from the National Acid Precipitation Program (NAPAP) 1985 emissions inventory with 20-km spatial resolution. Species included are CO, NO, NO_2, and nine hydrocarbon reactivity categories. Biogenic emissions data are also input, including isoprene explicitly, monoterpenes divided among the existing reactivity classes, and unidentified hydrocarbons (refer to Section 2.2.2).

Land use input data consist of 11 land use categories in $^1/_4$-degree longitude by $^1/_6$-degree latitude grid cells. Data are provided for the U.S. and Canada as far as 55 °N. The land use categories are (1) urban land, (2) agricultural land, (3) range land, (4) deciduous forests, (5) coniferous forests, (6) mixed forest with wetlands, (7) water, (8) barren land, (9) nonforested wetland, (10) mixed agricultural land and range land, and (11) rocky, open places occupied by low shrubs and lichens. The data were obtained from the EPA laboratory in Las Vegas, NV. Land use data are used to obtain biogenic emissions estimates as a function of the area of vegetative land cover, and for the determination of surface heat fluxes.

Topography input data consist of altitude matrices of elevations for 30"×30" cells in a $7^1/_2 \times 7^1/_2$ grid. The data are obtained from the GRIDS database operated by the U.S. Environmental Protection Agency's Office of Information Resources Management. Topography data are used in the calculation of layer heights.

2.4 SUITABILITY OF THE ROM FOR ASSESSMENT ACTIVITIES

The ROM was designed to predict regional patterns of maximum O_3 concentrations over multi-day periods. The ROM has performed adequately in this regard for the northeastern United States. Evaluation efforts have been included in the ROM development and testing since the initial phases of the program. The ROM program has undergone formal EPA scientific peer review approximately every three years since the inception of the program nearly ten years ago. The original developers of the model performed rigorous tests on its components, including the numerical solution procedures that were used to solve horizontal and vertical transport and the set of chemical reactions. The chemical kinetic mechanism was tested against hundreds of smog chamber data sets. The transport algorithms were subjected to tests using analytically derived flow fields. Tests of the full model with ambient O_3 data began with the first generation ROM in which an episode from August 1979 in the northeastern United States was modeled. Results from the series of evaluation studies indicate (1) that the ROM predicts ozone sufficiently well to be accepted by the EPA in assessing changes in maximum hourly ozone for alternative emission reduction strategies; and (2) that the model can be recommended for use in regulatory analyses, with some caveats. These caveats include the following:

- model biases vary from day to day and from area to area within the modeling domain, based on comparisons with observations;
- model performance degrades during some situations that have dynamic mesoscale wind flow conditions; and
- during episodic conditions, the model generally underpredicts regional background (remote rural) O_3 concentrations.

2.5 PERFORMANCE EVALUATIONS

The Regional Oxidant Model was designed for regional analyses of alternative strategies that differentially control emissions of NO_x and hydrocarbons. The primary objective in applications of the ROM is to assess the effectiveness of these regional emission control strategies in reducing maximum hourly O_3 concentrations. The level of the National Ambient Air Quality Standard (NAAQS) for O_3 is 0.12 ppm (120 ppb) for 1 hour, and has been set at a level designed to protect human respiratory systems from acute O_3 damage. The ROM therefore operates on a 1-hour time scale, but can also predict 7- or 8-hour average O_3 concentrations, the time scale of exposure most indicative of damage to crops and trees. These averaged O_3 concentrations can be studied to determine the impact of the various emissions controls on terrestrial systems. States containing the most severe ozone nonattainment areas are required to provide a computer-modeled demonstration of attainment of the ozone NAAQS. The demonstration must include the anticipated change in ozone concentrations in future years, as well as the impact of proposed emission controls.

Emission control strategies for mitigating acidic deposition may also involve the reduction of NO_x from major industrial sources. Thus, emission control strategies consider the effects of all currently-proposed legislation that affects ozone precursor emissions.

The ROM is being used in a separate EPA program, the Regional Ozone Modeling for Northeast Transport (ROMNET) program, to assess the effectiveness of various emission control strategies in lowering O_3 concentrations to nationally-mandated levels for the protection of human health, forests, and crops. As part of the ROMNET program, the ROM is also being used to provide regionally-consistent initial and upwind boundary conditions to smaller-scale urban models for simulations of future years.

Thus far, all model simulations have been performed in one or more of three domains. Two domains are: the northeastern United States (NEROS) and the southeastern United States (SEROS); both are the same number of grid cells, 60 cells in the east-west direction and 42 cells in the north-south direction, for a total domain size of approximately 1100 km by 780 km. The ROMNET domain consists of 64 cells in the east-west direction, and 52 cells in the north-south direction, for a total domain size of approximately 1200 km by 960 km. The three domains are shown in Figure 2.4.

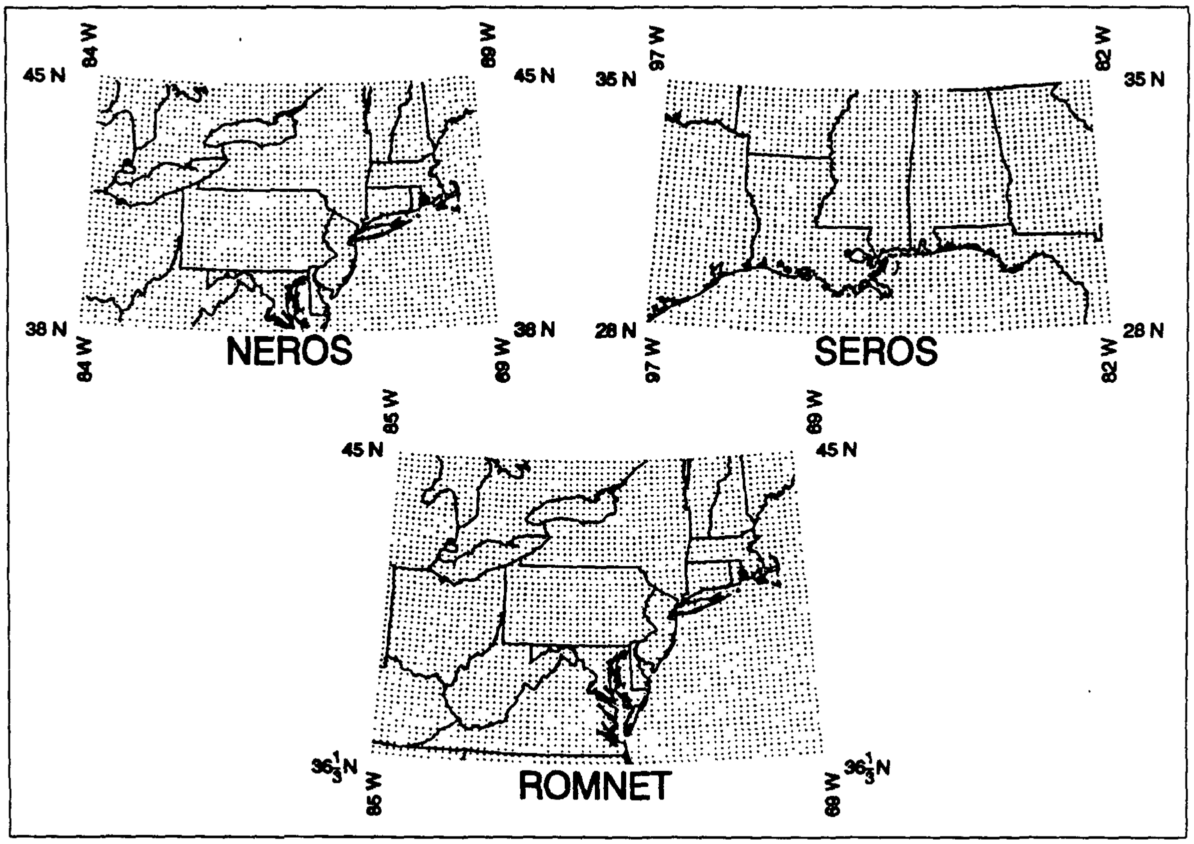

Figure 2.4. NEROS, SEROS, and ROMNET modeling domains for the ROM; dots represent grid cell corners.

The most complete testing of ROM2.0 was accomplished in an evaluation with the 50-day (July 12 to August 31, 1980) NEROS database (Schere and Wayland, 1989). Tests against ambient data were relatively stressful, given that complex flow situations were known to exist in the northeastern United States during the model application period. ROM2.0 underestimated the highest values and overestimated the lowest. The model showed good performance (an overall 2% overprediction) in predicting maximum daily O_3 concentrations averaged over aggregate groups of monitoring stations. A key indicator of model performance on the regional scale is the accuracy of simulating the spatial extent and location, as well as the magnitude, of the pollutant concentrations within plumes from significant source areas. In ROM2.0 performance analyses, plumes from the major metropolitan areas of the Northeast Corridor, including Washington DC, Baltimore, New York, and Boston, could be clearly discerned in the model predictions under episodic conditions. Generally the plumes were well characterized by the model, although there was evidence of a westerly transport bias and underprediction of O_3 concentrations near the center of the plume. Using aircraft data, ROM2.0 was found to underpredict the regional tropospheric burden of ozone.

The evaluation of ROM2.1 (Pierce *et al.*, 1990b), unlike that of ROM2.0, was based on routinely-archived data from state and local agency monitoring sites rather than on an intensive field-study period. The evaluation consisted of comparisons of observed and modeled O_3 concentrations during selected episodes (totalling 26 days) of high ozone observed during the summer of 1985. ROM2.1 underestimated the highest values and slightly overestimated the lowest; underestimates of the upper percentiles tended to be more prevalent in the southern and western areas of the ROMNET domain. The model showed good performance (an overall 1.4% overprediction) in predicting maximum daily O_3 concentrations averaged over aggregate groups of monitoring stations. ROM2.1 appears to correct for the westerly transport bias of high-ozone plumes in the Northeast Corridor seen in ROM2.0. As with ROM2.0, model performance degraded as a function of increasingly-complex mesoscale wind fields.

The ROM was tested for its ability to predict changes in O_3 concentrations effected by changes in precursor emissions. We assume that if the model performs well in an absolute sense, it should also perform well in a relative sense (i.e., under conditions of reduced emissions). In general, the results provided by the ROM2.0 control strategy simulations have shown that controlling VOC emissions alone reduces peak O_3 concentrations in and downwind of urban areas. Controlling NO_x alone can have mixed effects on O_3 concentrations. Increases in maximum O_3 concentrations of up to 20% were observed close to some urban areas, while localized decreases in maximum O_3 concentrations, from 5% to 10%, were observed in rural areas. Preliminary results from ROM simulations employing combined VOC and NO_x emissions controls indicate that these controls are more effective than VOC controls alone in some remote rural areas, although the effects can vary with meteorological conditions.

This page is intentionally left blank.

SECTION 3

METEOROLOGY DATA PREPROCESSING

Sections 3.1, 3.2, and 3.3 are a guide to the flow of preprocessing programs that take the raw buoy, surface, and upper-air meteorology data, perform quality control on them, and prepare them for input into the meteorology processors. Since human visual inspection of the data is required at various stages of the procedure, we do not recommend the use of a concatenated batch run stream that would take all the raw data and produce finalized data sets. We suggest that each routine be run individually. Descriptions of the programs begin in Section 3.4; these include sample batch run streams that are specific for each program. The programs are presented in alphabetical order.

Summaries of the flow of programs are tabularly listed in Sections 3.1.3 (buoy), 3.2.6 (surface), and 3.3.7 (upper air). A chart showing the flow of programs is shown in Figure 3.1.

All U.S. data sets discussed below can be obtained from the National Climatic Data Center (NCDC) - E/CC42, Attn: User Services Branch, Federal Building, Asheville, NC. 28801. The NCDC telephone number is (704)259-0682. Canadian data can be obtained from the Canadian Climate Centre (CCC), 4905 Dufferin St., Downsview, Ontario M3H 5T4. The telephone number of the CCC Digital Archive is (416)739-4335. All data files referenced in this document (TD-1171, 3280, 6200; and TDF-1440) have format documentation that you can request from the NCDC or the CCC.

3.1 BUOY DATA

Roughly one-fourth of the ROMNET domain consists of the Atlantic Ocean and its bays and inlets, or is part of the Great Lakes system. We use approximately 30 buoys to provide surface meteorological data over water areas in the ROMNET domain.

3.1.1 The Data (1980 to Present)

The National Data Buoy Center (NDBC), part of the NCDC, processes both buoy and Coastal Marine Automated Stations (C-MAN) data. These data are stored in TD-1171 format, with a fixed record length and block size.

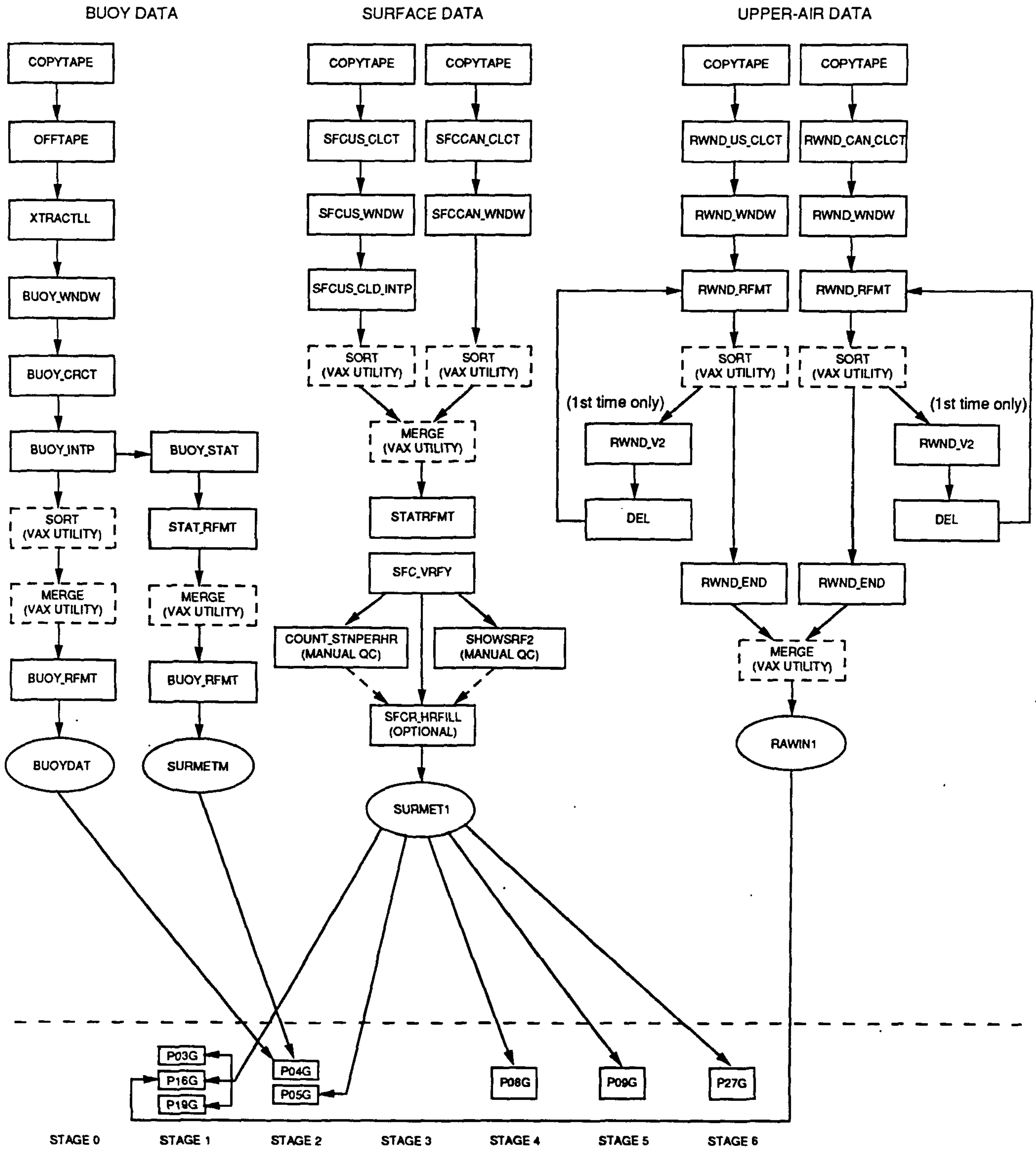

Figure 3.1. Flow of programs for meteorology input data. VAX-specific programs are shown in broken-line boxes; arrows with broken-line shafts imply that the path exists only if errors or omissions are encountered.

3.1.2 Buoy Data Preprocessing

The hourly buoy data are put through a series of programs that manipulate them so that their final format is the same as the standard surface meteorology data file (refer to Section 3.2.1 for the surface meteorology data file list), with three exceptions: (1) Sea surface temperature (SST) replaces the dew point (#16); (2) the difference between SST and air temperature (SST - T) replaces the station pressure (#17); and (3) the cloud data (#4 - #11) are reported as missing. Buoy data are reported in Greenwich Mean Time (GMT).

Data processing proceeds as follows: The data are read from the NCDC tape by OFFTAPE, windowed spatially[1] by XTRACTLL, and then windowed temporally (by months) by BUOY_WNDW. Latitude and longitude coordinates, if missing, are added by BUOY_CRCT. Temporal interpolation and quality control are performed by BUOY_INTP. Statistics are calculated by BUOY_STAT. After the data have been sorted by the VAX system's SORT utility,[2] the final hourly data set is prepared by BUOY_RFMT in the SURMET1 format that can be read by the meteorology processors. An example of this format is shown in the P03G processor documentation, Section (Part 2, Section 3.4).

Since monthly-averaged SST's are sometimes helpful, processing of the statistics produced by BUOY_STAT is conducted. The log file from that program is read by STAT_RFMT to prepare the monthly-averaged data file. The date and time associated with each report is the first hour, reported as GMT, of the month for which the data have been averaged, in a year/month/day/hour field. The monthly-averaged data file is then reformatted to a SURMET1-type file by BUOY_RFMT.

Note that buoy data prior to 1988 are all on one tape, while buoy data for 1988 and later are on two tapes - fixed buoys on one tape and Coastal Marine Automated Stations (C-Man) data on another. When you receive two tapes of data, you should process the data in parallel until after both data sets are sorted. These data sets must then be merged before you run BUOY_RFMT.

The data sets from BUOY_RFMT require human intervention to produce the final data set. Many buoys report air temperature but not sea surface temperature, and thus cannot be used as data points to determine the sea surface temperature field needed by P04G. Since SST and air temperature are highly correlated, we think that the air temperature data can be used to "improve" upon the SST field that would be produced by using only observed SST values. Our general approach involves plotting

1. Currently, XTRACTLL windows the northern Eastern Seaboard and the Great Lakes areas.
2. For more information on the SORT utility, refer to the *VMS Sort/Merge Utility Manual*, Order Number AA-LA09A-TE. Digital Equipment Corporation, Corporate User Publications - Spit Brook, ZK01-3/J35 110 Spit Brook Rd., Nashua, NH 03062.

the buoy SST, T, and (SST - T) data, and then using meteorological experience to estimate the SST for those stations with missing data. The missing SST reports are then manually changed to the estimated values, while the (SST - T) field is left with a "missing" flag.

3.1.3 Summary

Table 3.1 below shows the sequence of programs to produce the final hourly data set for the buoy observations.

TABLE 3.1. PREPROCESSOR PROGRAMS FOR BUOY OBSERVATIONS [a]

ROUTINE (.FOR)	INPUT (.DAT)	OUTPUT (.DAT)	COMMENTS
OFFTAPE	RAW DATA TAPES	NDBCYY	
XTRACTLL	NDBCYY	XTR_NDBC	
BUOY_WNDW	XTR_NDBC	BUOY_MMM_1MMM_2YY_WNDW	
BUOY_CRCT	BUOY_MMM_1MMM_2YY_WNDW	BUOY_MMM_1MMM_2YY_CRCT	
BUOY_INTP	BUOY_MMM_1MMM_2YY_CRCT	BUOY_MMM_1MMM_2YY_INTP	
SORT	BUOY_MMM_1MMM_2YY_INTP	BUOY_MMM_1MMM_2YY_SORT	sorted
BUOY_RFMT	BUOY_MMM_1MMM_2YY_SORT	BUOY_MMM_1MMM_2YY	reformatted
BUOY_STAT[b]	BUOY_MMM_1MMM_2YY_INTP	BUOY_STAT.LOG	
STAT_RFMT[b]	BUOY_STAT.LOG	STAT_MMM_1MMM_2YY	
BUOY_RFMT[b]	STAT_MMM_1MMM_2YY	BUOY_MMM_1AVG	
manually adjusted[b]	BUOY_MMM_1AVG	BUOY_MMM_1AVG	final averaged data set

[a] MMM_1 is the beginning month of the data, MMM_2 the ending month.

[b] These final four programs are all that are required for the monthly-averaged data.

3.2 SURFACE METEOROLOGY DATA

Both Canadian and U.S. surface meteorology data are required for the ROM. The data are reported hourly during the hours of operation of airports,[3] and are specified in Local Standard Time (LST). These data arrive on tapes in different formats, thus requiring separate processing in the early stages. The major disparities between the two data sets involve the station identification and cloud data. The U.S. data set uses WBAN numbers as station identifiers, while the Canadians use their own identification system. The U.S. data set reports layered cloud information every three hours only, while the Canadian data generally report this information every hour. *Depending on the record length and block size of the data on tape, the surface data preprocessing programs may require slight modifications (i.e., changing the RECL and BLKSZE parameters).*

3. Some smaller airports close during the night.

The sequence of programs that process raw surface meteorological data involves initial parallel manipulation of U.S. and Canadian data. The programs that process the surface data for both the United States and Canada, merge them, and perform quality control on the merged set, are described below, beginning with Section 3.2.2.

3.2.1 Parameter Descriptions for Surface Data

All U.S. and Canadian stations in the final data set contain the 17 surface data parameters below:

1 - Station WBAN number.[4]

2 - Station latitude (° N).

3 - Station longitude (° W).

4 - Opaque sky cover in percent; "-99." if indeterminate.

5 - Total sky cover in coded form:

0	=	Clear	5	=	Scattered
1	=	Partial obscuration	6	=	Broken
2	=	Thin scattered	7	=	Overcast
3	=	Thin broken	8	=	Obscured
4	=	Thin overcast	-99	=	Indeterminate.

6 - Amount, in percent, of lowest cloud layer. A missing report is coded as "-99.". A missing value indicates that either the report was incomplete, or that this level could not be seen due to obscuring phenomena at the surface. If more than three cloud layers exist, this layer is the lowest of the three most significant cloud layers.

7 - Cloud base of lowest layer in meters. A value of "-8888." means that clear conditions existed at this level and at all levels above it. A missing report is coded as "-9999.".

8 - Amount, in percent, of second-lowest cloud layer. A missing report is coded as "-99.". If reports for lower levels exist, a missing report here implies an overcast at a lower level, and the remaining columns for cloud reports should also be "-99.".

9 - Same as 7, but for the second-lowest level.

10 - Amount, in percent, of the third-lowest layer. A missing report is coded as "-99.".

11 - Same as 7, but for the third-lowest layer.

12 - Sea level pressure in millibars (mb), -9999.0 = missing.[5]

13 - Wind direction in degrees, -9999.0 = missing.

14 - Wind speed in $m \cdot s^{-1}$, -9999.0 = missing.

15 - Temperature in degrees Celsius, -9999.0 = missing.

16 - Dew point in degrees Celsius, -9999.0 = missing.

17 - Station pressure in millibars, -9999.0 = missing.

4. Canadian data have either (1) a converted WBAN number, or (2) a pseudo-WBAN number that we created.

5. 1000 mb = 10^5 Pa = 10^5 $N \cdot m^{-2}$

3.2.2 United States Data Preprocessing

The U.S. data are in TD-3280 format, and therefore have variable block sizes and record lengths. A year's worth of data typically arrives on four tapes, with the data organized such that an entire year's data for an individual station is followed by a year's data for the next station, and so on. Thus, it is a CPU-intensive procedure to extract data for synoptic purposes. The desire to conserve CPU time must be balanced with the need to keep the output data sets a reasonable length.

- The first program of the sequence allows you to choose which months of data are to be extracted. This program, SFCUS_CLCT.FOR, needs to be run for each of the raw tapes. The output files are called SFCUS_*MMM_1MMM_2YY**_CLCT.DAT, where MMM_1 is the beginning month of the data, MMM_2 the ending month, *YY* the year; the * is replaced by a character that indicates from which tape[6] the data were extracted.
- The next program to run is SFCUS_WNDW.FOR; it reads the output files SFCUS_*MMM_1MMM_2YY**_CLCT.DAT and performs several operations on the data. With the help of the look-up file STNS_US_DEGFRAC.LIS, station latitude and longitude are added to the data, and the file is windowed for a particular region with boundaries that you declare. The times of the meteorological observations are converted from LST to GMT, and a particular month of data is windowed, again as requested by you. Regardless of how many files are read, only one file, SFCUS_*MMMYY*_WNDW.DAT, is output.
- Because the ROM processor P05G needs layered cloud data at every hour, the three-hourly cloud data given in the TD-3280 format is not sufficient. SFCUS_CLD_INTP.FOR is employed to duplicate cloud layer reports on both sides of a standard reporting hour, if no layered cloud information otherwise exists. SFCUS_*MMMYY*_INTP.DAT is the name of the output file.
- The VAX system's SORT utility is then applied to the data to produce a synoptic data set, called SFCUS_*MMMYY*_SORT.DAT, which awaits merger with the Canadian surface data.

3.2.3 Canadian Data Preprocessing

The Canadian data are in fixed record (TDF-1440) format.

- SFCCAN_CLCT.FOR[7] creates SFCCAN_*MMM_1MMM_2YY**_CLCT.DAT.
- The Canadian station identifiers are converted to WBAN numbers when these data sets are read by SFCCAN_WNDW.FOR. In addition to this, all the other functions performed by its U.S. counterpart are also performed by this program in producing the single data set SFCCAN_*MMMYY*_WNDW.DAT.
- The VAX system's SORT utility then creates SFCCAN_*MMMYY*_SORT.DAT.

---

6. Usually you will have a choice of one of four surface data tapes, named A, B, C, or D.
7. Parallels the U.S. data's preprocessor SFCUS_CLCT.FOR.

3.2.4 Surface Data Merging

The VAX system's MERGE utility[8] is used to combine the U.S. and Canadian data sets into SFC_*MMMYY*_SORT.DAT.

3.2.5 Surface Data Quality Control

At this point the record header and record format of the data must be modified to the required header and body format through SFC_RFMT.FOR to produce SFC_*MMMYY*_RFMT.DAT.

In this format, the quality assurance program SFC_VRFY.FOR is able to apply the Barnes Objective Analysis routine (Barnes, 1973) to the data to weed out dubious temperature, dew point, pressure, or wind values. The resultant data set, SFC_*MMMYY*_VRFY.DAT, is stripped of its file header information (which is not needed by the ROM processors), and is the final data set.

You should track the output log of SFC_VRFY.FOR to see how many stations are discarded. Normally, you should not expect to see more than five or six stations per hour to be flagged for erroneous data. However, the program probably will discard all stations along a major discontinuity, such as a strong cold front. Since one of the built-in assumptions of the ROM is that the modeling domain consists of a single airmass, elimination of these stations is consistent with this premise. SFC_VRFY.FOR may also discard sub-synoptic scale wind data; you should expect to see, therefore, discarding of winds in areas with thunderstorms.

The resultant data are in a final form, called SFC_*MMMYY*_VRFY.DAT, and can be processed by the ROM, but should be passed through one more quality control step. You begin this step by examining the initial line of each of the data set's first few hours, sampling from both daylight and night periods. These initial lines are parameterized as *YYMMMDDHH* (# of STATIONS). You can see approximately how many stations are reporting for each hour (this number will probably vary over the course of a day). You can then call the routine COUNT_STNPERHR.FOR, which requires that you declare two criteria: your lower- and upper-bound estimates of the number of stations, which you determine from your initial examination of how many stations are reporting for each hour. An appropriate rule of thumb is to assign these bounds at ∓10% of the lowest and highest number of stations observed during your examination. The routine flags hours that have less than the lower-bound number of stations, as well as hours that are entirely missing from the data set; and it flags hours that exceed the upper bound, in case two hours have been concatenated. If hours are flagged,

8. For more information on the MERGE utility, refer to the *VMS Sort/Merge Utility Manual*, Order Number AA-LA09A-TE. Digital Equipment Corporation, Corporate User Publications - Spit Brook, ZK01-3/J35 110 Spit Brook Rd., Nashua, NH 03062.

you must investigate the flagged data to evaluate whether correction is necessary.[9] If you determine that crucial data are missing (i.e. some stations are not reporting for that hour; the flagged hour is entirely missing; or there are too few stations reporting for the modeling region), you can run the routine SFCR_HRFILL.FOR, which copies the existing data from an earlier or later hour to the missing data's location. This routine also requires that you declare a number (of stations) as a lower-bound criterion.[10] Only if the actual number of stations is less than the criterion will missing information be replaced by a prior or following hour's data. For example, if two consecutive hours of data are missing, the first hour will be filled by data from the hour preceding the gap, the second by data from the hour following the gap. If three hours are missing, the first two hours take on the data values of the preceding hour, while the third hour is filled by the following hour. *We do not recommend that you use this routine if the data gap exceeds five or six hours; violating this limit will introduce major inaccuracies into your data set.*

The final surface data quality control check is to plot all U.S., Canadian, and buoy data with the output quality control program SHOWMET (see Section 7.7). This routine displays wind speed and direction, air temperature and dew point, and pressure. You will usually find that the plots of the buoy, U.S., and Canadian data sets are coherent; on occasion, the land-sea or U.S.-Canada boundaries will show data discontinuities. If the plot does not appear to be coherent, you have cause to question the input data. There may be an obvious time shift in one set of data reports, for example. Your staff meteorologist can assist you in determining whether the raw data can be corrected.

3.2.6 Summary

Table 3.2 on the next page summarizes the flow of programs that read U.S. and Canadian surface data, perform quality control procedures on them, and produce the final hourly surface meteorology data that are input to the ROM.

9. The routine always flags hour 0; it is not sufficiently sophisticated to recognize the continuity between hour 23 and hour 0.
10. Not necessarily the same lower-bound number as you declared for COUNT_STNPERHR.FOR.

TABLE 3.2. PREPROCESSOR PROGRAMS FOR SURFACE METEOROLOGY OBSERVATIONS [a]

United States Data				Canadian Data			
ROUTINE (.FOR)	INPUT (.DAT)	OUTPUT (.DAT)	COMMENTS	ROUTINE (.FOR)	INPUT (.DAT)	OUTPUT (.DAT)	COMMENTS
SFCUS_CLCT	RAW DATA TAPES	SFCUS_$M_1M_2YY^*$_CLCT		SFCCAN_CLCT	RAW DATA TAPES	SFCCAN_$M_1M_2YY^*$_CLCT	
SFCUS_WNDW	SFCUS_$M_1M_2YY^*$_CLCT	SFCUS_*MMMYY*_WNDW		SFCCAN_WNDW	SFCCAN_$M_1M_2YY^*$_CLCT	SFCCAN_*MMMYY*_WNDW	
SFCUS_CLD_INTP	SFCUS_*MMMYY*_WNDW	SFCUS_*MMMYY*_INTP					N/A
SORT	SFCUS_*MMMYY*_INTP	SFCUS_*MMMYY*_SORT	VAX system sort	SORT	SFCCAN_*MMMYY*_WNDW	SFCCAN_*MMMYY*_SORT	VAX system sort

ROUTINE (.FOR)	INPUT (.DAT)	OUTPUT (.DAT)	COMMENTS
MERGE	SFCUS_*MMMYY*_SORT and SFCCAN_*MMMYY*_SORT	SFC_*MMMYY*_SORT	VAX system merge
SFC_RFMT	SFC_*MMMYY*_SORT	SFC_*MMMYY*_RFMT	reformatted
SFC_VRFY	SFC_*MMMYY*_RFMT	SFC_*MMMYY*_VRFY	final data set
COUNT_STNPERHR	SFC_*MMMYY*_VRFY	SFC_*MMMYY*_VRFY	recommended QC [b]
SFCR_HRFILL	SFC_*MMMYY*_VRFY	SFC_*MMMYY*_VRFY	recommended QC
SHOWMET	SFC_*MMMYY*_VRFY	SFC_*MMMYY*_VRFY	recommended QC

a Because of page space limitations, we abbreviate MMM_1 as M_1, and MMM_2 as M_2.

b After you complete all quality controls, change the final data file name to SRFC_*MMMYY*.DAT, the SURMET1 naming convention.

3.3 UPPER-AIR DATA PREPROCESSING

Both Canadian and U.S. upper-air data are required for the ROM. The data are reported every 12 hours (at the nominal times of 0000 GMT and at 1200 GMT), and are specified in GMT[11]. These data arrive on tapes in different formats, thus requiring separate processing in the early stages. The U.S. data set uses WBAN numbers as station identifiers, while the Canadians use their own identification system. *Depending on the record length and block size of the data on tape, the upper-air data preprocessing programs may require slight modifications (i.e., changing the RECL and BLKSZE parameters).*

3.3.1 The Data (1983 to Present)

The upper-air data used in ROM processors have two distinct origins. The United States data (including some extraneous sites such as Monterrey, Mexico, the Bahamas, Bermuda, etc.) are originally found on raw data tapes in TD-6200 format, with variable block sizes and record lengths. The Canadian data, on the other hand, arrive with a format of fixed record lengths, in a CCC format designated as "UAS-UAW". Besides the format distinction, the major difference in the two data sets is the wind field information. As a rule, the U.S. data report winds at each level that measurements are taken,[12] while the Canadian data reported winds at mandatory levels[13] only through 1987.[14] Additionally, surface winds are not regularly reported in the Canadian data. Because the two data types are so different, they are treated separately in the initial stages of processing.

The sequence of programs that process raw upper-air meteorological data involves initial parallel manipulation of U.S. and Canadian data. The programs that process the upper-air data for both the U.S. and Canada, merge them, and perform quality control on the merged set, are described below.

3.3.2 Parameter Descriptions for Upper-Air Data

All U.S. and Canadian stations in the final data set contain the 14 upper-air data parameters below:

1 - *YY*
2 - *MMM*
3 - *DD*
4 - *HH* as GMT
5 - Station WBAN number[15]

11. The nominal *upper-air sounding* times of 0000 and 1200 GMT are times of the release of the balloon, with a tolerance of ± 1 hour. If the reported time is not one of these times, then it is the actual time of release.
12. Technically called "significant levels."
13. Surface, 850 mb, 700 mb, 500 mb, 300 mb, 250 mb and 200 mb pressure levels.
14. 1988 Canadian data reported significant-level winds.
15. Canadian data have either (1) a converted WBAN number, or (2) a pseudo-WBAN number that we created.

6 - Current level number and total number of pressure levels reported.

7 - Atmospheric pressure at the current level in mb; -9999.0 = missing.

8 - Height of the current level in whole meters.

9 - Temperature of the level in degrees Celsius and tenths; -9999.0 = missing.

10 - Dew point of the level in degrees Celsius and tenths; -9999.0 = missing.

11 - Relative humidity of the level in whole percent; -9999.0 = missing.

12 - Wind direction in degrees from North; -9999.0 = missing.

13 - Wind speed in $m \cdot s^{-1}$; -9999.0 = missing..

14 - Latitude and longitude of the station in degrees and fractions.

3.3.3 United States Data Preprocessing

Sounding information for a given year is found on at most two data tapes. Each tape is organized by station; thus, an entire year's worth of data is given for each site before a new station is encountered. The variable record and block lengths require reading the tape one record at a time. This means that it take nearly as much CPU time to extract only a month of data as it does to extract the entire year of data. This, in addition to the frequent hardware problems that can occur in reading tapes, led to the decision to extract the entire year's worth of data for the stations in the area of interest.

- The program that collects the needed data is RWND_US_CLCT, which requires you to declare the latitude and longitude boundaries of the modeling domain, as well as the pressure level at which the soundings are to be truncated. Currently the program is set up to produce one file from either one or two input tapes, but will require modification if, at some future time, the NCDC produces the data on more than two tapes. The program's output is a large file named RWND_*YY*US_CLCT.DAT, where *YY* represents the year for which the data were extracted.
- The next program, RWND_WNDW, temporally windows the data by month (since months are convenient delineating periods, and windowing reduces disk storage requirements). It also eliminates data at times other than 00Z, 06Z, 12Z, and 18Z,[16] and includes only those soundings that are within a ±1 hour tolerance of a "standard hour" sounding that reported early or late. RWND_*YY*US*MMM*_WNDW.DAT is the name of the resultant data set, where *MMM* is the month abbreviation.
- At this point the actual sounding data need to be massaged so that they can be used by the ROM processors. This data manipulation is performed by RWND_RFMT. This routine (1) deletes levels that have missing data, (2) ensures that no two levels occupy the same height, (3) deletes soundings with a missing surface temperature or dew point (a very uncommon occurrence), and (4) vertically interpolates missing temperatures, dew points, or winds. A simple linear interpolation is used to fill these missing levels whenever possi-

16. Z stands for Zulu time, equivalent to GMT. Sounding data are reported in Zulu time.

ble, though the situation is more complicated for missing surface winds. In this case a simplified Ekman spiral (but one that is still based on a 45° shear) is assumed and the lowest reporting wind is used to approximate the top of the spiral. The name of the resultant data set is RWND_*YY*US*MMM*_RFMT.DAT.

- This file is then sorted by the VAX system's SORT utility to produce a chronologically ordered data set, RWND_*YY*US*MMM*_SORT.DAT, which awaits merger with the preprocessed Canadian data.

3.3.4 Canadian Data Preprocessing

- The Canadian upper-air data station identification scheme is unique to Canada. To assure compatibility with the U.S. data, the extraction program RWND_CAN_CLCT converts from the Canadian stations' identifiers to WBAN numbers. The program also performs the same functions as does the corresponding U.S. routine, RWND_US_CLCT. The output file is called RWND_*YY*CAN_CLCT.DAT, and is named in parallel to its U.S. equivalent, as are the subsequent files produced by this sequence of programs.
- The same programs that are used to window temporally (RWND_WNDW) and to reformat the U.S. data (RWND_RFMT) are used at this point to produce two data files, named RWND_*YY*CAN*MMM*_WNDW.DAT and RWND_*YY*CAN*MMM*_RFMT.DAT. (In fact, the quirks in the Canadian data are the primary reason for including the sections of code that treat missing or duplicate sounding levels in the programs described earlier for the U.S. data.)
- The VAX system's SORT utility is used to produce RWND_*YY*CAN*MMM*_SORT.DAT.

3.3.5 Upper-Air Data Merging

After the U.S. and Canadian data have been sorted, they are combined by the VAX system's MERGE utility to produce RWND_*YYMMM*.DAT.

3.3.6 Upper-Air Data Quality Control

The quality control program RWND_V2 is then executed on the merged data set. This program tests the sounding data against established sounding criteria, which are:

- Out-of-bounds layer-thickness based superadiabatic lapse rates for layers >200 m AGL:
  $dT/dz < -0.015$ °C/m, $dz > 100$ m ($\simeq 1.5 \times$ the adiabatic lapse rate);
  $dT/dz < -0.022$ °C/m, $50 \text{ m} \leq dz \leq 100$ m;
  $dT/dz < -0.030$ °C/m, $dz \leq 50$ m;
  where T = temperature, z = geopotential height.
- Out-of-bounds positive lapse rates (i.e. a temperature inversion) based on the layer thickness as a function of layer height, calculated as an exponentially-decaying empirical function.

- Out-of-bounds pressure surfaces (P < 200 mb or P > 1050 mb).
- Out-of-bounds heights (H < 0 m or H > 14000 m).
- Out-of-bounds relative humidities (RH < 0% or RH > 100%).
- Unexpected surface dry air (P > 950 mb and RH ≤ 15%).
- If either wind speed or direction equals zero, but the other of the dyad does not.
- Out-of-bounds wind direction, dd (dd < 0° or dd > 360°).
- Out-of-bounds wind speeds (< 0, or > a height-based empirical formula).
- Out-of-bounds wind shear (based on an exponential empirical function).
- It ensures that pressure decreases with height rather than increasing with it.

RWND_V2 flags questionable data; it then plots hodographs for flagged wind data and soundings for flagged thermodynamic data. You must manually examine these plots to determine the appropriate course required to correct (or accept) these data. Three courses of action can be followed; the first two require you to manually create the data file *MMMYY*_RWND_DEL.DAT. This data file must contain the date and time of the sounding, the station identification, and a flag indicating which course of action you have followed.

- Delete the entire sounding.
- Delete all the winds from the sounding while keeping the thermodynamic variables.
- The most common course of action is to make manual changes to the windowed data sets (RWND_*YY*US*MMM*_WNDW.DAT or RWND_*YY*CAN*MMM*_WNDW.DAT) to modify questionable data points. Here, you need to determine that the erroneous data are (a) simply due to a keypunch error, and correct them, or (b) that the erroneous data are truly incorrect, in which case remove the data and replace them with a "missing" flag.

  When you determine that a keypunch error is likely, you should attempt to determine which station(s) violate the continuity premise of the data you are questioning, and consider whether you can legitimately correct the typographical error. For example, you can replace a reported surface temperature of 28 °C for a station, when surrounding stations report temperatures that range from 17 °C to 20 °C, with 18 °C. You may, alternatively, want to discard the station's data (replacing them with the appropriate "missing" flag) from the windowed data set if you cannot determine how to correct the data. For example, a station may report a dew point temperature higher than a dry-bulb temperature (replace the temperature and dew point parameters with the "missing" flag), or consistently report winds as calm while also reporting a wind direction (replace the wind speed parameter with the "missing" flag).

Your staff meteorologist can help you determine which of these three courses to follow, and how to correct your data if that is necessary.

If you judge all sounding data to be correct, you must still create *MMMYY*_RWND_DEL.DAT and enter a single line of 9's as the uninterrupted series, 99999999 99999. The two sets of 9's are separated by two blanks, and are followed by any flag character.

After correcting/accepting the two windowed data sets, the corrected/accepted versions are then reprocessed through the reformatting, sorting, and merging stages to produce the revised version of RWND_*YYMMM*.DAT.[17]

The RWND_*YYMMM*.DAT and the *MMMYY*_RWND_DEL.DAT files are then read by the RWND_END routine. Again, the *MMMYY*_RWND_DEL.DAT contains a single line of data for each sounding that you have determined to have incorrect information, consisting of the date and time of the sounding, the station identification, and a flag that indicates your chosen course of action.

- If the flag is an **X**, RWND_END ignores the entire sounding.
- If the index is an **W**, only the winds are ignored.

For example, "85081012 13840 W" indicates that the 12Z sounding of August 10, 1985 for station 13840 must have its wind field data removed in the final data set. The RWND_*YYMMM*.DAT file header information, which is unnecessary for later processing, is also removed. If the data file *MMMYY*_RWND_DEL.DAT consists only of a single line of 9's and a flag character (i.e., all soundings are judged correct), then only the header information is removed.

The output file from RWND_END, RWND_*YYMMM*_VRFY.DAT, is now ready for the ROM processors.

3.3.7 Summary

Table 3.3 on the next page summarizes the flow of programs that read U.S. and Canadian upper-air data, perform quality control procedures on them, and produce the final upper-air meteorology data that are input to the ROM.

17. This is necessary in part because some procedures, e.g., interpolation of wind direction from an Ekman spiral, introduce additional layers into the data set.

TABLE 3.3. PREPROCESSOR PROGRAMS FOR UPPER-AIR METEOROLOGY OBSERVATIONS

United States Data				Canadian Data			
ROUTINE (.FOR)	INPUT (.DAT)	OUTPUT (.DAT)	COMMENTS	ROUTINE (.FOR)	INPUT (.DAT)	OUTPUT (.DAT)	COMMENTS
RWND_US_CLCT RWND_WNDW RWND_RFMT SORT	RAW DATA TAPES RWND_YYUS_CLCT RWND_YYUSMMM_WNDW RWND_YYUSMMM_RFMT	RWND_YYUS_CLCT RWND_YYUSMMM_WNDW RWND_YYUSMMM_RFMT RWND_YYUSMMM_SORT	VAX system sort	RWND_CAN_CLCT RWND_WNDW RWND_RFMT SORT	RAW DATA TAPES RWND_YYCAN_CLCT RWND_YYCANMMM_WNDW RWND_YYCANMMM_RFMT	RWND_YYCAN_CLCT RWND_YYCANMMM_WNDW RWND_YYCANMMM_RFMT RWND_YYCANMMM_SORT	VAX system sort

ROUTINE (.FOR)	INPUT (.DAT)	OUTPUT (.DAT)	COMMENTS
MERGE	RWND_YYUSMMM_SORT; RWND_YYCANMMM_SORT	RWND_YYMMM	VAX system merge
RWND_V2	RWND_YYMMM	plots flagged data	quality control
DEL or manual	uncorrectable soundings	MMMYY_RWND_DEL	quality control
manual	corrected errors	RWND_YYUSMMM_WNDW; RWND_YYCANMMM_WNDW	quality control
RWND_RFMT; SORT; MERGE	RWND_YYUSMMM_WNDW; RWND_YYCANMMM_WNDW	RWND_YYMMM	repeated
RWND_END	RWND_YYMMM and MMMYY_RWND_DEL	RWND_YYMMM_VRFY	final data set

3.4 PREPROCESSOR BUOY_CRCT

I. PREPROCESSOR FUNCTION

This preprocessor adds latitude and longitude values to buoy reports that do not include these data.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

   None

2. Nonstandard Input Files

   One nonstandard file, BUOY_*MMM*$_1$*MMM*$_2$*YY*_WNDW.DAT, that contains the data shown in Table 3.4 (note that sea level pressure is repeated) is required as input. The file is output by preprocessor BUOY_WNDW. The READ and FORMAT statements for the file are listed below.

```
100 READ (15, 150, END = 999) ITIME, ID, LAT, LON, ADATA, SLP,
   &                          WDIR, WSPD, TEMP, SST, PR
150 FORMAT (I8, 1X, I5, 1X, F5.2, 1X, F6.2, A42, 1X, 5F7.1, 1X,
   &        F7.1)
```

TABLE 3.4. BUOY_*MMM*$_1$*MMM*$_2$*YY*_WNDW.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ITIME	GMT	Integer*4	Date and time as *YYMMDDHH*
2	ID_n		Integer*4	Buoy identifier in WBAN format
3	LAT_n	° N	Real*4	Latitude
4	LON_n	° W	Real*4	Longitude
5	$ADATA_{n,h}$		Char*42	Missing cloud reports
6	$SLP_{n,h}$	mb	Real*4	Sea level pressure
7	$WDIR_{n,h}$	deg	Real*4	Wind direction
8	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
9	$TEMP_{n,h}$	°C	Real*4	Air temperature
10	$SST_{n,h}$	°C	Real*4	Sea surface temperature
11	$PR_{n,h}$	mb	Real*4	Sea level pressure

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

BUOY_CRCT outputs one nonstandard file, BUOY_*MMM*$_1$*MMM*$_2$*YY*_CRCT.DAT, that contains the data shown in Table 3.5 (note that sea level pressure is repeated). The file is input to preprocessor BUOY_INTP. The WRITE and FORMAT statements for BUOY_*MMM*$_1$*MMM*$_2$*YY*_CRCT.DAT are listed below.

```
      WRITE (16, 150) ITIME, ID, LAT, LON, ADATA, SLP, WDIR, WSPD,
     &                TEMP, SST, PR
  150 FORMAT (I8, 1X, I5, 1X, F5.2, 1X, F6.2, A42, 1X, 5F7.1, 1X,
     &        F7.1)
```

TABLE 3.5. BUOY_*MMM*$_1$*MMM*$_2$*YY*_CRCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ITIME	GMT	Integer*4	Date and time as *YYMMDDHH*
2	ID_n		Integer*4	Buoy identifier in WBAN format
3	LAT_n	° N	Real*4	Latitude
4	LON_n	° W	Real*4	Longitude
5	$ADATA_{n,h}$		Char*42	Missing cloud reports
6	$SLP_{n,h}$	mb	Real*4	Sea level pressure
7	$WDIR_{n,h}$	deg	Real*4	Wind direction
8	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
9	$TEMP_{n,h}$	°C	Real*4	Air temperature
10	$SST_{n,h}$	°C	Real*4	Sea surface temperature
11	$PR_{n,h}$	mb	Real*4	Sea level pressure

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

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	3	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	3	blocks
Executable file:	1	file	6	blocks
	3	files	12	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:06:00
Buffered I/O count:	523
Direct I/O count:	4154
Peak working-set size:	603
Peak virtual size:	3755

C. Space Requirements: Log and Print Files

BUOY_CRCT.LOG	3 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.6 shows the input file and output file space requirements.

TABLE 3.6. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	BUOY_MMM_1MMM_2YY_WNDW.DAT	Nonstd	±50000	7 months
Output	BUOY_MMM_1MMM_2YY_CRCT.DAT	Nonstd	±50000	7 months

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_CRCT.LNK follows:

```
$ LINK USER2$DISK:[DAO.BUOY]BUOY_CRCT.OBJ,
```
(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_CRCT.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM in USER2$DISK:[DAO.BUOY]BUOY_CRCT.COM
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_APROCT88_WNDW.DAT        INPUT
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_APROCT88_CRCT.DAT        OUTPUT
$ RUN USER2$DISK:[DAO.BUOY]BUOY_CRCT.EXE
```

(end of run stream)

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

A. Main Program

BUOY_CRCT

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

3.5 PREPROCESSOR BUOY_INTP

I. PREPROCESSOR FUNCTION

This preprocessor interpolates buoy data over periods that have data missing. We use a linear interpolation algorithm over no more than five consecutive hours at a time. BUOY_INTP also performs simple quality control; each hour's data is compared with data from the previous hour and the following hour, as well as with the mean of these two adjacent hours. If, in all three cases, the tolerances that you specify in the control cards are exceeded, the hour's data are replaced by the mean of the adjacent hours' values.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

One nonstandard file, BUOY_*MMM*$_1$*MMM*$_2$*YY*_CRCT.DAT, that contains the data shown in Table 3.7 is required as input. The file is output by preprocessor BUOY_CRCT. The READ and FORMAT statements for the file are listed below.

```
      READ (UNITIN, 150, END = 160) ITIME, BUOYID, LAT, LON, ADATA,
     &                              PR, DIR, SPD, T, H2OTMP
  150 FORMAT (I8, 1X, I5, 1X, F5.2, 1X, F6.2, A42, 1X, 5F7.1)
```

TABLE 3.7. BUOY_*MMM*$_1$*MMM*$_2$*YY*_CRCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ITIME	GMT	Integer*4	Date and time as *YYMMDDHH*
2	$BUOYID_n$		Integer*4	Buoy identifier in WBAN format
3	LAT_n	° N	Real*4	Latitude
4	LON_n	° W	Real*4	Longitude
5	$ADATA_{n,h}$		Char*42	Missing cloud reports
6	$PR_{n,h}$	mb	Real*4	Sea level pressure
7	$DIR_{n,h}$	deg	Real*4	Wind direction
8	$SPD_{n,h}$	$m{\cdot}s^{-1}$	Real*4	Wind speed
9	$T_{n,h}$	°C	Real*4	Air temperature
10	$H2OTMP_{n,h}$	°C	Real*4	Sea surface temperature

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

BUOY_INTP outputs one nonstandard file, BUOY_MMM_1MMM_2YY_INTP.DAT, that contains the data shown in Table 3.8. The file is input to preprocessors BUOY_STAT and BUOY_RFMT.[18,19] The WRITE and FORMAT statements for BUOY_MMM_1MMM_2YY_INTP.DAT are listed below.

```
      WRITE (UNITOUT, 850) TIMDAT(L), OLDID, OLDLAT, OLDLON,
     &                     ADATA, SLP(L), WDIR(L), WSOUT(L),
     &                     TMPOUT(L), SSTOUT(L), TDELTA
  850 FORMAT (I8, 1X, I5, 1X, F5.2, 1X, F6.2, A42, 1X,
     &        5F7.1, 1X, F7.1)
```

TABLE 3.8. BUOY_MMM_1MMM_2YY_INTP.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	TIMDAT	GMT	Integer*4	Date and time as *YYMMDDHH*
2	$OLDID_n$		Integer*4	Buoy identifier in WBAN format
3	$OLDLAT_n$	° N	Real*4	Latitude
4	$OLDLON_n$	° W	Real*4	Longitude
5	$ADATA_{n,h}$		Char*42	Missing cloud reports
6	$SLP_{n,h}$	mb	Real*4	Sea level pressure
7	$WDIR_{n,h}$	deg	Real*4	Wind direction
8	$WSOUT_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
9	$TMPOUT_{n,h}$	°C	Real*4	Air temperature
10	$SSTOUT_{n,h}$	°C	Real*4	Sea surface temperature
11	$TDELTA_{n,h}$	°C	Real*4	(Sea surface temp.) - (air temp.)

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

III. CONTROL CARDS

Three control cards are used in the format shown below. Table 3.9 defines the control card variables.

18. Prior to input to BUOY_RFMT, the file must be sorted using the VAX system's SORT utility.

19. If the raw buoy data arrive on two tapes, you will have two sorted data sets. Prior to being input to BUOY_RFMT, these must first be sorted as above, then merged using the VAX system's MERGE utility.

BEGTIM, ENDTIM
TOTHR
TMPTOL, SSTTOL, WSPDTOL

TABLE 3.9. CONTROL CARD VARIABLES

Variable Name	Unit	Description
BEGTIM	GMT	Beginning time of processing as *YYMMDDHH*
ENDTIM	GMT	Ending time of processing as *YYMMDDHH*
TOTHR	h	Total number of hours to be processed
TMPTOL	°C	Air temperature tolerance
SSTTOL	°C	Sea surface temperature tolerance
WSPDTOL	$m \cdot s^{-1}$	Wind speed tolerance

Note: if specified *tolerances* are exceeded, the hour's data are replaced by the mean of the adjacent hours' values.

Example:

88040100 88103123
5136
5.0 2.0 6.0

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	42	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	18	blocks
Executable file:	1	file	18	blocks
	3	files	78	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:06:00
Buffered I/O count:	549
Direct I/O count:	4262
Peak working-set size:	875
Peak virtual size:	3755

C. Space Requirements: Log and Print Files

BUOY_INTP.LOG	30 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.10 shows the input file and output file space requirements.

TABLE 3.10. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	BUOY_MMM_1MMM_2YY_CRCT.DAT	Nonstd	±50000	7 months
Output	BUOY_MMM_1MMM_2YY_INTP.DAT	Nonstd	±51000	7 months

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_INTP.LNK follows:

```
$ ! ACCOUNTING CODE
$ ASSIGN / USER USER2$DISK:[DAO.BUOY]BUOY_INTP.OBJ                  A1
$ ASSIGN / USER DISK$VAXSET:[ROMOBJ.UTILIO]GREG.OBJ                 A2
$ ASSIGN / USER DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ                A3
$ ASSIGN / USER DISK$VAXSET:[ROMOBJ.UTILIO]JULIAN.OBJ               A4
$ LINK -
  / EXECUTABLE = USER2$DISK:[DAO.BUOY]BUOY_INTP.EXE -
  A1, -
  A2, -
  A3, -
  A4
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_INTP.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM in USER2$DISK:[DAO.BUOY]BUOY_INTP.COM
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_APROCT88_CRCT.DAT        INPUT
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_APROCT88_INTP.DAT        OUTPUT
$ RUN USER2$DISK:[DAO.BUOY]BUOY_INTP.EXE
  88040100    88103123          !  TIME SPAN
  5136                          !  TOTAL HOURS IN TIME SPAN
  5.0  2.0  6.0                 !  TEMP, SST AND WSPD TOLERANCES
```

(end of run stream)

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

A. Main Program

BUOY_INTP

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

USER2$DISK:[ROMLIB.UTILIO]

GREG

B. Functions

USER2$DISK:[ROMLIB.UTILIO]

JUNIT
JULIAN

IX. INCLUDE FILES

None

3.6 PREPROCESSOR BUOY_RFMT

I. PREPROCESSOR FUNCTION

This preprocessor performs the final reformatting to the SURMET1-type arrangement on (1) the hourly buoy data, and (2) the monthly-averaged buoy data.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

Either of the two nonstandard files below is required each time BUOY_RFMT is run.

a. BUOY_*MMM*$_1$*MMM*$_2$*YY*_SORT.DAT (Hourly buoy data)

BUOY_*MMM*$_1$*MMM*$_2$*YY*_SORT.DAT is the result of using the VAX system's SORT utility on BUOY_*MMM*$_1$*MMM*$_2$*YY*_INTP.DAT, the output from preprocessor BUOY_INTP. (Recall that if the raw buoy data arrive on two tapes, you will have two sorted data sets that must be system MERGED prior to being used as input to this preprocessor.) The READ and FORMAT statements to input the file are listed below, and its parameters are shown in Table 3.11.

```
160 READ (UNITIN, 180, END = 999) DATIME, WBAN, INFO
180 FORMAT (I8.8, 1X, I5.5, A99)
```

TABLE 3.11. BUOY_*MMM*$_1$*MMM*$_2$*YY*_SORT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATIME	GMT	Integer*4	Date and time as *YYMMDDHH*
2	$WBAN_n$		Integer*4	Buoy identifier in WBAN format
3	$INFO_{n,h}$		Char*99	Data stream, comprising:
	LAT_n	° N	Real*4	Latitude
	LON_n	° W	Real*4	Longitude
	$ADATA_{n,h}$		Char*42	Substitution for missing cloud reports
	$PR_{n,h}$	mb	Real*4	Sea level pressure
	$DIR_{n,h}$	deg	Real*4	Wind direction
	$SPD_{n,h}$	$m{\cdot}s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Air temperature (AT)
	$SST_{n,h}$	°C	Real*4	Sea surface temperature (SST)
	$TDELT_{n,h}$	°C	Real*4	(SST) - (AT)

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

b. BUOY_STAT*YY*.DAT (Monthly-averaged buoy data)

BUOY_AVG*YY*.DAT, containing the data shown in Table 3.12, is written by STAT_RFMT. The READ and FORMAT statements are listed below.

```
160 READ (UNITIN, 180, END = 999) DATIME, WBAN, INFO
180 FORMAT (I8.8, 1X, I5.5, A99)
```

TABLE 3.12. BUOY_AVG*YY*.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATIME	GMT	Integer*4	First hour of the month as *YYMMDDHH*
2	$WBAN_n$		Integer*4	Buoy identifier in WBAN format
3	$INFO_n$		Char*99	Data stream, comprising:
	LAT_n	° N	Real*4	Latitude
	LON_n	° W	Real*4	Longitude
	$ADATA_n$		Char*44	Substitution for cloud data
	$BDATA_n$		Char*20	Substitution for pressure and wind data
	MT_n	°C	Real*4	Monthly average air temperature (AT)
	$MSST_n$	°C	Real*4	Monthly average sea surface temperature (SST)
	$DELT_n$	°C	Real*4	Monthly average (SST) - (AT)

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

One of the two files below is output each time BUOY_RFMT is run; the files contain the finalized buoy data.

a. BUOY_MMM_1MMM_2YY.DAT (Hourly buoy data)

The WRITE and FORMAT statements for this file are listed below, and its parameters are shown in Table 3.13.

```
    WRITE (UNITOUT, 200) OLDTIM, I
200 FORMAT (I8.8, 5X, I3)

    WRITE (UNITOUT, 230) STNID(K), ADATA(K)
230 FORMAT (I5.5, A99)
```

TABLE 3.13. BUOY_MMM_1MMM_2YY.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	OLDTIM	GMT	Integer*4	Date and time as *YYMMDDHH*
2	I		Integer*4	No. of buoys reporting for this hour
3	$STNID_n$		Integer*4	Buoy identifier in WBAN format
4	$ADATA_{n,h}$		Char*99	Data stream, comprising:
	LAT_n	° N	Real*4	Latitude
	LON_n	° W	Real*4	Longitude
	$CLDMID_{n,h}$		Char*42	Substitution for missing cloud reports
	$PR_{n,h}$	mb	Real*4	Sea level pressure
	$DIR_{n,h}$	deg	Real*4	Wind direction
	$SPD_{n,h}$	$m{\cdot}s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Air temperature (AT)
	$SST_{n,h}$	°C	Real*4	Sea surface temperature (SST)
	$TDIF_{n,h}$	°C	Real*4	(SST) - (AT)

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

b. SURMETM (Monthly-averaged buoy data)

The WRITE and FORMAT statements for this file are listed below, and its parameters are shown in Table 3.14.

```
    WRITE (UNITOUT, 200) OLDTIM, I
200 FORMAT (I8.8, 5X, I3)

    WRITE (UNITOUT, 230) STNID(K), ADATA(K)
230 FORMAT (I5.5, A99)
```

TABLE 3.14. SURMETM PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	OLDTIM	GMT	Integer*4	First hour of the month as *YYMMDDHH*
2	I		Integer*4	No. of buoys reporting for this hour
3	$STNID_n$		Integer*4	Buoy identifier in WBAN format
4	$ADATA_n$		Char*99	Data stream, comprising:
	LAT_n	° N	Real*4	Latitude
	LON_n	° W	Real*4	Longitude
	$CLDS_n$		Char*44	Substitution for cloud data
	PW_n		Char*20	Substitution for pressure and wind data
	MAT_n	°C	Real*4	Monthly average air temperature (AT)
	$MSST_n$	°C	Real*4	Monthly average sea surface temperature (SST)
	$TDIF_n$	°C	Real*4	Monthly average (SST) - (AT)

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

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	6	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	3	blocks
Executable file:	1	file	9	blocks
	3	files	18	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:04:00
Buffered I/O count:	871
Direct I/O count:	8621
Peak working-set size:	603
Peak virtual size:	3755

C. Space Requirements: Log and Print Files

BUOY_RFMT.LOG	3 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.15 shows the input file and output file space requirements.

TABLE 3.15. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	BUOY_MMM_1MMM_2YY_INTP.DAT	Nonstd	±50000	7 months
	or			7 months
	BUOY_AVG*YY*.DAT	Nonstd	±100	
Output	BUOY_MMM_1MMM_2YY.DAT	Nonstd	±46000	7 months
	or			7 months
	SURMETM	Nonstd	±100	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_RFMT.LNK follows:

```
$ ! ACCOUNTING CODE
$ LINK    USER2$DISK:[DAO.BUOY]BUOY_INTP.OBJ, -
$         DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_RFMT.COM for hourly data follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_APROCT88_INTP.DAT      INPUT
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_APROCT88.DAT           OUTPUT
$ RUN USER2$DISK:[DAO.BUOY]BUOY_RFMT.EXE
```

(end of run stream)

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

A. Main Program

BUOY_RFMT

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

USER2$DISK:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.7 PREPROCESSOR BUOY_STAT

I. PREPROCESSOR FUNCTION

This preprocessor computes the statistics of monthly-averaged buoy (1) air temperatures, (2) sea surface temperatures, and (3) air-sea temperature differences.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

   None

2. Nonstandard Input Files

   One nonstandard file, BUOY_*MMM*$_1$*MMM*$_2$*YY*_INTP.DAT, that contains the data shown in Table 3.16 is required as input. The file is output by preprocessor BUOY_INTP. The READ and FORMAT statements for the file are listed below.

```
    READ (UNITIN, 150, END = 160) ITIME, BUOYID, LAT, LON,
   &                              T, H2OTMP, TDELTA
150 FORMAT (I8, 1X, I5, 1X, F5.2, 1X, F6.2, 64X, 2F7.1, F8.1)
```

TABLE 3.16. BUOY_*MMM*$_1$*MMM*$_2$*YY*_CRCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ITIME	GMT	Integer*4	Date and time as *YYMMDDHH*
2	$BUOYID_n$		Integer*4	Buoy identifier in WBAN format
3	LAT_n	° N	Real*4	Latitude
4	LON_n	° W	Real*4	Longitude
5	$T_{n,h}$	°C	Real*4	Air temperature
6	$H2OTMP_{n,h}$	°C	Real*4	Sea surface temperature
7	$TDELTA_{n,h}$	°C	Real*4	(Sea surface temp.) - (air temp.)

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

B. Output Files

1. Standard Output Files

   None

2. Nonstandard Output Files

   None (the statistics are written to the log file).

3. Log File

BUOY_STAT outputs to the log file BUOY_STAT.LOG that contains the data shown in Table 3.17. The file is input to preprocessor STAT_RFMT. The WRITE and FORMAT statements are listed below.

```
          WRITE (6, 333) OLDID, OLDLAT, OLDLON, OLDMON
333       FORMAT ('  STATISTICAL SUMMARY FOR BUOY ',I5,' (', F5.2,','
    &         ,F6.2, ') MONTH: ', I2.2,'  FOLLOWS: ')
          PRINT *, ' '

             WRITE (6, 340) K
340          FORMAT ('  HOUR: ', I2)

             WRITE (6, 620) AVTMP, TCNT(K)
620          FORMAT (' AVG TEMP : ', F7.1, ' BASED ON ',I4,' REPORTS')
             WRITE (6, 621) AVSST, SCNT(K)
621          FORMAT (' AVG SST  : ', F7.1, ' BASED ON ',I4,' REPORTS')
             WRITE (6, 622) AVDELT, DCNT(K)
622          FORMAT (' AVG DELT : ', F7.1, ' BASED ON ',I4,' REPORTS')
```

TABLE 3.17. BUOY_STAT.LOG PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DUMMY		Char*80	Dummy record
2	OLDID$_n$		Integer*4	Buoy identifier in WBAN format
3	OLDLAT$_n$	° N	Real*4	Latitude
4	OLDLON$_n$	° W	Real*4	Longitude
5	OLDMON$_n$		Integer*4	Month as *MM*
6	K	GMT	Integer*4	Hour as *HH*
7	AVTMP$_n$	°C	Real*4	Hourly averaged air temperature (AT)
8	TCNT$_n$		Integer*4	Number of reports averaged
9	AVSST$_n$	°C	Real*4	Hourly averaged sea surface temperature (SST)
10	SCNT$_n$		Integer*4	Number of reports averaged
11	AVDELT$_n$	°C	Real*4	Averaged (SST) - (AT)
12	DCNT$_n$		Integer*4	Number of reports averaged

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

III. CONTROL CARDS

One control card is used in the format shown below. Table 3.18 defines the control card variables; they are used in spatially windowing the buoy data.

MINLAT, MAXLAT, MINLON, MAXLON

TABLE 3.18. CONTROL CARD VARIABLES

Variable Name	Unit	Description
MINLAT	° N	Southern latitude boundary
MAXLAT	° N	Northern latitude boundary
MINLON	° W	Eastern longitude boundary
MAXLON	° W	Western longitude boundary

Example:

20.0 50.0 64.0 99.0

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	21	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	12	blocks
Executable file:	1	file	18	blocks
	3	files	51	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:02:00
Buffered I/O count:	212
Direct I/O count:	3278
Peak working-set size:	597
Peak virtual size:	3660

C. Space Requirements: Log and Print Files

BUOY_STAT.LOG	1353 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.19 shows the input file and output file space requirements.

TABLE 3.19. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	BUOY_*MMM*$_1$*MMM*$_2$*YY*_INTP.DAT	Nonstd	±51000	7 months
Output	BUOY_STAT.LOG	Nonstd	±1300	7 months

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_STAT.LNK follows:

```
$ ! ACCOUNTING CODE
$ ASSIGN / USER USER2$DISK:[DAO.BUOY]BUOY_STAT.OBJ                A1
$ ASSIGN / USER DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ              A2
$ LINK -
  / EXECUTABLE = USER2$DISK:[DAO.BUOY]BUOY_STAT.EXE -
  A1, -
  A2
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_STAT.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM in USER2$DISK:[DAO.BUOY]BUOY_STAT.COM
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_APROCT88_INTP.DAT        INPUT
$ RUN USER2$DISK:[DAO.BUOY]BUOY_STAT.EXE
 20.0  50.0  64.0  99.0
```

(end of run stream)

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

A. Main Program

BUOY_STAT

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

USER2$DISK:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.8 PREPROCESSOR BUOY_WNDW

I. PREPROCESSOR FUNCTION

This preprocessor (1) windows buoy data by time, and (2) modifies the original buoy identifiers to a WBAN-type. The preprocessor outputs the data in a pre-SURMET1 format, and provides a variable for each SURMET1 field.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. BUOY_XTABLE.DAT

BUOY_XTABLE.DAT is the "look-up" file that contains our pseudo-WBAN identifiers for the buoys. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 3.20.

```
50  READ (UNITX, 60, END = 70) BUOYID(K), FAKEID(K)
60  FORMAT (1X, A5, 2X, I5.5)
```

TABLE 3.20. BUOY_XTABLE.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	BUOYID		Char*5	Original buoy identifier
2	FAKEID		Integer*4	Pseudo-WBAN buoy identifier

b. XTR_BUOYYY.DAT

This file contains the buoy parameters output by preprocessor XTRACTLL. The READ and FORMAT statements for the file are listed below, and its parameters are shown in Table 3.21.

```
80 READ (UNITIN, 100, END = 999) ID, YR, MON, DAY, HR, LAT,
  &                               ALAT, LON, ALON, TEMP, SST,
  &                               SLP, ADIR, ASPD
100 FORMAT (9X, A5, 2X, I4, 3I2.2, 2X, F3.1, A1, F4.1, A1, 3X,
  &        F4.1, 4X, F4.1, F5.1, A3, A4)
```

TABLE 3.21. XTR_BUOYYY.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ID_n		Char*5	Buoy identifier
2	YR		Integer*4	Year
3	MON		Integer*4	Month
4	DAY		Integer*4	Day
5	HR	GMT	Integer*4	Hour
6	LAT_n		Real*4	Latitude
7	ALAT		Char*1	Latitude hemisphere indicator
8	LON_n		Real*4	Longitude
9	ALON		Char*1	Longitude hemisphere indicator
10	$TEMP_{n,h}$	°C	Real*4	Air temperature
11	$SST_{n,h}$	°C	Real*4	Sea surface temperature
12	$SLP_{n,h}$	mb	Real*4	Sea level pressure
13	$ADIR_{n,h}$	deg	Char*3	Wind direction string
14	$ASPD_{n,h}$	kts	Char*4	Wind speed string

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

BUOY_WNDW outputs one nonstandard file, BUOY_*MMM*$_1$*MMM*$_2$*YY*_WNDW.DAT, that contains the data shown in Table 3.22 (note that sea level pressure is repeated). The file is input to preprocessor BUOY_CRCT. The WRITE and FORMAT statements for BUOY_*MMM*$_1$*MMM*$_2$*YY*_WNDW.DAT are listed below.

```
      WRITE (UNITOUT, 150) DATIME, NEWID, LAT, LON, CLDMIS, SLP,
     &                     WDIR, WSPD, TEMP, SST, SLP
  150 FORMAT (I8.8, 1X, I5.5, 1X, F5.2, 1X, F6.2, 1X, A41, 1X,
     &        5F7.1, F8.1)
```

TABLE 3.22. BUOY_*MMM*$_1$*MMM*$_2$*YY*_WNDW.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATIME	GMT	Integer*4	Date and time as *YYMMDDHH*
2	$NEWID_n$		Integer*4	Buoy identifier in WBAN format
3	LAT_n	° N	Real*4	Latitude
4	LON_n	° W	Real*4	Longitude
5	$CLDMIS_{n,h}$		Char*41	Missing cloud reports
6	$SLP_{n,h}$	mb	Real*4	Sea level pressure
7	$WDIR_{n,h}$	deg	Real*4	Wind direction
8	$WSPD_{n,h}$	$m{\cdot}s^{-1}$	Real*4	Wind speed
9	$TEMP_{n,h}$	°C	Real*4	Air temperature
10	$SST_{n,h}$	°C	Real*4	Sea surface temperature
11	$SLP_{n,h}$	mb	Real*4	Sea level pressure

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

III. CONTROL CARDS

One control card is used in the format shown below. Table 3.23 defines the control card variables.

BEGMON, ENDMON

TABLE 3.23. CONTROL CARD VARIABLES

Variable Name	Unit	Description
BEGMON		Beginning month of output as *MM*
ENDMON		Ending month of output as *MM*

Example:

04, 10

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	9	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	9	blocks
Executable file:	1	file	12	blocks
	3	files	30	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:08:00
Buffered I/O count:	605
Direct I/O count:	6108
Peak working-set size:	597
Peak virtual size:	3660

C. Space Requirements: Log and Print Files

BUOY_WNDW.LOG	6 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.24 shows the input file and output file space requirements.

TABLE 3.24. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	XTR_BUOYYY.DAT	Nonstd	±80000	1 year
Output	BUOY_*MMM*$_1$*MMM*$_2$*YY*_WNDW.DAT	Nonstd	±50000	7 months

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_WNDW.LNK follows:

```
$ LINK USER2$DISK:[DAO.BUOY]BUOY_WNDW.OBJ, -
      DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

BUOY_WNDW.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM in USER2$DISK:[DAO.BUOY]BUOY_WNDW.COM
$ ASSIGN MET27:[SCRATCH.DAO]XTR_BUOY88.DAT              INPUT
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_XTABLE.DAT             XTABLE
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_APROCT88_WNDW.DAT      OUTPUT
$ RUN USER2$DISK:[DAO.BUOY]BUOY_WNDW.EXE
  04, 10
```

(end of run stream)

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

A. Main Program

BUOY_WNDW

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.9 PREPROCESSOR COPYTAPE (Data tape to VAX-system tape copy)

I. PREPROCESSOR FUNCTION

This preprocessor copies a raw data tape onto a VAX-system tape in order to ensure that valuable data are not lost. You need to understand the limitations of this preprocessor. One feature of this program is that a parity error will not stop the copying of a tape - *the bad tape block on the original will not be written to the copy*. However, no message stating that this has occurred is written to the log file. Therefore, we recommend that you use a scanning routine to compare the data on the original tape to the data on the copy. If such a routine is unavailable, you should monitor the error counts on the tape drives, Although an increase in the error count does not necessarily indicate a bad copy, no increase in the error count guarantees a good copy.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

Any ASCII tape can be used as input.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

The output is written to a VAX-system tape. The program is hard-coded to skip the label of the system tape.

III. CONTROL CARDS

One control card is used in the format shown below. Table 3.25 defines the control card variable.

IFILE

TABLE 3.25. CONTROL CARD VARIABLE

Variable Name	Unit	Description
IFILE		Number of files to be copied; this number represents the minimum number of files to be copied. A number greater than the actual number of files on the original tape guarantees that the entire original tape is copied.

Example:

99

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	12	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	2	files	9	blocks
Executable file:	1	file	9	blocks
	5	files	30	blocks

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

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:00:15 per tape
Buffered I/O count:	392
Direct I/O count:	27219
Peak working-set size:	530
Peak virtual size:	4564

C. Space Requirements: Log and Print Files

COPYTAPE.LOG	24 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

None

E. Space Requirements: Tape Files

Two tapes are required, an original and a blank VAX-system tape. Table 3.26 shows the input file and output file space requirements for a representative tape.

TABLE 3.26. TAPE FILE SPACE REQUIREMENTS

File Group	Tape Number	File Number	Record Length (bytes)	Block Size (bytes)
Input	B20012	1	495	24750
Output	102479	1 (label)	80	80
		2	495	24750

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

COPYTAPE.LNK follows:

```
$ ASSIGN USER2$DISK:[DAO.METTAPE]COPYTAPE.OBJ                    A1
$ ASSIGN USER2$DISK:[DAO.METTAPE]VATAPE.OBJ                      A2
$ LINK -
  /EXECUTABLE = USER2$DISK:[DAO.METTAPE]COPYTAPE.EXE -
  /MAP = MET1:[S01S]COPYTAPE.MAP -
  A1, -
  A2
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

COPYTAPE.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ TAPE SELECT ADRIVE BTAPE
$ TAPE LOAD ADRIVE B20012
$ MOUNT/NOWRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=12000  -
  ADRIVE B20012 OLDTAP:
$ TAPE SELECT BDRIVE 102479
$ TAPE LOAD/RING BDRIVE 102479
$ MOUNT/WRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=12000  -
  BDRIVE 102479 COPTAP:
$ SHOW DEV/FULL ADRIVE
$ SHOW DEV/FULL BDRIVE
$ SHOW LOGICAL OLDTAP
$ SHOW LOGICAL COPTAP
$ ! SET MAGTAPE/SKIP=FILES:1 OLDTAP:      ! Use this line only if
$ !                                       ! old tape has label
$ RUN USER2$DISK:[DAO.METTAPE]COPYTAPE.EXE
  99
$ RUN USER2$DISK:[BRO]TSCAN.EXE
ADRIVE
$ RUN USER2$DISK:[BRO]TSCAN.EXE
BDRIVE
$ @REWIND OLDTAP:
$ @REWIND COPTAP:
$ SHOW DEV/FULL ADRIVE
$ SHOW DEV/FULL BDRIVE
$ DISMOUNT OLDTAP:
$ DEALLOCATE ADRIVE
$ DISMOUNT COPTAP:
$ DEALLOCATE TAPEDRIVE
$ DEALLOCATE BDRIVE
```

(end of run stream)

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

A. Main Program

COPYTAPE

B. Subroutines

USER2$DISK:[DAO.METTAPE]

VATAPE

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

3.10 PREPROCESSOR COUNT_STNPERHR (METEOROLOGY QUALITY CONTROL)

I. PREPROCESSOR FUNCTION

This preprocessor scans the output of preprocessor SFC_VRFY and flags hours that have either an unusually high or an unusually low number of reporting stations. A message is also written for each hour that does not numerically follow the previous hour; thus, COUNT_STNPERHR writes a message at the start of each day to indicate a possible data gap. You should ignore those reports that provide no useful information. If a data gap is present, or if some hours have only partial data, then you will need to run an additional preprocessor, SFCR_HRFILL.

We intend preprocessor COUNT_STNPERHR to be a guide that can alert you to potential problems. As such, you are required to set your own standards as to what constitutes a data-poor hour and what does not. You will need some prior knowledge of the input file in order to input proper values for the control cards. You must manually examine many hours of data in the input file to gain an idea of the normal daily variation in the number of reporting stations. You should also run program SHOWMET for one hour of data to ensure that there is adequate spatial coverage of stations for an arbitrary hour and for your needs. You will have a benchmark for setting the limits of the control cards after taking these steps.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

COUNT_STNPERHR uses one nonstandard input file, SFC_*MMMYY*_VRFY.DAT, and reads the parameters shown in Table 3.27. This file is output by preprocessor SFC_VRFY; its READ and FORMAT statements are listed below.

```
    READ (12, 30, END = 99) DATIME, NUMSTN
30  FORMAT (I8.8, 5X, I3)

    READ (12, 31) HOGWASH
31  FORMAT(A1)
```

TABLE 3.27. SFC_*MMMYY*_VRFY.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATIME	GMT	Integer*4	Date and time as *YYMMDDHH*
2	NUMSTN		Integer*4	No. of stations reporting for this hour
3	HOGWASH		Char*1	Dummy variable: skips hourly data

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

None; the output is written to the log file.

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 3.28 defines the control card variables.

MINSTA, MAXSTA

TABLE 3.28. CONTROL CARD VARIABLES

Variable Name	Unit	Description
MINSTA		Minimum number of stations reporting per hour
MAXSTA		Maximum number of stations reporting per hour

Example:

285, 315

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	3	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	6	blocks
Executable file:	1	file	6	blocks
	3	files	15	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:01:20
Buffered I/O count:	193
Direct I/O count:	3390
Peak working-set size:	530
Peak virtual size:	3771

C. Space Requirements: Log and Print Files

COUNT_STNPERHR.LOG 9 blocks
Print Files: None

D. Space Requirements: Input and Output Files

Table 3.29 shows the input file and output file space requirements.

TABLE 3.29. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SFC_*MMMYY*_VRFY.DAT	Nonstd	46278	1 month
Output	N/A			

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

COUNT_STNPERHR.LNK follows:

```
$ LINK   USER2$DISK:[DAO.ROMWORK]COUNT_STNPERHR.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

COUNT_STNPERHR.COM follows:

```
$ ! PREP0001
$ ASSIGN MET27:[SCRATCH.DAO]SFC_APR88_VRFY.DAT          INPUT
$ RUN USER2$DISK:[DAO.ROMWORK]COUNT_STNPERHR.EXE
  285, 315
```

(end of run stream)

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

A. Main Program

COUNT_STNPERHR

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

3.11 PREPROCESSOR DEL

I. PREPROCESSOR FUNCTION

This preprocessor is an alternative to your interactive (manual) correction of the upper-air data following your inspection of the plots from preprocessor RWND_V2. Rather than enter an editor to make line-by-line corrections, DEL offers an automated course of action. DEL has two modes of operation: (1) it replaces erroneous data with the "missing" value -9999.0 for all nonsurface data; (2) it replaces erroneous surface temperature, dew point, and humidity data with "appropriate" data. Note that these appropriate data are fully discussed in Section 3.3.6.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

One nonstandard file, RWND_*YYUSMMM*_RWND.DAT for U.S. data, or, for Canadian data, RWND_*YYCANMMM*_RWND.DAT, is the input to the preprocessor. The file consists of a single month's data, including station identification codes, the date and time (as GMT) of the observation, the pressure level and its geopotential height, and meteorological parameters associated with that level -- temperature, dew point, relative humidity, wind direction, and wind speed. Both mandatory (including standard) and significant pressure levels are currently reported (see Section 3.3.1 for definitions). The READ and FORMAT statements follow, and the file's parameters are shown in Table 3.30.

```
      READ(8,201,END=6)IY,IM,ID,IH,ISTA,II(I),N,P(I),H(I),T(I),TD(I),
     & RH(I),WD(I),WS(I),ALA(I),ALO(I)
  201 FORMAT(4I3,I6,I4,I2,7F8.1,2F7.2)
```

TABLE 3.30. RWND_*YYUSMMM*_RWND.DAT AND RWND_*YYCANMMM*_RWND.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IY		Integer*4	Year as *YY*
2	IM		Integer*4	Month as *MM*
3	ID		Integer*4	Day as *DD*
4	IH	GMT	Integer*4	Hour as *HH*
5	ISTA		Integer*4	Station identifier
6	II		Integer*4	Level indicator
7	N		Integer*4	Number of levels
8	$P_{m,II}$	mb	Real*4	Pressure at level II
9	$H_{m,II}$	m	Real*4	Height at level II
10	$T_{m,II}$	°C	Real*4	Temperature at level II
11	$TD_{m,II}$	°C	Real*4	Dew point at level II
12	$RH_{m,II}$	%	Real*4	Relative humidity at level II
13	$WD_{m,II}$	deg	Real*4	Wind direction at level II
14	$WS_{m,II}$	$m \cdot s^{-1}$	Real*4	Wind speed at level II
15	ALA_m	° N	Real*4	Station latitude
16	ALO_m	° W	Real*4	Station longitude

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

RWND_*YYUSMMM*_UPDT.DAT or RWND_*YYCANMMM*_UPDT.DAT are generated as corrected versions of the input data. These files contain parameters identical to those of the input files above, and are produced by the following WRITE and FORMAT statements:

```
      WRITE(9,202)IY,IM,ID,IH,ISTA,II(J),N,P(J),H(J),T(J),TD(J),
     & RH(J),WD(J),WS(J),ALA(J),ALO(J)
  202 FORMAT(4(1X,I2.2),1X,I5.5,2X,2I2.2,7F8.1,2F7.2)
```

III. CONTROL CARDS

Four control cards are used, in the format shown below:

KST,KYMDH,IDX,MMM,NNN
HT
HTT
HHTT,TTPP,DDPP,RRHH

The iterative code that effects the changes in the data files is listed below, and the control card variables are shown in Table 3.31.

```
      READ(5,50,END=6) KST,KYMDH,IDX,MMM,NNN
      IF(IDX.EQ.1.OR.IDX.EQ.3) READ(5,51) (HT(I),I=1,MMM)
      IF(IDX.EQ.2.OR.IDX.EQ.3) READ(5,51) (HTT(I),I=1,NNN)
      IF(IDX.EQ.7) READ(5,51) HHTT,TTPP,DDPP,RRHH
   50 FORMAT(I6,2X,I8,I3,2I5)
   51 FORMAT(10F7.0,/,10F7.0)
```

TABLE 3.31. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Data Type	Description
1	KST		Integer*4	Station ID (99999 ⇒ end of control data)
2	KYMDH		Integer*4	Date and hour as *YYMMDDHH*
3	IDX_m		Integer*4	Index (see note below)
4	MMM_m		Integer*4	No. of levels of thermodynamic data
5	NNN_m		Integer*4	No. of levels of wind data
6	$HT_{m,MMM}$	m	Real*4	Height of each thermodynamic level
7	$HTT_{m,NNN}$	m	Real*4	Height of each wind level
8	$HHTT_m$	m	Real*4	Height of surface level for thermodynamic data
9	$TTPP_m$	°C	Real*4	Temperature at the surface
10	$DDPP_m$	°C	Real*4	Dew point at the surface
11	$RRHH_m$	%	Real*4	Relative humidity at the surface

Note: IDX = 1 ⇒ erroneous nonsurface temperature data
2 ⇒ erroneous nonsurface wind data
3 ⇒ erroneous nonsurface temperature and wind data
7 ⇒ erroneous surface temperature, dew point, and humidity data
m = upper-air station identifier (one or more digits)

Example:

```
3937 88051012 1 3 0
3039. 3179. 3214.
12717 88052012 2 0 1
1042.
12912 88050812 7 0 0
33. 23.6 21.4 87.0
```

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION, U.S. DATA

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	8	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	8	blocks
Executable file:	1	file	9	blocks
	3	files	25	blocks

B. Execution Time Requirements (Representative Values for a One-Month Data Span):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:04:31
Buffered I/O count:	249
Direct I/O count:	2673
Peak working-set size:	592
Peak virtual size:	4286

C. Space Requirements: Log and Print Files

DEL.LOG:	282 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.32 shows the input file and output file space requirements for May 1988 data. Note that both the input and output storage requirements will vary from month to month.

TABLE 3.32. I/O FILE SPACE REQUIREMENTS FOR MAY 1988 U.S. DATA

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	RWND_88USMAY_WNDW.DAT	Nonstd	17444	1 month
Output	RWND_88USMAY_UPDT.DAT	Nonstd	17444	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

DEL.LNK follows:

```
$ ASSIGN USER2$DISK:[RTT.ROM] A
$ LINK A:DEL
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

DEL.COM for U.S. data follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ !
$ ! RUN STREAM IS IN THE FILE USER2$DISK:[RTT.ROM]DEL.RUNMAY88US
$ !
$ ASSIGN MET25:[RTT]RWND_US88MAY_WNDW.DAT      INPUT1
$ ASSIGN MET25:[RTT]RWND_US88MAY_UPDT.DAT      OUTPUT
$ RUN USER2$DISK:[RTT.ROM]DEL
  3860  88051500  1    4    0
  6068.  6131.  6291.  6823.
  3860  88052112  1    1    0
  6800.
  3860  88052312  1    1    0
 10322.
  3879  88050812  1    1    0
  6539.
                .
                .
                .
 94823  88051100  1    4    0
  4182.  4195.  4220.  4245.
 94823  88051612  1    1    0
   817.
 99999  99999999  9   99   99
```

(end of run stream)

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

A. Main Program

DEL

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

3.12 PREPROCESSOR OFFTAPE

I. PREPROCESSOR FUNCTION

This preprocessor extracts buoy data from tape and writes them to a disk file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

Any ASCII tape with a fixed block size can be used as input. In practice, we only use the copy (from preprocessor COPYTAPE) of the original National Buoy Data Center tape.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

OFFTAPE produces one nonstandard file, BUOYYY.DAT, that contains the parameter shown in Table 3.33. The WRITE statement for this file is listed below.

```
WRITE(OUTU) BUF((I-1)*RCDSZ+1:I*RCDSZ)
```

TABLE 3.33. BUOYYY.DAT PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	BUF		Char*10640	Data stream, written as 80 characters per record

III. CONTROL CARDS

One control card is used in the format shown below. Table 3.34 defines the control card variable.

IBC, RCDSZ, PARCND, IRPT

TABLE 3.34. CONTROL CARD VARIABLES

Variable Name	Unit	Description
IBC	bytes	Block size of input tape
RCDSZ	bytes	Output record size
PARCND		Parity error condition code: T ⇒ terminate on parity error S ⇒ skip bad block and continue processing
IRPT		Number of files to skip before beginning processing

Example:

10640
152
T
1

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	18	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	2	files	12	blocks
Executable file:	1	file	15	blocks
	5	files	45	blocks

B. Execution Time Requirements (per tape):

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:02:00
Buffered I/O count:	1468
Direct I/O count:	13225
Peak working-set size:	603
Peak virtual size:	3755

C. Space Requirements: Log and Print Files

OFFTAPE.LOG	6 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.35 shows the input and output file space requirements.

TABLE 3.35. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	N/A			
Output	BUOYYY.DAT	Nonstd	±180000	1 year

E. Space Requirements: Tape Files

One input tape is required. Table 3.36 shows the input file space requirements for a representative tape.

TABLE 3.36. TAPE FILE SPACE REQUIREMENTS

File Group	Tape Number	File Number	Record Length (bytes)	Block Size (bytes)
Input	101229	1 (label)	80	80
		2	80	10640

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

OFFTAPE.LNK follows:

```
$ LINK   USER2$DISK:[SER.STREAM.TAPES]OFFTAPE.OBJ, -
         USER2$DISK:[DAO.METTAPE]VATAPE.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

OFFTAPE.COM follows:

```
$ ! ACCOUNTING CODE
$ !
$ ! This program copies a file from tape to disk.
$ !
$ ! EXPECTED INPUT:
$ !
$ !   integer - BLOCKSIZE IN BYTES
$ !      integer - RECORDSIZE IN BYTES
$ !   character*1 - STATUS ON PARITY ERROR (T TO TERMINATE, S TO SKIP)
$ !   integer - NUMBER OF INPUT FILES TO SKIP
$ !
$ ! Additionally the program expects to read from "INTAPE"
$ ! and write to "OUTDISK"
$ !
$ TAPE SELECT DRIVE1 101229
$ TAPE LOAD/NORING DRIVE1 101229
$ MOUNT/NOWRITE/FOREIGN/BLOCKSIZE=10640 DRIVE1 101229 INTAPE:
$ ASSIGN MET27:[SCRATCH.DAO]BUOY88.DAT   OUTDISK
$ RUN USER2$DISK:[SER.STREAM.TAPES]OFFTAPE.EXE
10640
152
T
1
$ !
$ @REWIND INTAPE:
$ DISMOUNT INTAPE:
$ DEALLOCATE DRIVE1
$ EXIT
```

(end of run stream)

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

A. Main Program

OFFTAPE

B. Subroutines

USER2$DISK:[DAO.METTAPE]

VATAPE

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

3.13 PREPROCESSOR RWND_CAN_CLCT

I. PREPROCESSOR FUNCTION

This preprocessor extracts (windows) rawinsonde data from the Canadian upper-air data tape, and stores the windowed data on the VAX disk. Latitude and longitude control cards define the horizontal extent of the windowing, while the vertical extent is controlled by another control card. RWND_CAN_CLCT also calculates the dew points from the sounding data, which are reported as dry-bulb temperatures and relative humidities.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. Canadian Upper-Air Data File

A VAX system tape, created by the COPYTAPE preprocessor (see Section 3.9), contains the Canadian upper-air data (referenced as Tape 100640 in the run stream below).[20] These data contain a uniquely-Canadian station identification code, (which is converted to a pseudo-WBAN identifier using the next nonstandard input file), the date and time (as GMT) of the observation, the pressure level and its geopotential height, and meteorological parameters associated with that level -- temperature, relative humidity, wind direction, and wind speed. Both mandatory (including standard) and significant pressure levels are currently reported (see Section 3.3.1 for definitions). The data currently exist as two separate record types within a single file; the first 90 records contains the meteorological variables (listed as ELEM(1) to ELEM(6) in Table 3.37, Tape Parameters, below) for the mandatory and significant levels. The second record type contains wind data at levels where there are significant changes (ELEM(7) to ELEM(10)), and consists of 120 records. Each fixed-length record is read from the tape using the following READ and FORMAT statements:

```
      READ (UNITIN,200,END=888,ERR=850) BUFFER
200   FORMAT (A)
      READ (BUFFER(NBYTES+1:NBYTES+MAXRCL),250) STNID, YR, MON, DAY, HR,
     &     ELEM, (VALUE(J), F(J), J = 1,90), SRC
250   FORMAT (A7, I3, 3I2.2, I3, 90(F6.0,A1), 6X, A1)
```

b. WBAN Conversion File

This file contains a WBAN or pseudo-WBAN identifier for each Canadian upper-air station in the modeling domain. The pseudo-WBAN identifiers were assigned arbitrarily at Computer Sciences Corporation. The file's READ and FORMAT statements follow, and Table 3.38 shows its parameters.

```
        READ (UTABLE, 300, END = 350) CANID, WBAN, LAT, LON
300     FORMAT (1X, A7, 2X, I5.5, 2X, F5.2, 2X, F6.2)
```

20. COPYTAPE may not always be successfully executed on all Canadian upper-air data tapes. Our experience has shown that these data will sometimes be recorded as one file that spans two tapes, with no E-O-F marker on the first tape. If you receive data in this format, and you want to make a copy of the original tape, you must use the VAX system's COPY routine. Further information about this routine can be obtained from the VMS DCL Dictionary, publication number AA-LA12A-TE, available from Digital Equipment Corporation, Corporate User Publications, ZK01-3/J35, 110 Spit Brook Road, Nashua, NH 03062.

TABLE 3.37. TAPE PARAMETERS

Parm No.	Parm Name [a]	Unit [b]	Data Type	Description
1	STNID		Char*7	Station identifier
2	YR		Integer*4	Year as *YYY* (e.g., 988)
3	MON		Integer*4	Month as *MM*
4	DAY		Integer*4	Day as *DD*
5	HR	GMT	Integer*4	Hour as *HH*
6	$ELEM(1)_i$		Integer*4	Atmospheric pressure
7	$VALUE(1)_i$	0.01 kPa	Real*4	Value
8	$ELEM(2)_i$		Integer*4	Altitude above sea level
9	$VALUE(2)_i$	m	Real*4	Value
10	$ELEM(3)_i$		Integer*4	Temperature
11	$VALUE(3)_i$	0.1 °C	Real*4	Value
12	$ELEM(4)_i$		Integer*4	Relative humidity
13	$VALUE(4)_i$	%	Real*4	Value
14	$ELEM(5)_i$		Integer*4	Wind direction
15	$VALUE(5)_i$	deg	Real*4	Value
16	$ELEM(6)_i$		Integer*4	Wind speed
17	$VALUE(6)_i$	$m \cdot s^{-1}$	Real*4	Value
18	$ELEM(7)_i$		Integer*4	Pressure
19	$VALUE(7)_i$	0.01 kPa	Real*4	Value
20	$ELEM(8)_i$		Integer*4	Altitude above sea level
21	$VALUE(8)_i$	m	Real*4	Value
22	$ELEM(9)_i$		Integer*4	Wind direction
23	$VALUE(9)_i$	deg	Real*4	Value
24	$ELEM(10)_i$		Integer*4	Wind speed
25	$VALUE(10)_i$	$m \cdot s^{-1}$	Real*4	Value
26	F_i		Char*1	Flag (not used)
27	SRC_i		Char*1	Source code (not used)

a. ELEM(1) - ELEM(6) are parameters and data for mandatory and significant levels.
ELEM(7) - ELEM(10) are wind data at levels where there are significant changes.
i represents a discrete pressure level

b. 0.1 mb = 0.01 kPa

TABLE 3.38. WBAN CONVERSION FILE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CANID		Char*7	Station identifier
2	WBAN		Integer*4	Pseudo-WBAN number
3	LAT	° N	Real*4	Station latitude
4	LON	° W	Real*4	Station longitude

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

RWND_CAN_CLCT generates RWND_*YY*CAN_CLCT.DAT, which contains extracted data, dew points, and no flags. This file is produced by the following WRITE and FORMAT statements, and Table 3.39 shows its parameters.

```
          WRITE (UNITOUT,260) IYEAR, OLDMON, OLDDAY, OLDHR, WBAN,
     &          L, K, PR(L), HT(L), T(L), TD(L), RH(L), WDIR(L),
     &          WSPD(L), LAT, LON
260       FORMAT (4(1X,I2.2), 1X, I5.5, 2X, 2I2.2, 7F8.1, 2F7.2)
```

TABLE 3.39. RWND_*YY*CAN_CLCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IYEAR		Integer*4	Year as *YY*
2	OLDMON		Integer*4	Month as *MM*
3	OLDDAY		Integer*4	Day as *DD*
4	OLDHR	GMT	Integer*4	Hour as *HH*
5	WBAN		Integer*4	Station identifier
6	L		Integer*4	Level indicator
7	K		Integer*4	Number of levels
8	PR_L	mb	Real*4	Pressure
9	HT_L	m	Real*4	Geopotential height
10	T_L	°C	Real*4	Temperature
11	TD_L	°C	Real*4	Dew point
12	RH_L	%	Real*4	Relative humidity
13	$WDIR_L$	deg	Real*4	Wind direction
14	$WSPD_L$	$m{\cdot}s^{-1}$	Real*4	Wind speed
15	LAT	° N	Real*4	Station latitude
16	LON	° W	Real*4	Station longitude

Note: *L* = level number (one or more digits).

III. CONTROL CARDS

Four control cards are used; the second and third define the region boundaries. The cards are input in the following format:

IFILE ENDFILE
MINLAT MAXLAT MINLON MAXLON
PMIN
TOTBLK

The boundaries for your particular modeling domain (if it is not the ROMNET region) can be ascertained from Chapter 3 of Sellars *et al.* Table 3.40 defines the control card variables for the ROMNET region.

TABLE 3.40. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Data Type	Description
1	IFILE		Integer*4	Number of files to skip
2	ENDFILE		Integer*4	Number of the last rawinsonde file
3	MINLAT	° N	Real*4	Latitude of southern bound
4	MAXLAT	° N	Real*4	Latitude of northern bound
5	MINLON	° W	Real*4	Longitude of eastern bound
6	MAXLON	° W	Real*4	Longitude of western bound
7	PMIN	mb	Real*4	Minimum sounding pressure
8	TOTBLK		Integer*4	Maximum number of blocks to process

Example (for ROMNET):

01 02

25.0 55.0 60.0 105.0

200.

100000

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	4	files	41	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	4	files	24	blocks
Executable file:	1	file	19	blocks
	9	files	84	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:00:51
Buffered I/O count:	259
Direct I/O count:	493
Peak working-set size:	615
Peak virtual size:	2934

C. Space Requirements: Log and Print Files

RWND_CAN_CLCT.LOG:	9 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.41 shows the input file and output file space requirements for 1988 data. Note that the output storage requirement will vary.

TABLE 3.41. I/O FILE SPACE REQUIREMENTS FOR 1988 DATA

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	STNS_CAN_RWND.DAT	Nonstd	3	N/A
Output	RWND_88CAN_CLCT.DAT	Nonstd	5564	1 year

E. Space Requirements: Tape Files

Table 3.42 shows the input tape file space requirements.

TABLE 3.42. INPUT TAPE FILE SPACE REQUIREMENTS

File Group	Tape No.	File No.	Record Length (in bytes)	Block Size (in bytes)
Input	100640	2	1 = 656 2 = 866	19024

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_CAN_CLCT.LNK follows:

```
$ ASSIGN USER2$DISK:[DAO.ROMWORK] A
$ LINK A:RWND_CAN_CLCT, [ROMLIB.UTILIO]ADATE.OBJ,-
  [ROMLIB.UTILIO]JUNIT.OBJ, [DAO.METTAPE]VATAPE.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_CAN_CLCT.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MET27:[SCRATCH.DAO]RWND_88CAN_CLCT.DAT     OUTPUT
$ ASSIGN MET27:[SCRATCH.DAO]STNS_CAN_RWND.DAT       XTABLE
$ TAPE SELECT TAPEDRIVE 100640
$ TAPE LOAD TAPEDRIVE 100640
$ MOUNT/NOWRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=19024 -
  TAPEDRIVE 100640 SFCTAP:
$ SHOW DEV/FULL TAPEDRIVE
$ RUN USER2$DISK:[DAO.ROMWORK]RWND_CAN_CLCT.EXE
  01  02
  22.0  55.0  60.0  105.0
  200.0
  100000
$ SHOW DEV/FULL TAPEDRIVE
$ DISMOUNT SFCTAP:
$ DEALLOCATE TAPEDRIVE
```

(end of run stream)

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

A. Main Program

RWND_CAN_CLCT

B. Subroutines

VATAPE

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

B. Functions

JUNIT

IX. INCLUDE FILES

None

3.14 PREPROCESSOR RWND_END

I. PREPROCESSOR FUNCTION

This preprocessor modifies the upper-air data set as a result of your inspection of the plots from preprocessor RWND_V2. It differs from preprocessor DEL in that it modifies the data set one sounding at a time, while DEL can modify one or more levels within specific soundings. RWND_END has two modes of operation: (1) it deletes an entire sounding from the data set; (2) it replaces dubious wind data in a sounding with the "missing" flag -9999.0, while retaining all the thermodynamic variables.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. RWND_*YYUSMMM*_SORT1.DAT or RWND_*YYCANMMM*_SORT1.DAT

RWND_*YYUSMMM*_UPDT.DAT or RWND_*YYCANMMM*_UPDT.DAT are generated as corrected versions of the input data by preprocessor DEL. These files are then reformatted by RWND_RFMT, and sorted by the VAX SORT utility, resulting in a chronologically ordered data set. The contents of these files includes the station identification codes, the date and time (as GMT) of the observation, the pressure level and its geopotential height, and meteorological parameters associated with that level -- temperature, dew point, relative humidity, wind direction, and wind speed. Both mandatory (including standard) and significant pressure levels are currently reported (see Section 3.3.1 for definitions). The READ and FORMAT statements follow, and the file's parameters are shown in Table 3.43.

```
      READ (UNIT1, 1200, END = 999) YR,MON,DAY,HR,WBAN,LEV,NLEV,PR(I),
     & HT(I),T(I),TD(I),RH(I),WD(I),WS(I),LAT(I),LON(I)
 1200 FORMAT (4(1X,I2.2), 1X, I5.5, 2X, 2I2.2, 7F8.1, 2F7.2)
```

TABLE 3.43. RWND_*YY*US*MMM*_SORT1.DAT AND RWND_*YY*CAN*MMM*_SORT1.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	YR		Integer*4	Year as *YY*
2	MON		Integer*4	Month as *MM*
3	DAY		Integer*4	Day as *DD*
4	HR	GMT	Integer*4	Hour as *HH*
5	WBAN		Integer*4	Station identifier
6	LEV		Integer*4	Level indicator
7	NLEV		Integer*4	Number of levels
8	$PR_{m,L}$	mb	Real*4	Pressure at level *L*
9	$HT_{m,L}$	m	Real*4	Height at level *L*
10	$T_{m,L}$	°C	Real*4	Temperature at level *L*
11	$TD_{m,L}$	°C	Real*4	Dew point at level *L*
12	$RH_{m,L}$	%	Real*4	Relative humidity at level *L*
13	$WD_{m,L}$	deg	Real*4	Wind direction at level *L*
14	$WS_{m,L}$	$m \cdot s^{-1}$	Real*4	Wind speed at level *L*
15	LAT_m	° N	Real*4	Station latitude
16	LON_m	° W	Real*4	Station longitude

Note: *m* = upper-air station number (one or more digits); *L* = level number

b. *YYMMM*_RWND_DEL.DAT

To accomplish the removal of questionable data, you must prepare a data file containing the date, the time, and the station ID (the WBAN number) of each sounding to be modified. In addition, a one-character flag must be included on each line indicating whether the entire sounding is to be deleted (flag = A), or whether only the winds are to be reported as "missing" (flag = W). This file must be in chronological and numerical order for RWND_END to execute properly. The READ and FORMAT statements follow, and the file's parameters are shown in Table 3.44.

```
     READ (UNIT2, 1100, END = 100) CDATE, CWBAN, FLAG
1100 FORMAT (I8, 2X, I5, 2X, A1)
```

TABLE 3.44. *YYMMM*_RWND_DEL.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CDATE	GMT	Integer*4	Date and time as *YYMMDDHH*
2	CWBAN		Integer*4	Station identifier
3	$FLAG_m$		Char*1	Flag: W reports winds as "missing" A deletes the entire sounding

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

RWND_*YYUSMMM*_VRFY.DAT or RWND_*YYCANMMM*_VRFY.DAT are generated as modified versions of the input data. These files contain parameters identical to those of the input files above, and are produced by the following WRITE and FORMAT statements:

```
      WRITE (UNITOUT, 1200) YR,MON,DAY,HR,WBAN,I,NLEV,
     & PR(I),HT(I),T(I),TD(I),RH(I),WD(I),WS(I),
     & LAT(I),LON(I)
 1200 FORMAT (4(1X,I2.2), 1X, I5.5, 2X, 2I2.2, 7F8.1, 2F7.2)
```

III. CONTROL CARDS

One control card is used, in the format below; the control card variable is shown in Table 3.45.

KCAN

TABLE 3.45. CONTROL CARD VARIABLE

Parm No.	Parm Name	Unit	Data Type	Description
1	KCAN		Integer*4	Indicator for U.S. and Canadian data: 0 ⇒ one country's data only 1 ⇒ both countries' data

Example:

0

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION, U.S. DATA

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	9	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	8	blocks
Executable file:	1	file	10	blocks
	3	files	27	blocks

B. Execution Time Requirements (Representative Values for a One-Month Data Span):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:04:32
Buffered I/O count:	442
Direct I/O count:	2565
Peak working-set size:	592
Peak virtual size:	4286

C. Space Requirements: Log and Print Files

RWND_END.LOG:	3 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.46 shows the input file and output file space requirements for May 1988 data. Note that both the input and output storage requirements will vary from month to month.

TABLE 3.46. I/O FILE SPACE REQUIREMENTS FOR MAY 1988 U.S. DATA

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	RWND_88USMAY_SORT1.DAT	Nonstd	17420	1 month
	88MAY_RWND_DEL.DAT	Nonstd	1	
			17421	
Output	RWND_88USMAY_VRFY.DAT	Nonstd	17419	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_END.LNK follows:

```
$ ASSIGN USER2$DISK:[RTT.ROM] A
$ LINK A:RWND_END, [ROMLIB.UTILIO]JUNIT.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_END.COM for U.S. data follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MET25:[RTT]RWND_US88MAY_SORT1.DAT          INPUT1
$ ASSIGN USER2$DISK:[RTT.ROM]MAY88_RWND_DEL.DAT     INPUT2
$ ASSIGN MET25:[RTT]RWND_US88MAY_VRFY.DAT           OUTPUT
$ RUN USER2$DISK:[RTT.ROM]RWND_END.EXE
0
```

(end of run stream)

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

A. Main Program

RWND_END

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.15 PREPROCESSOR RWND_RFMT

I. PREPROCESSOR FUNCTION

This preprocessor interpolates wind speed, wind direction, and thermodynamic variables from existing upper-air data to replace missing data. It uses a simplified Ekman spiral to replace missing surface winds; other missing winds are linearly interpolated, as are missing thermodynamic variables. RWND_RFMT also discards inappropriately-reported data from mandatory levels that lie beneath the terrain surface, as well as discarding an entire sounding if the surface temperature is missing.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

RWND_*YY*US*MMM*_WNDW.DAT and RWND_*YY*CAN*MMM*_WNDW.DAT are produced by preprocessor RWND_WNDW and contain the parameters shown in Table 3.47. The READ and FORMAT statements for either input file are listed below.

```
    READ (UNITIN,180,END=999) DATIME(1), STNID(1), LEV, NLEV, PR(1),
   & HT(1), T(1), TD(1), RH(1), WD(1), WS(1), LAT(1), LON(1)
180 FORMAT (A12, 1X, I5, 2X, 2I2, 7F8.1, 2F7.2)
```

TABLE 3.47. METCOND.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATIME	GMT	Char*12	Date and time as *YYMMDDHH*
2	STNID		Integer*4	WBAN station identifier
3	LEV_m		Integer*4	Level indicator
4	$NLEV_m$		Integer*4	Number of levels
5	$PR_{m,L}$	mb	Real*4	Pressure
6	$HT_{m,L}$	m	Real*4	Height
7	$T_{m,L}$	°C	Real*4	Temperature
8	$TD_{m,L}$	°C	Real*4	Dew point
9	$RH_{m,L}$	%	Real*4	Relative humidity
10	$WD_{m,L}$	°	Real*4	Wind direction
11	$WS_{m,L}$	$m \cdot s^{-1}$	Real*4	Wind speed
12	LAT_m	° N	Real*4	Latitude
13	LON_m	° W	Real*4	Longitude

Note: m = upper-air station number; L = level.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

RWND_*YYUSMMM*_RFMT.DAT is identical to the input file, except that some missing data may have been interpolated and added, and/or may have been deleted. The WRITE and FORMAT statements are listed below.

```
      WRITE (UNITOUT,800) DATIME(L), STNID(L), LEV, NLEV,
    & PR(L), HT(L), T(L), TD(L), RH(L), WD(L), WS(L),
    & LAT(L), LON(L)
800   FORMAT (A12, 1X, I5.5, 2X, 2I2.2, 7F8.1, 2F7.2)
```

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	3	files	45	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	3	files	27	blocks
Executable file:	1	file	19	blocks
	7	files	91	blocks

B. Execution Time Requirements (Representative Values for a One-Month Data Span):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:03:28
Buffered I/O count:	330
Direct I/O count:	2485
Peak working-set size:	545
Peak virtual size:	924

C. Space Requirements: Log and Print Files

RWND_RFMT.LOG	15 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.48 shows the input file and output file space requirements.

TABLE 3.48. SAMPLE I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	RWND_88USSEP_WNDW.DAT	Nonstd	±5500	1 month
Output	RWND_88USSEP_RFMT.DAT	Nonstd	±5500	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_RFMT.LNK follows:

```
$ ASSIGN USER2$DISK:[RTT.ROM] A
$ LINK A:RWND_RFMT, [ROMLIB.UTILIO]ADATE.OBJ,-
  [ROMLIB.UTILIO]JUNIT.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_RFMT.COM follows:

```
$ ! ACCOUNTING CODE
$ ! RUN STREAM IS IN THE FILE USER2$DISK:[RTT.ROM]RWND_RFMT.RUN
$ SET VERIFY
$ ASSIGN MET25:[RTT]RWND_US88APR_WNDW.DAT    INPUT
$ ASSIGN MET25:[RTT]RWND_US88APR_RFMT.DAT    OUTPUT
$ RUN USER2$DISK:[RTT.ROM]RWND_RFMT.EXE
```

(end of run stream)

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

A. Main Program

RWND_RFMT

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

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.16 PREPROCESSOR RWND_US_CLCT

I. PREPROCESSOR FUNCTION

This preprocessor reads rawinsonde data from the U.S. upper-air data tape (TD6200 from the NCDC), and extracts only those stations with a WBAN identification number, storing these data on the VAX disk. RWND_US_CLCT thus excludes soundings from unorthodox sites such as ships or other mobile platforms. Latitude and longitude control cards define the horizontal extent of the windowing you require, while the vertical extent is controlled by another control card. The preprocessor also calculates the dew points from the sounding data, which are reported as dry-bulb temperatures and relative humidities.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

U.S. Upper-Air Data File

A VAX system tape, created by the COPYTAPE preprocessor (see Section 3.9), contains the U.S. upper-air data (referenced as Tape 100326 in the run stream below).[21] These data consist of the station identifier, the date and time (as GMT) of the observation, the pressure level and its geopotential height, and meteorological parameters associated with that level -- temperature, relative humidity, wind direction, and wind speed. Both mandatory (including standard) and significant pressure levels are reported (see Section 3.3.1 for definitions). Each fixed-length record is read from the tape using the following READ and FORMAT statements:

21. COPYTAPE may not always be successfully executed on all U.S. upper-air data tapes. Our experience has shown that these data will sometimes be recorded as one file that spans two tapes, with no E-O-F marker on the first tape. If you receive data in this format, and you want to make a copy of the original tape, you must use the VAX system's COPY routine. Further information about this routine can be obtained from the VMS DCL Dictionary, publication number AA-LA12A-TE, available from Digital Equipment Corporation, Corporate User Publications, ZK01-3/J35, 110 Spit Brook Road, Nashua, NH 03062.

```
     READ (UNITIN,300,END=999,IOSTAT=IOST,ERROR=800) BUFFER
 300 FORMAT (A)
     READ (BUFFER(NBEG+4:NBEG+NBYTES-1), 400, ERR = 750) PREFIX, WBAN,
    & LATDEG, LATMIN, LATA, LONDEG, LONMIN, LONA, YEAR, MONTH,
    & DAY, HOUR, NUMLEV, (QIND(J), ETIME(J), PRESS(J), HGT(J),
    & TEMP(J), RH(J), WD(J), WS(J), TIMEF(J), PRESSF(J), HGTF(J),
    & TEMPF(J), RHF(J), WINDF(J), TYPLEV(J), J = 1,NUMLEV)
 400 FORMAT (A3, A5, 2F2.0, A1, F3.0, F2.0, A1, I4, 3(I2), I3, 200(A1,
    & F4.1, F5.2, F6.0, F4.1, F3.0, F3.0, F3.0, 7A1))
```

Table 3.49 shows the tape parameters.

TABLE 3.49. TAPE PARAMETERS

Parm No.	Parm Name [a]	Unit [b]	Data Type	Description
1	PREFIX		Char*3	Station index
2	WBAN		Char*5	Station identifier
3	LATDEG	deg	Real*4	Latitude (degrees)
4	LATMIN	min	Real*4	Latitude (minutes)
5	LATA		Char*1	Latitude code (N or S)
6	LONDEG	deg	Real*4	Longitude (degrees)
7	LONMIN	min	Real*4	Longitude (minutes)
8	LONA		Char*1	Longitude code (W or E)
9	YEAR		Integer*4	Year as *YYYY*
10	MONTH		Integer*4	Month as *MM*
11	DAY		Integer*4	Day as *DD*
12	HOUR	GMT	Integer*4	Hour as *HH*
13	NUMLEV		Integer*4	Number of levels
14	QIND		Char*1	Level indicator
15	$ETIME_i$	min	Real*4	Elapsed time since release
16	$PRESS_i$	kPa	Real*4	Atmospheric pressure
17	HGT_i	m	Real*4	Altitude above sea level
18	$TEMP_i$	°C	Real*4	Temperature
19	RH_i	%	Real*4	Relative humidity
20	WD_i	deg	Real*4	Wind direction
21	WS_i	$m{\cdot}s^{-1}$	Real*4	Wind speed
22	$TIMEF_i$		Char*1	Flag (not used)
23	$PRESSF_i$		Char*1	Flag (not used)
24	$HGTF_i$		Char*1	Flag (not used)
25	$TEMPF_i$		Char*1	Flag (not used)
26	RHF_i		Char*1	Flag (not used)
27	$WINDF_i$		Char*1	Flag (not used)
28	$TYPELEV_i$		Char*1	Flag (not used)

a. *i* represents a discrete pressure level
b. 1 kPa = 10 mb = 1 centibar

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

a. RWND_*YY*US_CLCT.DAT

This file contains the extracted data from the tape, the calculated dew points, and no flags. The file is produced by the following WRITE and FORMAT statements, and Table 3.50 shows its parameters.

```
      WRITE (UNITOUT,600) IYEAR, MONTH, DAY, HOUR, WBAN, L, K,
     & PRESS(L), HGT(L), TEMP(L), TD(L), RH(L), WD(L), WS(L),
     & LAT, LON
  600 FORMAT (4(1X, I2.2), 1X, A5, 2X, 2I2.2, 7F8.1, 2F7.2)
```

TABLE 3.50. RWND_*YY*US_CLCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IYEAR		Integer*4	Year as *YY*
2	MONTH		Integer*4	Month as *MM*
3	DAY		Integer*4	Day as *DD*
4	HOUR	GMT	Integer*4	Hour as *HH*
5	WBAN		Char*5	Station identifier
6	L		Integer*4	Level indicator
7	K		Integer*4	Number of levels
8	$PRESS_L$	mb	Real*4	Pressure
9	HGT_L	m	Real*4	Geopotential height
10	$TEMP_L$	°C	Real*4	Temperature
11	TD_L	°C	Real*4	Dew point
12	RH_L	%	Real*4	Relative humidity
13	WD_L	deg	Real*4	Wind direction
14	WS_L	$m \cdot s^{-1}$	Real*4	Wind speed
15	LAT	° N	Real*4	Station latitude
16	LON	° W	Real*4	Station longitude

Note: L = level number (one or more digits).

b. ERROR_YYUS.DAT

This file contains the location of errors encountered during the data extraction process. We generally do not encounter such errors, however. Such errors are recorded by the IERR parameter, which is reported in the .LOG file. You can also check whether this file's size equals zero; if it does not, errors have been found. The file is produced by the following WRITE statement, and Table 3.51 shows its parameters.

```
WRITE (UNITERR,*) RECORD, WBAN, NBEG, IERR, IBLOCK, IFILE
```

TABLE 3.51. ERROR_YYUS.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	RECORD		Char*2672	Record string
2	WBAN		Char*5	Station identifier
3	NBEG	bytes	Integer*4	Size of record
4	IERR		Integer*4	Error count
5	IBLOCK		Integer*4	Block number of error
6	IFILE		Integer*4	File number of error

III. CONTROL CARDS

Two control cards input the region boundaries; they are used in the following format:

MINLAT MAXLAT MINLON MAXLON
PMIN

The boundaries for your particular modeling domain (if it is not the ROMNET region) can be ascertained from Chapter 3 of Sellars *et al.* Table 3.52 defines the control card variables for the ROMNET region.

TABLE 3.52. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Data Type	Description
1	MINLAT	° N	Real*4	Latitude of southern bound
2	MAXLAT	° N	Real*4	Latitude of northern bound
3	MINLON	° W	Real*4	Longitude of eastern bound
4	MAXLON	° W	Real*4	Longitude of western bound
5	PMIN	mb	Real*4	Minimum sounding pressure

Example (for ROMNET):

25.0 55.0 60.0 105.0

200.

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	3	files	29	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	3	files	20	blocks
Executable file:	1	file	15	blocks
	7	files	64	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	01:05:43
Buffered I/O count:	1725
Direct I/O count:	23743
Peak working-set size:	508
Peak virtual size:	1450

C. Space Requirements: Log and Print Files

RWND_US_CLCT.LOG:	14 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.53 shows the input file and output file space requirements for 1983 data. Note that the output storage requirement varies year by year, but will require storage of the magnitude shown.

TABLE 3.53. I/O FILE SPACE REQUIREMENTS FOR 1983 DATA

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input			0	
Output	ERROR_83US.DAT	Nonstd	0	1 year
	RWND_83US_CLCT.DAT	Nonstd	260000	1 year
			260000	

E. Space Requirements: Tape Files

Table 3.54 shows the input tape file space requirements.

TABLE 3.54. INPUT TAPE FILE SPACE REQUIREMENTS

File Group	Tape No.	File No.	Record Length (in bytes)	Block Size (in bytes)
Input	100326	2	7236	12000

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_US_CLCT.LNK follows:

```
$ ASSIGN USER2$DISK:[RTT.ROM] A
$ LINK A:RWND_US_CLCT, [ROMLIB.UTILIO]ADATE.OBJ,-
  [ROMLIB.UTILIO]JUNIT.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_US_CLCT.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ !
$ ! RUN STREAM IS IN THE FILE USER2$DISK:[RTT.ROM]RWND_US_CLCT.RUN
$ !
$ ASSIGN MET25:[RTT]RWND_83US_CLCT.DAT  OUTPUT
$ ASSIGN MET25:[RTT]ERROR83US.DAT       ERRCHK
$ ASSIGN MLAB: SYS$PRINT
$ @TINUSE
$ TAPE SELECT TAPEDRIVE 100326
$ SHOW TIME
$ TAPE LOAD TAPEDRIVE 100326
$ SHOW TIME
$ MOUNT/NOWRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=12000 -
  TAPEDRIVE 100326 R1TAP:
$ SHOW DEV/FULL TAPEDRIVE
$ SHOW TIME
$ SET MAGTAPE/SKIP=FILES:1 R1TAP:
$ SHOW TIME
$ RUN USER2$DISK:[RTT.ROM]RWND_US_CLCT.EXE
  25.0  55.0  60.0  105.0
 200.0
$ SHOW DEV/FULL TAPEDRIVE
$ @REWIND R1TAP:
$ DISMOUNT R1TAP:
$ DEALLOCATE TAPEDRIVE
```

(end of run stream)

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

A. Main Program

RWND_US_CLCT

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

B. Functions

JUNIT

IX. INCLUDE FILES

None

3.17 PREPROCESSOR RWND_WNDW

I. PREPROCESSOR FUNCTION

This preprocessor extracts (windows) by month the rawinsonde data from the U.S. upper-air data file RWND_YYUS_CLCT.DAT or RWND_YYCAN_CLCT.DAT, the Canadian upper-air data file. RWND_WNDW ignores soundings that are not at standard times (i.e., at 00Z, 06Z, 12Z, or 18Z); however, it accepts soundings that are within one hour of a standard time. When the program accepts such a sounding, it modifies the actual release time of the rawinsonde to that of the standard time. This section is written for the windowing of U.S. data; the requirements for windowing Canadian data are identical; however, file sizes will be smaller.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

One nonstandard file, RWND_*YY*US_CLCT.DAT for U.S. data[22], is the input to the preprocessor. The file contains station identification codes, the date and time (as GMT) of the observation, the pressure level and its geopotential height, and meteorological parameters associated with that level -- temperature, dew point, relative humidity, wind direction, and wind speed. Both mandatory (including standard) and significant pressure levels are currently reported (see Section 3.3.1 for definitions). The READ and FORMAT statements follow, and the file's parameters are shown in Table 3.55.

```
      READ (UNITIN, 150) YR(1), MON(1), DAY(1), HR(1), WBAN(1), LEV(1),
     &                  NLEV(1), ADATA(1)
  150 FORMAT (4(1X,I2.2), 1X, A5, 2X, 2I2.2, A70)
```

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

RWND_WNDW generates RWND_*YY*US*MMM*_WNDW.DAT. This file contains parameters identical to those of the input file, but includes only a single month's data. This file is produced by the following WRITE and FORMAT statements:

22. The Canadian input file RWND_*YY*CAN_CLCT.DAT contains the same data.

```
WRITE (UNITOUT, 150) YR(1), MON(1), DAY(1), HR(1), WBAN(1), LEV(1),
&                    NLEV(1), ADATA(1)
150 FORMAT (4(1X,I2.2), 1X, A5, 2X, 2I2.2, A70)
```

TABLE 3.55. RWND_YYUS_CLCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	YR		Integer*4	Year as *YY*
2	MON		Integer*4	Month as *MM*
3	DAY		Integer*4	Day as *DD*
4	HR	GMT	Integer*4	Hour as *HH*
5	WBAN		Char*5	Station identifier
6	LEV		Integer*4	Level indicator
7	NLEV		Integer*4	Number of levels
8	$ADATA_L$		Char*70	Data stream
	$PRESS_L$	mb	Real*4	Pressure
	HGT_L	m	Real*4	Geopotential height
	$TEMP_L$	°C	Real*4	Temperature
	TD_L	°C	Real*4	Dew point
	RH_L	%	Real*4	Relative humidity
	WD_L	deg	Real*4	Wind direction
	WS_L	$m{\cdot}s^{-1}$	Real*4	Wind speed
	LAT	° N	Real*4	Station latitude
	LON	° W	Real*4	Station longitude

Note: L = level number (one or more digits).

III. CONTROL CARDS

One control card is used, in the format shown below. It selects the month for windowing. Table 3.56 describes the control card variable.

TARMON

TABLE 3.56. CONTROL CARD VARIABLE

Parm No.	Parm Name	Unit	Data Type	Description
1	TARMON		Integer*4	The number of the month for windowing

Example:

09

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION, U.S. DATA

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	5	files	34	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	5	files	20	blocks
Executable file:	1	file	13	blocks
	11	file	67	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:04:20
Buffered I/O count:	313
Direct I/O count:	5286
Peak working-set size:	529
Peak virtual size:	927

C. Space Requirements: Log and Print Files

RWND_WNDW.LOG:	4 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.57 shows the input file and output file space requirements for September 1988 data. Note that the output storage requirement will vary.

TABLE 3.57. I/O FILE SPACE REQUIREMENTS FOR SEPTEMBER 1988 U.S. DATA

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	RWND_88US_CLCT.DAT	Nonstd	260000	1 year
Output	RWND_88USSEP_WNDW.DAT	Nonstd	5500	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_WNDW.LNK follows:

```
$ ASSIGN USER2$DISK:[RTT.ROM] A
$ LINK A:RWND_WNDW, [ROMLIB.UTILIO]ADATE.OBJ,-
  [ROMLIB.UTILIO]JUNIT.OBJ, [ROMLIB.UTILIO]GREG.OBJ,-
  [ROMLIB.UTILIO]JULIAN.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_WNDW.COM for U.S. data follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ !
$ ! RUN STREAM IS IN THE FILE USER2$DISK:[RTT.ROM]RWND_WNDW.RUN
$ !
$ ASSIGN MET27:[SCRATCH.DAO]RWND_88US_CLCT.DAT    INPUT
$ ASSIGN MET25:[RTT]RWND_88USSEP_WNDW.DAT         OUTPUT
$ !
$ ! ASSIGN MET27:[SCRATCH.DAO]RWND_88CAN_CLCT.DAT    INPUT
$ ! ASSIGN MET25:[RTT]RWND_88CANSEP_WNDW.DAT         OUTPUT
$ !
$ RUN USER2$DISK:[RTT.ROM]RWND_WNDW.EXE
  09
```

(end of run stream)

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

A. Main Program

RWND_WNDW

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
GREG

B. Functions

JUNIT
JULIAN

IX. INCLUDE FILES

None

3.18 PREPROCESSOR RWND_V2

I. PREPROCESSOR FUNCTION

This preprocessor facilitates the validation of upper-air data by plotting (for human interpretation) skew-T/log-P thermodynamic diagrams and/or 3-D hodographs of questionable rawinsonde soundings. The data can thus be checked for inappropriate conditions such as superadiabatic lapse rates, excessively large vertical wind shear, unnaturally deep inversion layers, etc.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

RWND_*YYUSMMM*_WNDW.DAT and RWND_*YYCANMMM*_WNDW.DAT are produced by preprocessor RWND_WNDW, checked by RWND_RFMT, and are then sorted by the VAX SORT utility. The files that are input to RWND_V2 are RWND_*YYUSMMM*_SORT.DAT and RWND_*YYCANMMM*_SORT.DAT; these files contain the parameters shown in Table 3.58. The READ and FORMAT statements for either input file are listed below.

```
      READ(8,201,END=6)IY,IM,ID,IH,ISTA,I,N,P(I),H(I),T(I),TD(I),RH(I),
     $ WD(I),WS(I)
  201 FORMAT(4I3,I6,I4,I2,7F8.0)
```

TABLE 3.58. SORTED UPPER-AIR DATA FILES' PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IY		Integer*4	Year as *YY*
2	IM		Integer*4	Month as *MM*
3	ID		Integer*4	Day as *DD*
4	IH	GMT	Integer*4	Hour as *HH*
5	ISTA		Integer*4	WBAN station identifier
6	I_m		Integer*4	Level indicator
7	N_m		Integer*4	Number of levels
8	$P_{m,I}$	mb	Real*4	Pressure
9	$H_{m,I}$	m	Real*4	Height
10	$T_{m,I}$	°C	Real*4	Temperature
11	$TD_{m,I}$	°C	Real*4	Dew point
12	$RH_{m,I}$	%	Real*4	Relative humidity
13	$WD_{m,I}$	°	Real*4	Wind direction
14	$WS_{m,I}$	$m \cdot s^{-1}$	Real*4	Wind speed

Note: m = upper-air station number; I = level.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

RWND_V2 produces a plot file each time it is run (e.g., R88_04V2.OUT in the run stream that follows).

III. CONTROL CARDS

Three control cards are used, in the format shown below. Table 3.59 describes the control card variables.

TPLAP, ISDEX, KSTA, KCAN
NSTRT NSTOP
MODE

TABLE 3.59. CONTROL CARD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	TPLAP	°C·m-1	Real*4	Lapse rate criterion
2	ISDEX		Integer*4	Station flag: = 0 ⇒ check all > 0 ⇒ check only one
3	KSTA		Integer*4	WBAN station identifier [a]
4	KCAN		Integer*4	Canadian data flag: = 0 ⇒ exclude > 0 ⇒ include
5	NSTRT	GMT	Integer*4	Start date/time as *YYMMDDHH*
6	NSTOP	GMT	Integer*4	End date/time as *YYMMDDHH*
7	MODE		Char*4	Program operation mode [b]

a. Even if ISDEX = 0, a value is required here.
b. MANUAL displays each output plots on your terminal, and lets you decide whether or not to print the plot; AUTO saves all plots to the output file.

Example

-0.015,0,03937,0
88042200 88042423
AUTO

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	8	files	86	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	8	files	59	blocks
Executable file:	1	file	140	blocks
	17	files	285	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:02:25
Buffered I/O count:	225
Direct I/O count:	1293
Peak working-set size:	693
Peak virtual size:	3000

C. Space Requirements: Log and Print Files

RWND_V2.LOG	Variable (±15 blocks)
Print Files:	Variable (±2300 blocks)

D. Space Requirements: Input and Output Files

Table 3.60 shows the input file and output file space requirements.

TABLE 3.60. SAMPLE I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	RWND_88USSEP_SORT.DAT	Nonstd	±5500	1 month
Output	R88_04V2.OUT	Nonstd	±2300	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_V2.LNK follows:

```
$ LINK USER2$DISK:[RTT.ROM]RWND_V2.OBJ, -
  USER2$DISK:[RTT.ROM]MAKOPY.OBJ, -
  USER2$DISK:[RTT.ROM]QARAOB2.OBJ, -
  USER2$DISK:[RTT.ROM]SKEWT3.OBJ, -
  USER2$DISK:[RTT.ROM]HDGRAF2.OBJ, -
  USER2$DISK:[RTT.ROM]LISTNC.OBJ, -
  USER2$DISK:[RTT.ROM]HODO3D.OBJ, -
  USER2$DISK:[RTT.ROM]NUMTXT.OBJ, -
  USER2$DISK:[BRO.RUSS3D]OBJLIB.OLB, -
  USER2$DISK:[UGL]UGT.OLB
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

RWND_V2.COM follows:

```
$ ! ACCOUNTING CODE
$ ! RUN STREAM IS IN THE FILE USER2$DISK:[RTT.ROM]RWND_V2.RUN
$ SAVE_VERIFY = F$VERIFY("YES")
$ ASSIGN MET25:[RTT]RWND_US88APR_SORT.DAT INPUT1
$ DEFINE/USER_MODE SYS$OUTPUT MET25:[RTT]R88_04V2.OUT
$ RUN USER2$DISK:[RTT.ROM]RWND_V2
-0.015,0,03937,0
88042200 88042423
AUTO
$ IF SAVE_VERIFY THEN SET VERIFY
$ IF .NOT. SAVE_VERIFY THEN SET NOVERIFY
$ EXIT
```

(end of run stream)

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

A. Main Program

RWND_V2

B. Subroutines

MAKOPY
QARAOB2
SKEWT3
HDGRAF2
LISTNC
HODO3D
NUMTXT

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

USER2$DISK:[BRO.RUSS3D]

OBJLIB.OLB

USER2$DISK:[UGL]

UGT.OLB

Note that this subroutine is part of a modified Tektronix PLOT-10/Advanced Graphing II package, which must be purchased by you directly from a Tektronix dealer before we are able to supply the modifications.

B. Functions

None

IX. INCLUDE FILES

None

3.19 PREPROCESSOR SFCCAN_CLCT

I. PREPROCESSOR FUNCTION

This preprocessor extracts the surface meteorological data from the TDF-1440 format tapes acquired from the Canadian Climate Centre (CCC). SFCCAN_CLCT (1) converts some units to conform with SURMET1-type units, and (2) adjusts cloud data to conform with the SURMET1 format.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

Surface meteorology data tapes from the CCC supply the data for SFCCAN_CLCT. Two tapes contain each year's data. As a precaution against losing valuable data, a system tape copy is made for each original tape via the COPYTAPE preprocessor, and all processing proceeds on the copies. The READ and FORMAT statements for the tapes are listed below, and the parameters that are on the tapes are shown in Table 3.61.

```
130 READ (UNITIN,.150,END=900) ABLOCK
150 FORMAT (A)

    READ (BUFFER(M),170) STNID,IYR,IMON,IDAY,(DATA(I),I=1,6)
170 FORMAT (4X,A5,3I2.2,6A80)

    READ (DATA(I),190) IHR,ADIR,AWS,AT,ATD,ASLP,ASTNPR,
   &                   (ACOD(J),J=1,4),ATCOV,AOPCOV,AAMT(1),
   &                   ATYP(1),ABASE(1),AAMT(2),ATYP(2),ABASE(2),
   &                   AAMT(3),ATYP(3),ABASE(3),AAMT(4),ATYP(4),
   &                   ABASE(4),AWD,ADUM
190  FORMAT (I2.2,8X,A2,A3,A3,3X,A3,4X,A5,A4,1X,8A1,A3,
   &         2A1,A3,1X,2A1,A3,1X,2A1,A3,8X,A2,A4)
```

Note that as indicated above, the Canadian data are read in three stages. First, an entire block of data is read into ABLOCK. Individual records of data are found in BUFFER(*M*), with each record containing six hours of data, located in DATA(*I*). Our experience with these data indicates that while the record length of 495 bytes remains constant, the block size can change depending on how many records are stored in a single block. To run this preprocessor correctly, you must know the block size before executing the routine and, if necessary, make the appropriate changes to the code.

TABLE 3.61. TAPE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ABLOCK		Char*(MAXBLK)	Variable holding the data for an entire tape block of length MAXBLK
2	BUFFER		Char*(MAXREC)	Variable array equivalenced to ABLOCK, length = MAXREC, dimension = MAXBLK/MAXREC
3	STNID		Char*5	Canadian station identifier
4	IYR		Integer*4	Year (*YY*)
5	IMON		Integer*4	Month (*MM*)
6	IDAY		Integer*4	Day (*DD*)
7	DATA		Char*80	Data string comprising:
	$IHR_{n,h}$	LST	Integer*4	Hour as *HH*
	$ADIR_{n,h}$		Char*2	Wind direction (16-pt WBAN code)
	$AWS_{n,h}$	kt	Char*3	Wind speed
	$AT_{n,h}$	°F	Char*3	Temperature
	$ATD_{n,h}$	°F	Char*3	Dew point
	$ASLP_{n,h}$	mb	Char*5	Sea level pressure
	$ASTNPR_{n,h}$	in. Hg	Char*4	Station pressure
	$ACOD(J)_{n,h}$	1/10	Char*1	Cloud layer *J* sky condition, $J = 1, 4$
	$ATCOV_{n,h}$	1/10	Char*1	Total sky cover
	$AOPCOV_{n,h}$	1/10	Char*1	Total opaque sky cover
	$AAMT(1)_{n,h}$	1/10	Char*1	Amount of cloud layer 1
	$ATYP(1)_{n,h}$		Char*1	Type of cloud layer 1
	$ABASE(1)_{n,h}$	10^2 ft	Char*3	Height of cloud layer 1
	$AAMT(2)_{n,h}$	1/10	Char*1	Amount of cloud layer 2
	$ATYP(2)_{n,h}$		Char*1	Type of cloud layer 2
	$ABASE(2)_{n,h}$	10^2 ft	Char*3	Height of cloud layer 2
	$AAMT(3)_{n,h}$	1/10	Char*1	Amount of cloud layer 3
	$ATYP(3)_{n,h}$		Char*1	Type of cloud layer 3
	$ABASE(3)_{n,h}$	10^2 ft	Char*3	Height of cloud layer 3
	$AAMT(4)_{n,h}$	1/10	Char*1	Amount of cloud layer 4
	$ATYP(4)_{n,h}$		Char*1	Type of cloud layer 4
	$ABASE(4)_{n,h}$	10^2 ft	Char*3	Height of cloud layer 4
	$AWD_{n,h}$	10 deg	Char*2	Wind direction
	$ADUM_{n,h}$		Char*4	Dummy variable

Note: *n* = surface station number (one or more digits); *h* = hour; for sky cover, - represents overcast.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

SFCCAN_CLCT outputs one nonstandard file for each of the two input tapes, SFCCAN_MMM_1MMM_2YYx_CLCT.DAT, where x ranges from A to B. The output files are used by preprocessor SFCCAN_WNDW. The file is written in two steps: (1) a header step, and (2) a data step. The first two WRITE statements below form the header step, while the third is the data step. The file parameters are shown in Table 3.62.

```
      WRITE (UNITOUT, 370) IYR, IMON, IDAY, IHR, STNID, OPCOV,
     &                     TCOV, AMT1, BASE1, AMT2, BASE2, AMT3,
     &                     BASE3, SLP, WD, WS, T, TD, STNPR
  370 FORMAT (4I2.2, 1X, A5, 1X, 2F4.0, 3(1X, F4.0, F6.0), 1X,
     &        5F7.1, 1X, F7.1)
```

TABLE 3.62. SFCCAN_MMM_1MMM_2YYx_CLCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IYR		Integer*4	Year as *YY*
2	IMON		Integer*4	Month as *MM*
3	IDAY		Integer*4	Day as *DD*
4	IHR	LST	Integer*4	Hour as *HH*
5	$STNID_n$		Char*5	Canadian station identifier
6	$OPCOV_{n,h}$	%	Real*4	Opaque sky cover
7	$TCOV_{n,h}$	%	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
8	$AMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
9	$BASE1_{n,h}$	m	Real*4	Lowest cloud base
10	$AMT2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
11	$BASE2_{n,h}$	m	Real*4	Second-lowest cloud base
12	$AMT3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
13	$BASE3_{n,h}$	m	Real*4	Third-lowest cloud base
14	$SLP_{n,h}$	mb	Real*4	Sea level pressure
15	$WD_{n,h}$	deg	Real*4	Wind direction
16	$WS_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
17	$T_{n,h}$	°C	Real*4	Temperature
18	$TD_{n,h}$	°C	Real*4	Dew point
19	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

III. CONTROL CARDS

Two control cards are used in the format shown below. Table 3.63 defines the control card variables.

IFILE, ENDFILE
BEGMON, ENDMON, TARYR

TABLE 3.63. CONTROL CARD VARIABLES

Variable Name	Unit	Description
IFILE		Number of files to skip to reach data
ENDFILE		Number of the last data file to process
BEGMON		Number of the first month to be extracted
ENDMON		Number of the last month to be extracted
TARYR		Year of the data to be extracted

Example:

01, 02
04, 10, 1988

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	42	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	21	blocks
Executable file:	<u>1</u>	file	<u>24</u>	blocks
	3	files	87	blocks

B. Execution Time Requirements (per tape):

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:35:00
Buffered I/O count:	2037
Direct I/O count:	16625
Peak working-set size:	597
Peak virtual size:	3660

C. Space Requirements: Log and Print Files

SFCCAN_CLCT.LOG	24 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.64 shows the approximate output file space requirements.

TABLE 3.64. OUTPUT FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Output	SFCCAN_*MMM*$_1$*MMM*$_2$*YY*A_CLCT.DAT	Nonstd	80000	4 months
	SFCCAN_*MMM*$_1$*MMM*$_2$*YY*B_CLCT.DAT	Nonstd	40000	4 months
			120000	

E. Space Requirements: Tape Files

Two tapes per year contain the Canadian surface meteorology data. Table 3.65 shows the space requirements for a representative tape.

TABLE 3.65. INPUT TAPE FILE SPACE REQUIREMENTS

File Group	Tape Number	File Number	Record Length (in bytes)	Block Size (in bytes)
Input	102400	1	495	24750

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCCAN_CLCT.LNK follows:

```
$ LINK   USER2$DISK:[DAO.ROMWORK]SFCCAN_CLCT.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]JULIAN.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ, -
         USER2$DISK:[DAO.METTAPE]VATAPE.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCCAN_CLCT.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MET27:[SCRATCH.DAO]SFCCAN_APROCT888_CLCT.DAT      OUTPUT
$ ASSIGN MLAB: SYS$PRINT
$ TAPE SELECT TAPEDRIVE 102400
$ TAPE LOAD TAPEDRIVE 102400
$ MOUNT/NOWRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=24750 -
  TAPEDRIVE 102400 SFCTAP:
$ SHOW DEV/FULL TAPEDRIVE
$ RUN USER2$DISK:[DAO.ROMWORK]SFCCAN_CLCT.EXE
   01, 02
```

```
  04, 10, 1988
$ SHOW DEV/FULL TAPEDRIVE
$ @REWIND SFCTAP:
$ DISMOUNT SFCTAP:
$ DEALLOCATE TAPEDRIVE
```

(end of run stream)

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

A. Main Program

SFCCAN_CLCT

B. Subroutines

SIGNCK

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

USER2$DISK:[DAO.METTAPE]

VATAPE

B. Functions

JULIAN
JUNIT

IX. INCLUDE FILES

None

3.20 PREPROCESSOR SFCCAN_WNDW

I. PREPROCESSOR FUNCTION

This preprocessor (1) spatially and temporally windows the data output from SFCCAN_CLCT, (2) adds latitude and longitude coordinates to each station, (3) converts the time parameter from LST to GMT, and (4) converts a station's Canadian identifier to a WBAN or pseudo-WBAN number.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. CANID_XTABLE.DAT

CANID_XTABLE.DAT is the "look-up" file that contains the WBAN or our pseudo-WBAN identifiers for Canadian stations, as well as the stations' geographical coordinates and time offset from GMT. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 3.66.

```
      READ (UNITX, 120, END = 130) NEWID(K), LAT(K), LON(K),
     &                             ISHIFT(K), WBAN(K)
  120 FORMAT (1X, A5, 2X, F5.2, 2X, F6.2, 9X, I2, 2X, I5.5)
```

TABLE 3.66. CANID_XTABLE.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$NEWID_n$		Char*5	Canadian identifier
2	LAT_n	° N	Real*4	Station latitude
3	LON_n	° W	Real*4	Station longitude
4	$ISHIFT_n$	h	Integer*4	Offset from GMT
5	$WBAN_n$		Integer*4	WBAN or pseudo-WBAN identifier

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

b. SFCCAN_MMM_1MMM_2YYx_CLCT.DAT

SFCCAN_MMM_1MMM_2YYx_CLCT.DAT, where *x* ranges from A to B, is output by SFCCAN_CLCT. The file's parameters are shown in Table 3.67, and its READ and FORMAT statements are listed below.

```
140 READ (UNITIN, 150) (DUMMY(I), I = 1, 2)
150 FORMAT (A4)

160 READ (UNITIN, 180, END = 888) DATIME, ID, ADATA
180 FORMAT (I8.8, 1X, A5, A86)
```

TABLE 3.67. SFCCAN_MMM_1MMM_2YYx_CLCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DUMMY		Char*4	Dummy variable (skips the header record)
2	DATIME	LST	Integer*4	Date and time as *YYMMDDHH*
3	ID_n		Char*5	Canadian station identifier
4	$ADATA_{n,h}$		Char*86	Data stream, comprising:
	$OPCOV_{n,h}$	%	Real*4	Opaque sky cover
	$TCOV_{n,h}$		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
	$AMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
	$BASE1_{n,h}$	m	Real*4	Lowest cloud base
	$AMT2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
	$BASE2_{n,h}$	m	Real*4	Second-lowest cloud base
	$AMT3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
	$BASE3_{n,h}$	m	Real*4	Third-lowest cloud base
	$SLP_{n,h}$	mb	Real*4	Sea level pressure
	$WD_{n,h}$	deg	Real*4	Wind direction
	$WS_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Temperature
	$TD_{n,h}$	°C	Real*4	Dew point
	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

SFCCAN_WNDW produces SFCCAN_*MMMYY*_WNDW.DAT, a nonstandard file that is sorted and merged with the U.S. surface data; the resultant file is input to preprocessor SFC_RFMT. SFCCAN_*MMMYY*_WNDW.DAT is written in two steps: (1) a header step, and (2) a data step. The first WRITE statement below forms the header step, while the second forms the data step. The file header parameters are shown in Table 3.68, while the data parameters are shown in Table 3.69.

```
    WRITE (UNITOUT, 100) IHEAD, TOTFIL, XDAY, XTIME, PROGNM
100 FORMAT (1X, I5.5, ' ### ' , I1, ' FILES WINDOWED TO 1 ON ', A6,
  &        '/', A6, ' BY ', A<LEN>)

    WRITE (UNITOUT, 240) ZTIME, STNID, STNLAT, STNLON, ADATA
240 FORMAT (I8.8, 1X, I5.5, 1X, F5.2, 1X, F6.2, A86)
```

TABLE 3.68. SFCCAN_*MMMYY*_WNDW.DAT HEADER PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IHEAD		Integer*4	Number of output header lines
2	TOTFIL		Integer*4	Number of input data files
3	XDAY		Char*8	Date of program execution as *YYMMDD*
4	XTIME	LST	Char*8	Time of program execution as *HH*
5	PROGNM		Char*64	Program name

TABLE 3.69. SFCCAN_*MMMYY*_WNDW.DAT DATA PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ZTIME	GMT	Integer*4	Date and time as *YYMMDDHH*
2	$STNID_n$		Integer*4	WBAN/pseudo-WBAN identifier
3	$STNLAT_n$	° N	Real*4	Station latitude
4	$STNLON_n$	° W	Real*4	Station longitude
5	$ADATA_{n,h}$		Char*86	Data stream, comprising:
	$OPCOV_{n,h}$	%	Real*4	Opaque sky cover
	$TCOV_{n,h}$		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
8	$AMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
9	$BASE1_{n,h}$	m	Real*4	Lowest cloud base
10	$AMT2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
11	$BASE2_{n,h}$	m	Real*4	Second-lowest cloud base
12	$AMT3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
13	$BASE3_{n,h}$	m	Real*4	Third-lowest cloud base
14	$SLP_{n,h}$	mb	Real*4	Sea level pressure
15	$WD_{n,h}$	deg	Real*4	Wind direction
16	$WS_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
17	$T_{n,h}$	°C	Real*4	Temperature
18	$TD_{n,h}$	°C	Real*4	Dew point
19	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

III. CONTROL CARDS

Three control cards are used in the format shown below. Table 3.70 defines the control card variables.

BEGTIM, ENDTIM
MINLAT, MAXLAT, MINLON, MAXLON
TOTFIL

TABLE 3.70. CONTROL CARD VARIABLES

Variable Name	Unit	Description
BEGTIM	GMT	Beginning time for which data will be extracted as *YYMMDDHH*
ENDTIM	GMT	Ending time for which data will be extracted as *YYMMDDHH*
MINLAT	° N	Southern window boundary
MAXLAT	° N	Northern window boundary
MINLON	° W	Eastern window boundary
MAXLON	° W	Western window boundary
TOTFIL		Number of input data sets to be processed

Example:

88040100, 88043023
25.0 55.0 60.0 110.0
2

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	12	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	9	blocks
Executable file:	1	file	12	blocks
	4	files	33	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:04:00
Buffered I/O count:	426
Direct I/O count:	16790
Peak working-set size:	538
Peak virtual size:	3750

C. Space Requirements: Log and Print Files

SFCCAN_WNDW.LOG 45 blocks
Print Files: None

D. Space Requirements: Input and Output Files

Table 3.71 shows the approximate input and output file space requirements.

TABLE 3.71. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SFCCAN_*MMM*$_1$*MMM*$_2$*YY*A_CLCT.DAT	Nonstd	±80000	4 months
	SFCCAN_*MMM*$_1$*MMM*$_2$*YY*B_CLCT.DAT	Nonstd	± 40000	4 months
			±120000	
Output	SFCCAN_*MMMYY*_WNDW.DAT	Nonstd	±13000	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCCAN_WNDW.LNK follows:

```
$ LINK USER2$DISK:[DAO.ROMWORK]SFCCAN_WNDW.OBJ, -
       USER2$DISK:[DAO.DONLIB]IHSHFT.OBJ, -
       DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ, -
       DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCCAN_WNDW.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MET27:[SCRATCH.DAO]CANID_XTABLE.DAT              XTABLE
$ ASSIGN MET27:[SCRATCH.DAO]SFCCAN_APROCT88B_CLCT.DAT     INPUT1
$ ASSIGN MET27:[SCRATCH.DAO]SFCCAN_APROCT88A_CLCT.DAT     INPUT2
$ ASSIGN MET27:[SCRATCH.DAO]SFCCAN_JUN88_WNDW.DAT         OUTPUT
$ RUN USER2$DISK:[DAO.ROMWORK]SFCCAN_WNDW.EXE
 88060100  88063023
 25.0  55.0  60.0  110.0
 2
```

(end of run stream)

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

A. Main Program

SFCCAN_WNDW

B. Subroutines

USER2$DISK:[DAO.DONLIB]

IHSHFT

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

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.21 PREPROCESSOR SFC_RFMT

I. PREPROCESSOR FUNCTION

This preprocessor reformats the surface meteorology input data to conform with the SURMET1 format; it also adds header records that assist you in tracking the provenience of the output file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

SFC_RFMT uses one nonstandard input file, SFC_*MMMYY*_SORT.DAT, that contains the data shown in Table 3.72. This file is output by the VAX system SORT and MERGE routines on the output files from preprocessor SFCUS_CLD_INTP for U.S. data, and from preprocessor SFCCAN_WNDW for Canadian data. The READ and FORMAT statements for this file are listed below.

```
      READ (UNITIN, 60) HEADER
   60 FORMAT (A108)

  160 READ (UNITIN, 180, END = 999) DATIME, WBAN, INFO
  180 FORMAT (I8.8, 1X, I5.5, A99)
```

TABLE 3.72. SFC_*MMMYY*_SORT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATIME	GMT	Integer*4	Date and time as *YYMMDDHH*
2	$WBAN_n$		Integer*4	Station WBAN identifier
3	INFO		Char*99	Data stream, comprising:
	$SLAT_{n,h}$	° N	Real*4	Latitude
	$SLON_{n,h}$	° W	Real*4	Longitude
	$IOPAQ_{n,h}$	%	Real*4	Opaque sky cover
	$TOTCOV_{n,h}$		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
	$AMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
	$HT1_{n,h}$	m	Real*4	Lowest cloud layer base
	$AMT2_{n,h}$	%	Real*4	Coverage, second cloud layer
	$HT2_{n,h}$	m	Real*4	Second cloud layer base
	$AMT3_{n,h}$	%	Real*4	Coverage, third cloud layer
	$HT3_{n,h}$	m	Real*4	Third cloud layer base
	$SLP_{n,h}$	mb	Real*4	Sea level pressure
	$WDIR_{n,h}$	deg	Real*4	Wind direction
	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Temperature
	$TD_{n,h}$	°C	Real*4	Dew point
	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

SFC_RFMT outputs one nonstandard file, SFC_*MMMYY*_RFMT.DAT, that contains the data shown in Table 3.73, Table 3.74, and Table 3.75. Note that the file contains both file headers and an "hours" header in addition to the data parameters. The WRITE and FORMAT statements for SFC_*MMMYY*_RFMT.DAT are listed below.

```
    WRITE (UNITOUT, 60) HEADER
 60 FORMAT (A108)

    WRITE (UNITOUT, 100) J, XDAY, XTIME, PROGNM
100 FORMAT (1X, I5.5, ' ### FILE REFORMATTED ON ', A6, '/', A6,
   &        ' BY ', A<LEN>)

    WRITE (UNITOUT, 200) OLDTIM, I
200 FORMAT (I8.8, 5X, I3)

    WRITE (UNITOUT, 230) STNID(K), ADATA(K)
230 FORMAT (I5.5, A99)
```

TABLE 3.73. SFC_*MMMYY*_RFMT.DAT FILE HEADER PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	HEADER		Char*108	Previous header information
2	J		Integer*4	Record number of the new header
3	XDAY		Char*8	Date of processing as *MMDDYY*
4	XTIME	EST	Char*8	Time of processing as *HHMMSS*
5	PROGNM		Char*64	Program name

TABLE 3.74. SFC_*MMMYY*_RFMT.DAT "HOURS" HEADER PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	OLDTIM	GMT	Integer*4	Date and hour as *YYMMDDHH*
2	I		Integer*4	No. of reports at OLDTIM

TABLE 3.75. SFC_*MMMYY*_RFMT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$STNID_n$		Integer*4	Station WBAN identifier
2	ADATA		Char*99	Data stream, comprising:
	$SLAT_{n,h}$	° N	Real*4	Latitude
	$SLON_{n,h}$	° W	Real*4	Longitude
	$IOPAQ_{n,h}$	%	Real*4	Opaque sky cover
	$TOTCOV_{n,h}$		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
	$AMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
	$HT1_{n,h}$	m	Real*4	Lowest cloud layer base
	$AMT2_{n,h}$	%	Real*4	Coverage, second cloud layer
	$HT2_{n,h}$	m	Real*4	Second cloud layer base
	$AMT3_{n,h}$	%	Real*4	Coverage, third cloud layer
	$HT3_{n,h}$	m	Real*4	Third cloud layer base
	$SLP_{n,h}$	mb	Real*4	Sea level pressure
	$WDIR_{n,h}$	deg	Real*4	Wind direction
	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Temperature
	$TD_{n,h}$	°C	Real*4	Dew point
	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	21	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	12	blocks
Executable file:	1	file	15	blocks
	3	files	48	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:06:00
Buffered I/O count:	768
Direct I/O count:	6886
Peak working-set size:	507
Peak virtual size:	3750

C. Space Requirements: Log and Print Files

SFC_RFMT.LOG	3 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.76 shows the input file and output file space requirements.

TABLE 3.76. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SFC_*MMMYY*_SORT.DAT	Nonstd	±50000	1 month
Output	SFC_*MMMYY*_RFMT.DAT	Nonstd	±50000	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFC_RFMT.LNK follows:

```
$ LINK   USER2$DISK:[DAO.ROMWORK]SFC_RFMT.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFC_RFMT.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MET27:[SCRATCH.DAO]SFC_OCT88_SORT.DAT          INPUT
$ ASSIGN MET27:[SCRATCH.DAO]SFC_OCT88_RFMT.DAT          OUTPUT
$ RUN USER2$DISK:[DAO.ROMWORK]SFC_RFMT.EXE
```

(end of run stream)

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

A. Main Program

SFC_RFMT

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

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.22 PREPROCESSOR SFCR_HRFILL

I. PREPROCESSOR FUNCTION

This preprocessor reads the output of SFC_VRFY and looks for hours with data that are totally or partially missing. If only one hour is found to have data missing, the data are filled with the data for the previous hour.[23] If more than one consecutive hour has data missing, the first half of the period (including the center hour, if there is one) is filled with data from the last hour that has data. The second half of the period is filled with data from the next hour that has data.

Preprocessor COUNT_STNPERHR should be run before executing this preprocessor. If any short or missing hours exist the log of COUNT_STNPERHR shows the number of stations that report for those hours. The control card used in SFCR_HRFILL must be a number greater than the maximum number of stations in the hours that have partial data (where you want to replace these data), but less than the number that corresponds to the hour (among the data that you want to keep without change) that reports with the least number of stations. For example, you find an hour that reports with 155 stations and one that reports with 255 stations, while all other hours report with 300+ stations; you run program SHOWMET (Section 7.7) on the hour with 255 stations, and find good geographical representation and no obvious errors, and you decide to keep these data rather than replacing them. However, the data spread for the hour with 155 stations reporting is deficient, and you decide to replace these data. The number to be input into the control card must be greater than 155 but less than 255. However, there may be times when you cannot use this procedure, and you must develop your own quality control routines. For example, all hours except 2 report with 300+ stations; 1 of the 2 hours reports with 255 stations, the other with 260 stations. You run SHOWMET on both hours, and find that the hour with 255 stations has good geographical representation, and you decide to keep it. However, the hour with 260 stations reports no data from two states, and you decide to replace this hour. This preprocessor cannot accomplish the replacement of the hour with 260 reporting stations without also replacing the hour with 255 stations.

In practice, we have found that this preprocessor rarely needs to be used.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

---

23. Note that hours with partially missing data have the data that do exist replaced by the "filling" data.

2. Nonstandard Input Files

SFCR_HRFILL uses one nonstandard input file, SFC_*MMMYY*_VRFY.DAT;1, that contains the data shown in Table 3.77. This file is output by preprocessor SFC_VRFY, and its READ and FORMAT statements are listed below.

```
    READ (10, 200, END = 2) IDATE, NUMSTA
200 FORMAT (2I8)
    READ (10, 202) (DUMMY, I = 1, IDATE)
202 FORMAT (A1)
    READ (10, 202) (DUMMY, I = 1, NUMSTA)
    READ (10, 201) (TEXT(I), I = 1, NUMSTA)
201 FORMAT (A104)
```

TABLE 3.77. SFC_*MMMYY*_VRFY.DAT;1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IDATE	GMT	Integer*4	Date and time as *YYMMDDHH*
2	$NUMSTA_{n,h}$		Integer*4	No. of stations reporting for this hour
3	$DUMMY_{n,h}$		Char*1	Dummy variable (see note below)
4	$TEXT_{n,h}$		Char*104	Character string comprising:
	$WBAN_{n,h}$		Integer*4	Station WBAN identifier
	$SLAT_{n,h}$	° N	Real*4	Station latitude
	$SLON_{n,h}$	° W	Real*4	Station longitude
	$OPQSKY_{n,h}$	%	Real*4	Opaque sky cover
	$TOTSKY_{n,h}$		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
	$AMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
	$HGT1_{n,h}$	m	Real*4	Lowest cloud base
	$AMT2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
	$HGT2_{n,h}$	m	Real*4	Second-lowest cloud base
	$AMT3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
	$HGT3_{n,h}$	m	Real*4	Third-lowest cloud base
	$SLP_{n,h}$	mb	Real*4	Sea level pressure
	$WDIR_{n,h}$	deg	Real*4	Wind direction
	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Temperature
	$TD_{n,h}$	°C	Real*4	Dew point
	$STNPR_{n,h}$	mb	Real*4	Station pressure

Note: the DUMMY variable is used to skip header lines and data-short hours; *n* = surface station number (one or more digits); *h* = hour; missing values are either -99. or -9999.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

SFCR_HRFILL outputs one nonstandard file, SFC_*MMMYY*_VRFY.DAT;2, that contains the data shown in Table 3.78. This file is in the final SURMET1 format, and is input to processors P03G, P05G, P08G, P09G, P16G, P19G, and P27G. The WRITE and FORMAT statements for SFC_*MMMYY*_VRFY.DAT;2 are listed below.

```
    WRITE (11, 200) JDATE, NUMSTA
200 FORMAT (2I8)
    WRITE (11, 201) (TEXT(I), I = 1, NUMSTA)
201 FORMAT (A104)
```

TABLE 3.78. SFC_*MMMYY*_VRFY.DAT;2 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	JDATE	GMT	Integer*4	Date and time as *YYMMDDHH*
2	$NUMSTA_{n,h}$		Integer*4	No. of stations reporting for this hour
3	$TEXT_{n,h}$		Char*104	Character string comprising:
	$WBAN_{n,h}$		Integer*4	Station WBAN identifier
	$SLAT_{n,h}$	° N	Real*4	Station latitude
	$SLON_{n,h}$	° W	Real*4	Station longitude
	$OPQSKY_{n,h}$	%	Real*4	Opaque sky cover
	$TOTSKY_{n,h}$		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
	$AMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
	$HGT1_{n,h}$	m	Real*4	Lowest cloud base
	$AMT2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
	$HGT2_{n,h}$	m	Real*4	Second-lowest cloud base
	$AMT3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
	$HGT3_{n,h}$	m	Real*4	Third-lowest cloud base
	$SLP_{n,h}$	mb	Real*4	Sea level pressure
	$WDIR_{n,h}$	deg	Real*4	Wind direction
	$WSPD_{n,h}$	$m{\cdot}s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Temperature
	$TD_{n,h}$	°C	Real*4	Dew point
	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 3.79 defines the control card variable.

MINSTA

TABLE 3.79. CONTROL CARD VARIABLE

Variable Name	Unit	Description
MINSTA		Threshold for the minimum number of stations reporting per hour; hourly reports with the number of reporting stations falling below this threshold will have their data replaced.

Example:

250

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	3	files	12	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	3	files	12	blocks
Executable file:	1	file	9	blocks
	7	files	33	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:01:30
Buffered I/O count:	783
Direct I/O count:	6579
Peak working-set size:	522
Peak virtual size:	3885

C. Space Requirements: Log and Print Files

SFCR_HRFILL.LOG	3 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.80 shows the input file and output file space requirements.

TABLE 3.80. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SFC_*MMMYY*_VRFY.DAT;1	Nonstd	46278	1 month
Output	SFC_*MMMYY*_VRFY.DAT;2	Nonstd	46278	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCR_HRFILL.LNK follows:

```
$ LINK USER2$DISK:[DAO.ROMWORK]SFCR_HRFILL.OBJ, -
       USER2$DISK:[DAO.ROMWORK.LIB]NHSKIP.OBJ, -
       USER2$DISK:[DAO.ROMWORK.LIB]NEXTHR.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCR_HRFILL.COM follows:

```
$ ! PREP0001
$ SET VERIFY
$ ! RUN STREAM IS IN USER2$DISK:[DAO.ROMWORK]SFCR_HRFILL.COM
$ ASSIGN MET27:[SCRATCH.DAO]SFC_APR85_VRFY.DAT;1        SFCMET1
$ ASSIGN MET27:[SCRATCH.DAO]SFC_APR85_VRFY.DAT;2        SFCMET2
$ RUN USER2$DISK:[DAO.ROMWORK]SFCR_HRFILL.EXE
  250
```

(end of run stream)

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

A. Main Program

SFCR_HRFILL

B. Subroutines

None

C. Functions

USER2$DISK:[DAO.ROMWORK.LIB]

NEXTHR
NHSKIP

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

3.23 PREPROCESSOR SFCUS_CLCT

I. PREPROCESSOR FUNCTION

This preprocessor extracts the surface meteorological data from the TD-3280 format tapes acquired from the National Climatic Data Center (NCDC). SFCUS_CLCT (1) windows data given latitude and longitude delimiters, (2) converts some units to conform with SURMET1-type units, and (3) adjusts cloud data to conform with the SURMET1 format.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

Surface meteorology data tapes from the NCDC supply the data for SFCUS_CLCT. The tapes are in TD-3280 format, where the data are organized by station rather than by time. The NCDC archives each year of U.S. surface data on four tapes, all of which need to be processed for the ROMNET modeling region.[24] As a precaution against losing valuable data, a system tape copy is made for each original tape via the COPYTAPE preprocessor, and all processing proceeds on the copies. The READ and FORMAT statements for the tapes are listed below, and the parameters that are on the tapes are shown in Table 3.81.

```
      READ (UNITIN, 150, END = 777) BUFFER
150   FORMAT (A)
      READ (BUFFER(NBEG:NBEG+3), 200, ERR = 320) NBYTES
200   FORMAT (I4)
      READ (BUFFER(NBEG+4:NBEG+NBYTES-1), 220, ERR = 330) RECTYP,
     &     STNID, ELMTYP, EUNITS, IYEAR, IMON, SRC1, SRC2, IDAY,
     &     NUM, (ITIME(J), IVALUE(J), FLAG1(J), FLAG2(J), J = 1, NUM)
220   FORMAT (A3, I8, A4, A2, I4, I2, A1, A1, I2, I3, 100(I2, 2X, A6,
     &       A1, A1))
```

Note that an entire block of data is read into BUFFER (CHAR*12000), and then the record length of the individual records is obtained by reading NBYTES (INTEGER*4).

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

SFCUS_CLCT outputs one nonstandard file for each of the four input tapes,[25] SFCUS_*MMM*$_1$*MMM*$_2$*YYx*_CLCT.DAT, where *x* ranges from A to D. The output files are used by preprocessor SFCUS_WNDW. The file is written in two steps: (1) a header step, and (2) a data step. The first two WRITE statements below form the header step, while the third is the data step. The file parameters are shown in Table 3.82.

```
      WRITE (UNITOUT, 228) IYR, OLDMON, OLDDAY, K,
     &      OLDID, IOPAQ(K), TOTCOV(K), COVLOW(K),
     &      LOWHT(K), COVMID(K), MIDHT(K), COVHI(K),
     &      HIHT(K), SLPR(K), WDIR(K), WSPD(K),
     &      TEMP(K), DEWPT(K), STNPR(K)
228   FORMAT (4I2.2, 1X, I8.8, 1X, 2F4.0, 3(1X, F4.0, F6.0),
     &       1X, 5F7.1, 1X, F7.1)
```

24. Note that WBAN identification numbers for eastern North America are assigned first digits that can be either a 0, 1, 2, 5 or a 9. Second and third digits refer to the stations' geographical coordinates in units of 10° latitude and longitude respectively. Thus, stations that are geographically adjacent may be spatially far apart on the tapes because of the numerical ordering of the stations on the tapes.

25. ERROR*YYx*.DAT is written only if an error occurs in reading NBYTES, and in practice is never written provided that the tape data are error-free.

PRTEST*YYx*.DAT reports unidentified element types encountered when reading ELMTYP. The only element we ever found in this file was a sixth cloud layer, which would have a marginal effect on the ROM.

TABLE 3.81. TAPE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	RECTYP		Char*3	Type of data (always HLY)
2	$STNID_n$		Char*8	Station WBAN identifier
3	$ELMTYP_{n,h}$		Char*4	Element type, as follows:
	$ALTP_{n,h}$	in. Hg		Altimeter setting
	$CC51_{n,h}$			Sky condition, prior to 1951
	$CLC"x"_{n,h}$			Sky condition and cloud coverage for layer "x"
	$CLHT_{n,h}$	10^2 ft		Ceiling height
	$CLT"x"_{n,h}$			Cloud type and height for layer "x"
	$C2C3_{n,h}$			Fractional amount of sky covered by first 2 / first 3 cloud layers
	$DPTP_{n,h}$	°F		Dew point temperature
	$HZVS_{n,h}$	10^{-2} mi		Horizontal visibility
	$PRES_{n,h}$	in. Hg		Station pressure
	$PWTH_{n,h}$			Present weather
	$RHUM_{n,h}$	%		Relative humidity
	$SLVP_{n,h}$	mb		Sea level pressure
	$TMPD_{n,h}$	°F		Dry bulb temperature
	$TMPW_{n,h}$	°F		Wet bulb temperature
	$TSKC_{n,h}$			Total fractional sky/opaque cloud coverage
	$WD16_{n,h}$	knots		Wind direction and speed in 16 point WBAN code
	$WIND_{n,h}$	knots		Wind direction and speed
4	$EUNITS_{n,h}$		Char*2	Units code
5	$IYEAR_n$		Integer*4	Year
6	$IMON_n$		Integer*4	Month
7	$SRC1_{n,h}$		Char*1	Source code 1
8	$SRC2_{n,h}$		Char*1	Source code 2
9	$IDAY_n$		Integer*4	Day
10	NUM_n		Integer*4	No. of reports of ELMTYP for the day
11	$ITIME_n$	GMT	Integer*4	Time of report
12	$IVALUE_{n,h}$		Integer*4	Value of ELMTYP at ITIME
13	$FLAG1_{n,h}$		Char*1	Measurement value quality flag
14	$FLAG2_{n,h}$		Char*1	Quality flag

Note: n = surface station number (one or more digits); h = hour; missing values are typically represented by all 9's in IVALUE, or by the total absence of the IVALUE field.

TABLE 3.82. SFCUS_MMM_1MMM_2YYx_CLCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IYR		Integer*4	Year as *YY*
2	OLDMON		Integer*4	Month as *MM*
3	OLDDAY		Integer*4	Day as *DD*
4	K	LST	Integer*4	Hour as *HH*
5	$OLDID_n$		Integer*4	Station WBAN identifier
6	$IOPAQ_{n,h}$	%	Real*4	Opaque sky cover
7	$TOTCOV_{n,h}$		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
8	$COVLOW_{n,h}$	%	Real*4	Coverage, lowest cloud layer
9	$LOWHT_{n,h}$	m	Real*4	Lowest cloud base
10	$COVMID_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
11	$MIDHT_{n,h}$	m	Real*4	Second-lowest cloud base
12	$COVHI_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
13	$HIHT_{n,h}$	m	Real*4	Third-lowest cloud base
14	$SLPR_{n,h}$	mb	Real*4	Sea level pressure
15	$WDIR_{n,h}$	deg	Real*4	Wind direction
16	$WSPD_{n,h}$	$m{\cdot}s^{-1}$	Real*4	Wind speed
17	$TEMP_{n,h}$	°C	Real*4	Temperature
18	$DEWPT_{n,h}$	°C	Real*4	Dew point
19	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

III. CONTROL CARDS

One control card is used in the format shown below. Table 3.83 defines the control card variables.

BEGMON, ENDMON, TARYR

TABLE 3.83. CONTROL CARD VARIABLES

Variable Name	Unit	Description
BEGMON		Number of the first month to be extracted
ENDMON		Number of the last month to be extracted
TARYR		Year of the data to be extracted

Example:

04, 10, 1988

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	69	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	30	blocks
Executable file:	1	file	27	blocks
	3	files	126	blocks

B. Execution Time Requirements (per tape):

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:45:00
Buffered I/O count:	531
Direct I/O count:	10593
Peak working-set size:	511
Peak virtual size:	3750

C. Space Requirements: Log and Print Files

SFCUS_CLCT.LOG	9 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.84 shows the output file space requirements.

TABLE 3.84. OUTPUT FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Output	SFCUS_*MMM*$_1$*MMM*$_2$*YY*A_CLCT.DAT	Nonstd	50661	4 months
	SFCUS_*MMM*$_1$*MMM*$_2$*YY*B_CLCT.DAT	Nonstd	44244	4 months
	SFCUS_*MMM*$_1$*MMM*$_2$*YY*C_CLCT.DAT	Nonstd	41463	4 months
	SFCUS_*MMM*$_1$*MMM*$_2$*YY*D_CLCT.DAT	Nonstd	39468	4 months
	ERROR*YY*A.DAT	Nonstd	24 max	4 months
	PRTEST*YY*A.DAT	Nonstd	3 max	4 months
	ERROR*YY*B.DAT	Nonstd	24 max	4 months
	PRTEST*YY*B.DAT	Nonstd	3 max	4 months
	ERROR*YY*C.DAT	Nonstd	24 max	4 months
	PRTEST*YY*C.DAT	Nonstd	3 max	4 months
	ERROR*YY*D.DAT	Nonstd	24 max	4 months
	PRTEST*YY*D.DAT	Nonstd	3 max	4 months
			175944	

E. Space Requirements: Tape Files

Four tapes per year contain the U.S. surface meteorology data. Table 3.85 shows space requirements for a representative tape.

TABLE 3.85. INPUT TAPE FILE SPACE REQUIREMENTS

File Group	Tape Number	File Number	Record Length (in bytes)	Block Size (in bytes)
Input	101144	1	Variable, 1230 max	Variable, 12000 max

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCUS_CLCT.LNK follows:

```
$ LINK   USER2$DISK:[DAO.ROMWORK]SFCUS_CLCT.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]JULIAN.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCUS_CLCT.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MET27:[SCRATCH.DAO]SFCUS_SEPOCT88D_CLCT.DAT OUTPUT
$ ASSIGN MET27:[SCRATCH.DAO]ERROR88D.DAT DUMP
$ ASSIGN MET27:[SCRATCH.DAO]PRTEST88D.DAT TMPDAT
$ ASSIGN MLAB: SYS$PRINT
$ TAPE SELECT TAPEDRIVE 101144
$ TAPE LOAD TAPEDRIVE 101144
$ MOUNT/NOWRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=12000 -
  TAPEDRIVE 101144 SFCTAP:
$ SHOW DEV/FULL TAPEDRIVE
$ SET MAGTAPE/SKIP=FILES:1 SFCTAP:
$ RUN USER2$DISK:[DAO.ROMWORK]SFCUS_CLCT.EXE
   09, 10, 1988
$ SHOW DEV/FULL TAPEDRIVE
$ @REWIND SFCTAP:
$ DISMOUNT SFCTAP:
$ DEALLOCATE TAPEDRIVE
```

(end of run stream)

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

A. Main Program

SFCUS_CLCT

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

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JULIAN
JUNIT

IX. INCLUDE FILES

None

3.24 PREPROCESSOR SFCUS_CLD_INTP

I. PREPROCESSOR FUNCTION

Layered cloud data are routinely reported from surface observation sites at three-hourly intervals. This preprocessor copies each three-hourly cloud report to the hours immediately preceding and following the report. SFCUS_CLD_INTP also eliminates ground-level fog and other obscuring phenomena from the input data. The removal of these data prevents the potential masking of higher clouds, which are of greater importance in the ROM.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

SFCUS_CLD_INTP uses one nonstandard input file, SFCUS_*MMMYY*_WNDW.DAT, that contains the data shown in Table 3.86. This file is output by preprocessor SFCUS_WNDW. The READ and FORMAT statements for this file are listed below.

```
140 READ (UNITIN,160,END=888) YRDAY, HR, ID, ADATA, AMT1, HT1,
   &                          AMT2, HT2, AMT3, HT3, BDATA
160 FORMAT (I6.6,I2.2,1X,I5.5,A22,3(1X,F4.0,F6.0),A44)
```

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

SFCUS_CLD_INTP outputs one nonstandard file, SFCUS_*MMMYY*_WNDW.DAT, that contains the data shown in Table 3.87. This file is sorted by the VAX SORT utility, and merged with the sorted Canadian data using the VAX MERGE utility; the resultant file is input to preprocessor SFC_RFMT. The WRITE and FORMAT statements for SFCUS_*MMMYY*_WNDW.DAT are listed below.

```
    WRITE (UNITOUT,200) OLDTIME, OLDID, OLADAT, OLAMT1, OLHT1,
   &                    OLAMT2, OLHT2, OLAMT3, OLHT3, OLBDAT
200 FORMAT (I8.8,1X,I5.5,A22,3(1X,F4.0,F6.0),A44)
```

TABLE 3.86. SFCUS_*MMMYY*_WNDW.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	YRDAY		Integer*4	Date as *YYMMDD*
2	HR	GMT	Integer*4	Hour as *HH*
3	$ID_{n,h}$		Integer*4	Station ID WBAN (integer) number
4	ADATA		Char*22	Data stream, comprising:
	$SLAT_{n,h}$	° N	Real*4	Latitude
	$SLON_{n,h}$	° W	Real*4	Longitude
	$IOPAQ_{n,h}$	%	Real*4	Opaque sky cover
	$TOTCOV_{n,h}$		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
5	$AMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
6	$HT1_{n,h}$	m	Real*4	Lowest cloud base
7	$AMT2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
8	$HT2_{n,h}$	m	Real*4	Second-lowest cloud base
9	$AMT3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
10	$HT3_{n,h}$	m	Real*4	Third-lowest cloud base
11	BDATA		Char*44	Data stream, comprising:
	$SLP_{n,h}$	mb	Real*4	Sea level pressure
	$WDIR_{n,h}$	deg	Real*4	Wind direction
	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Temperature
	$TD_{n,h}$	°C	Real*4	Dew point
	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

TABLE 3.87. SFCUS_*MMMYY*_INTP.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	OLDTIME		Integer*4	Date and time as *YYMMDDHH*
2	$OLDID_{n,h}$		Integer*4	Station ID WBAN (integer) number
3	OLADAT		Char*22	Data stream, comprising:
	$SLAT_{n,h}$	° N	Real*4	Latitude
	$SLON_{n,h}$	° W	Real*4	Longitude
	$IOPAQ_{n,h}$	%	Real*4	Opaque sky cover
	$TOTCOV_{n,h}$		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
4	$OLAMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
5	$OLHT1_{n,h}$	m	Real*4	Lowest cloud base
6	$OLAMT2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
7	$OLHT2_{n,h}$	m	Real*4	Second-lowest cloud base
8	$OLAMT3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
9	$OLHT3_{n,h}$	m	Real*4	Third-lowest cloud base
10	OLBDAT		Char*44	Data stream, comprising:
	$SLP_{n,h}$	mb	Real*4	Sea level pressure
	$WDIR_{n,h}$	deg	Real*4	Wind direction
	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Temperature
	$TD_{n,h}$	°C	Real*4	Dew point
	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

III. CONTROL CARDS

One control card is used to input control data in the format shown below. This card cuts off the ultimate hour's data in the input file so that it is not written to the output file; this step eliminates future data redundancy. Table 3.88 defines the control card variable.

ENDTIM

TABLE 3.88. CONTROL CARD VARIABLE

Variable Name	Unit	Description
ENDTIM	hour as GMT	Noninclusive ending date and time for output file as *YYMMDDHH*

Example:

88080100 for July data

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	21	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	12	blocks
Executable file:	1	file	15	blocks
	3	files	48	blocks

B. Execution Time Requirements (Representative Values for a One-Month Data Span):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:06:00
Buffered I/O count:	612
Direct I/O count:	4842
Peak working-set size:	507
Peak virtual size:	3750

C. Space Requirements: Log and Print Files

SFCUS_CLD_INT.LOG: 3 blocks
Print Files: None

D. Space Requirements: Input and Output Files

Table 3.89 shows the input file and output file space requirements.

TABLE 3.89. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SFCUS_*MMMYY*_WNDW	Nonstd	34146	1 month
Output	SFCUS_*MMMYY*_INTP	Nonstd	34146	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCUS_CLD_INT.LNK follows:

```
$ LINK   USER2$DISK:[DAO.ROMWORK]SFCUS_WNDW.OBJ, -
         USER2$DISK:[DAO.DONLIB]IHSHFT.OBJ. -
         DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCUS_CLD_INT.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN WORK_SCRATCH:SFCUS_JUL88_WNDW.DAT          INPUT
$ ASSIGN WORK_SCRATCH:SFCUS_JUL88_INTP.DAT          OUTPUT
$ RUN USER2$DISK:[DAO.ROMWORK]SFCUS_CLD_INTP.EXE
  88080100
```

(end of run stream)

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

A. Main Program

SFCUS_CLD_INTP

B. Subroutines

None

C. Functions

IHSHFT

D. Block Data Files

None

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

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.25 PREPROCESSOR SFCUS_WNDW

I. PREPROCESSOR FUNCTION

This preprocessor (1) windows surface meteorological data by time and by region, (2) adds latitude and longitude coordinates to each record, (3) converts the time in each record from LST to GMT, and (4) for the ROMNET region, produces a single output file from four meteorology data input files.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. STNS_US_DEGFRAC.LIS

STNS_US_DEGFRAC.LIS is a station "look-up" file that contains (1) the latitude and longitude coordinates for all surface meteorology stations east of the Rocky Mountains, and (2) the difference, in hours, between GMT and LST. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 3.90.

```
      READ (UNITX, 120, END = 130) NEWID(K), LAT(K), ALAT(K), LON(K),
     &                             ALON(K), ISHIFT(K)
  120 FORMAT (1X, I5.5, 14X, F5.2, A1, 1X, F6.2, A1, 8X, I2)
```

TABLE 3.90. STNS_US_DEGFRAC.LIS PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$NEWID_n$		Integer*4	Station WBAN identifier
2	LAT_n	°	Real*4	Latitude
3	$ALAT_n$	N	Char*1	Hemisphere indicator
4	LON_n	°	Real*4	Longitude
5	$ALON_n$	W	Char*1	Hemisphere indicator
6	$ISHIFT_n$	h	Integer*4	Hour shift, LST → GMT

Note: n = surface station number

b. SFCUS_MMM_1MMM_2YYx_CLCT.DAT (x = A - D)

These four files contain the surface meteorology parameters, and are output by SFCUS_CLCT. The READ and FORMAT statements for the files are listed below, and their parameters are shown in Table 3.91. Note that the Char*4 variable DUMMY is used to bypass the file header information.

```
140 READ (UNITIN, 150) (DUMMY(I), I = 1, 2)
150 FORMAT (A4)
160 READ (UNITIN, 180, END = 888) DATIME, PREID, ID, ADATA
180 FORMAT (I8.8, 1X, I3, I5.5, A86)
```

TABLE 3.91. SFCUS_*MMMYY*_CLCT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DATIME	LST	Integer*4	Date and hour as *YYMMDDHH*
2	$PREID_n$		Char*3	WBAN ID prefix
3	ID_n		Integer*4	WBAN ID
4	ADATA		Char*86	Data stream, comprising:
	$IOPAQ_{n,h}$	%	Real*4	Opaque sky cover
	$TOTCOV_{n,h}$		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
	$COVLOW_{n,h}$	%	Real*4	Coverage, lowest cloud layer
	$LOWHT_{n,h}$	m	Real*4	Lowest cloud base
	$COVMID_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
	$MIDHT_{n,h}$	m	Real*4	Second-lowest cloud base
	$COVHI_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
	$HIHT_{n,h}$	m	Real*4	Third-lowest cloud base
	$SLP_{n,h}$	mb	Real*4	Sea level pressure
	$WDIR_{n,h}$	deg	Real*4	Wind direction
	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Temperature
	$TD_{n,h}$	°C	Real*4	Dew point
	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

SFCUS_WNDW outputs one nonstandard file, SFCUS_*MMMYY*_WNDW.DAT, that contains the data shown in Table 3.92. The file is input to SFCUS_CLD_INTP. The WRITE and FORMAT statements for SFCUS_*MMMYY*_WNDW.DAT are listed below.

```
    WRITE (UNITOUT, 100) IHEAD, TOTFIL, XDAY, XTIME, PROGNM
100 FORMAT (1X, I5.5, ' ### ' , I1, ' FILES WINDOWED TO 1 ON ', A6,
  & '/', A6, ' BY ', A<LEN>)
    WRITE (UNITOUT, 240) ZTIME, ID, STNLAT, STNLON, ADATA
240 FORMAT (I8.8, 1X, I5.5, 1X, F5.2, 1X, F6.2, A86)
```

TABLE 3.92. SFCUS_*MMMYY*_INTP.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IHEAD		Integer*4	No. of header lines per input file
2	TOTFIL		Integer*4	Total number of input files
3	XDAY		Char*8	Date of program run as *YYMMDD*
4	XTIME	LST	Char*8	Local time of program run as *HH*
5	PROGNM		Char*64	Program name
6	ZTIME	GMT	Integer*4	Date and hour as *YYMMDDHH*
7	$ID_{n,h}$		Integer*4	Station WBAN identifier
8	$STNLAT_{n,h}$	° N	Real*4	Latitude
9	$STNLON_{n,h}$	° W	Real*4	Longitude
10	$ADATA_{n,h}$		Char*86	Data stream, comprising:
	$IOPAQ_{n,h}$	%	Real*4	Opaque sky cover
	$TOTCOV_{n,h}$		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
	$COVLOW_{n,h}$	%	Real*4	Coverage, lowest cloud layer
	$LOWHT_{n,h}$	m	Real*4	Lowest cloud base
	$COVMID_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
	$MIDHT_{n,h}$	m	Real*4	Second-lowest cloud base
	$COVHI_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
	$HIHT_{n,h}$	m	Real*4	Third-lowest cloud base
	$SLP_{n,h}$	mb	Real*4	Sea level pressure
	$WDIR_{n,h}$	deg	Real*4	Wind direction
	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
	$T_{n,h}$	°C	Real*4	Temperature
	$TD_{n,h}$	°C	Real*4	Dew point
	$STNPR_{n,h}$	mb	Real*4	Station pressure

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

III. CONTROL CARDS

Three control cards are used in the format shown below. Table 3.93 defines the control card variables.

BEGTIM, ENDTIM
MINLAT, MAXLAT, MINLON, LAXLON
TOTFIL

TABLE 3.93. CONTROL CARD VARIABLES

Variable Name	Unit	Description
BEGTIM	GMT	Beginning date and time of output as *YYMMDDHH*
ENDTIM	GMT	Ending date and time of output as *YYMMDDHH*
MINLAT	° N	Southern latitude boundary of output
MAXLAT	° N	Northern latitude boundary of output
MINLON	° W	Eastern longitude boundary of output
MAXLON	° W	Western longitude boundary of output
TOTFIL		Number of input data files

Example:

88070100 88080100
25.0 55.0 60.0 110.0
4

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:	0	files	0	blocks
Object files:	1	file	9	blocks
Executable file:	1	file	12	blocks
	3	files	33	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:06:00
Buffered I/O count:	629
Direct I/O count:	13731
Peak working-set size:	535
Peak virtual size:	3750

C. Space Requirements: Log and Print Files

SFCUS_WNDW.LOG	15 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.94 shows the input file and output file space requirements.

TABLE 3.94. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SFCUS_*MMM*$_1$*MMM*$_2$*YY*A_CLCT.DAT	Nonstd	50661	4 months
	SFCUS_*MMM*$_1$*MMM*$_2$*YY*B_CLCT.DAT	Nonstd	44244	4 months
	SFCUS_*MMM*$_1$*MMM*$_2$*YY*C_CLCT.DAT	Nonstd	41463	4 months
	SFCUS_*MMM*$_1$*MMM*$_2$*YY*D_CLCT.DAT	Nonstd	39468	4 months
	STNS_US_DEGFRAC.LIS	Nonstd	30	
			175866	
Output	SFCUS_*MMMYY*_WNDW.DAT	Nonstd	34146	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCUS_WNDW.LNK follows:

```
$ LINK   USER2$DISK:[DAO.ROMWORK]SFCUS_WNDW.OBJ, -
         USER2$DISK:[DAO.ROMWORK]IHSHFT.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ, -
         DISK$VAXSET:[ROMOBJ.UTILIO]ADATE.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCUS_CLD_INT.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MET27:[SCRATCH.DAO]STNS_US_DEGFRAC.LIS          XTABLE
$ ASSIGN WORK_SCRATCH:SFCUS_MAYAUG88A_CLCT.DAT           INPUT1
$ ASSIGN WORK_SCRATCH:SFCUS_MAYAUG88B_CLCT.DAT           INPUT2
$ ASSIGN WORK_SCRATCH:SFCUS_MAYAUG88C_CLCT.DAT           INPUT3
$ ASSIGN WORK_SCRATCH:SFCUS_MAYAUG88D_CLCT.DAT           INPUT4
$ ASSIGN WORK_SCRATCH:SFCUS_JUL88_WNDW.DAT               OUTPUT
$ RUN USER2$DISK:[DAO.ROMWORK]SFCUS_WNDW.EXE
  88070100  88080100
  25.0  55.0  60.0  110.0
  4
```

(end of run stream)

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

A. Main Program

SFCUS_WNDW

B. Subroutines

None

C. Functions

IHSHFT

D. Block Data Files

None

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

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.26 PREPROCESSOR SFC_VRFY

I. PREPROCESSOR FUNCTION

This preprocessor replaces questionable surface meteorology parameter values with a "missing" flag. It compares observed magnitudes of winds, sea level pressure, air temperatures, and dew point temperatures against Barnes-analyzed values (Barnes, 1973) at the same points. If an observed parameter departs from the Barnes-analyzed value by greater than a margin that you define in the preprocessor's control card, then the observed parameter is replaced with the "missing" flag.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

SFC_VRFY uses one nonstandard input file, SFC_*MMMYY*_RFMT.DAT, that contains the data shown in Table 3.95. This file is output by preprocessor SFC_RFMT, and its READ and FORMAT statements are listed below.

```
 40 READ (UNITIN, 50) NUM
 50 FORMAT (I4.4)
    READ (UNITIN, 60) DUMMY
 60 FORMAT (A8)

    READ (UNITIN, 205, END = 3) IY, IM, ID, IH, NSTA
205 FORMAT (4I2.2, I8)

    READ (UNITIN, 206, END = 999) WBAN(N), SLAT(N), SLON(N),
    &   OPQSKY(N), TOTSKY(N), COV1(N), HGT1(N), COV2(N), HGT2(N),
    &   COV3(N), HGT3(N), SLVP(N), WDIR(N), WSPD(N), TEMP(N),
    &   DEWP(N), STAP(N)
206 FORMAT (I5.5, F6.2, F7.2, F5.0, F4.0, 3(F5.0, F6.0), F8.1,
&            4F7.1, F8.1)
```

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

SFC_VRFY outputs one nonstandard file, SFC_*MMMYY*_VRFY.DAT, that contains the data shown in Table 3.96. This file is in the final SURMET1 format, and is input to processors P03G, P05G, P08G, P09G, P16G, P19G, and P27G. The WRITE and FORMAT statements for SFC_*MMMYY*_VRFY.DAT are listed below.

```
    WRITE (UNITOUT, 205) IY, IM, ID, IH, NSTA
205 FORMAT (4I2.2, I8)
    WRITE (UNITOUT, 206) WBAN(N), SLAT(N), SLON(N), OPQSKY(N),
   &      TOTSKY(N), COV1(N), HGT1(N), COV2(N), HGT2(N), COV3(N),
   &      HGT3(N), SLVP(N), WDIR(N), WSPD(N), TEMP(N), DEWP(N),
   &      STAP(N)
206 FORMAT (I5.5, F6.2, F7.2, F5.0, F4.0, 3(F5.0, F6.0), F8.1,
   &       4F7.1, F8.1)
```

TABLE 3.95. SFCUS_*MMMYY*_RFMT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	NUM		Integer*4	Number of header lines
2	DUMMY		Char*8	Dummy variable
3	IY		Integer*4	Year as *YY*
4	IM		Integer*4	Month as *MM*
5	ID		Integer*4	Day as *DD*
6	IH	GMT	Integer*4	Hour as *HH*
7	NSTA		Integer*4	Number of reports at time IH
8a	$WBAN_{n,h}$		Char*4	Station ID non-WBAN format
8b	$WBAN_{n,h}$		Integer*4	Station WBAN identifier
9	$SLAT_{n,h}$	° N	Real*4	Latitude
10	$SLON_{n,h}$	° W	Real*4	Longitude
11	$OPQSKY_{n,h}$	%	Real*4	Opaque sky cover
12	$TOTSKY_{n,h}$		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
13	$COV1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
14	$HGT1_{n,h}$	m	Real*4	Lowest cloud base
15	$COV2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
16	$HGT2_{n,h}$	m	Real*4	Second-lowest cloud base
17	$COV3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
18	$HGT3_{n,h}$	m	Real*4	Third-lowest cloud base
19	$SLVP_{n,h}$	mb	Real*4	Sea level pressure
20	$WDIR_{n,h}$	deg	Real*4	Wind direction
21	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
22	$TEMP_{n,h}$	°C	Real*4	Temperature
23	$DEWP_{n,h}$	°C	Real*4	Dew point
24	$STAP_{n,h}$	mb	Real*4	Station pressure

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

TABLE 3.96. SFC_*MMMYY*_VRFY.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	IY		Integer*4	Year as *YY*
2	IM		Integer*4	Month as *MM*
3	ID		Integer*4	Day as *DD*
4	IH	GMT	Integer*4	Hour as *HH*
5	NSTA		Integer*4	Number of reports at time IH
6a	$WBAN_{n,h}$		Char*4	Station ID non-WBAN format
6b	$WBAN_{n,h}$		Integer*4	Station WBAN identifier
7	$SLAT_{n,h}$	° N	Real*4	Latitude
8	$SLON_{n,h}$	° W	Real*4	Longitude
9	$OPQSKY_{n,h}$	%	Real*4	Opaque sky cover
10	$TOTSKY_{n,h}$		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
11	$AMT1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
12	$HGT1_{n,h}$	m	Real*4	Lowest cloud base
13	$AMT2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
14	$HGT2_{n,h}$	m	Real*4	Second-lowest cloud base
15	$AMT3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
16	$HGT3_{n,h}$	m	Real*4	Third-lowest cloud base
17	$SLVP_{n,h}$	mb	Real*4	Sea level pressure
18	$WDIR_{n,h}$	deg	Real*4	Wind direction
19	$WSPD_{n,h}$	$m \cdot s^{-1}$	Real*4	Wind speed
20	$TEMP_{n,h}$	°C	Real*4	Temperature
21	$DEWP_{n,h}$	°C	Real*4	Dew point
22	$STAP_{n,h}$	mb	Real*4	Station pressure

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

III. CONTROL CARDS

One control card is used to input control data in the format shown below. This card defines the margin of difference that the program permits between the observed values and the Barnes-analyzed values before it replaces data with a "missing" flag. Table 3.97 defines the control card variables.

SLVPDF, WINDDF, TEMPDF, DEWPDF

TABLE 3.97. CONTROL CARD VARIABLES

Variable Name	Unit	Description
SLVPDF	mb	Deviation margin for sea level pressure
WINDDF	$m \cdot s^{-1}$	Deviation margin for winds
TEMPDF	°C	Deviation margin for air temperature
DEWPDF	°C	Deviation margin for dew point temperature

Example:

3.0 4.0 4.0 4.0

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	42	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	21	blocks
Executable file:	1	file	21	blocks
	3	files	82	blocks

B. Execution Time Requirements (Representative Values for a One-Month Data Span):

	VAX 8650
Charged CPU time (hh:mm:ss):	± 01:00:00
Buffered I/O count:	1193
Direct I/O count:	8205
Peak working-set size:	703
Peak virtual size:	3755

C. Space Requirements: Log and Print Files

SFC_VRFY.LOG	261 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.98 shows the input file and output file space requirements.

TABLE 3.98. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SFC_*MMMYY*_RFMT	Nonstd	46278	1 month
Output	SFC_*MMMYY*_VRFY	Nonstd	46278	1 month

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCUS_CLD_INT.LNK follows:

```
$ LINK    USER2$DISK:[DAO.ROMWORK]SFC_VRFY.OBJ, -
          DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

SFCUS_CLD_INT.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN WORK_SCRATCH:SFC_JUL88_RFMT.DAT              SFCIN
$ ASSIGN MET27:[SCRATCH.DAO]SFC_JUL88_VRFY.DAT        SFCOUT
$ RUN USER2$DISK:[DAO.ROMWORK]SFC_VRFY.EXE
  3.0, 4.0, 4.0, 4.0
```

(end of run stream)

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

A. Main Program

SFC_VRFY

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

3.27 PREPROCESSOR STAT_RFMT

I. PREPROCESSOR FUNCTION

This preprocessor reformats the statistical data that are contained in BUOY_STAT.LOG into an intermediate form from which preprocessor BUOY_RFMT produces a SURMET1-type format.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

One nonstandard file, BUOY_STAT.LOG, that contains the data shown in Table 3.99 is required as input. The file is output by preprocessor BUOY_STAT. The READ and FORMAT statements for the file are listed below.

```
    READ (13, 120, END = 999) DUMMY
120 FORMAT (A80)

    READ (DUMMY, 140) BUOYID, LAT, LON, IMON
140 FORMAT (30X, I5, 2X, F5.2, 1X, F6.2, 10X, I2)

    READ (13, 200) TEMP, IREP
    READ (13, 200) SST, IREP
    READ (13, 200) DELT, IREP
200 FORMAT (11X, F7.1, 10X, I4)
```

TABLE 3.99. BUOY_STAT.LOG PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DUMMY		Char*80	Dummy record
2	$BUOYID_n$		Integer*4	Buoy identifier in WBAN format
3	LAT_n	° N	Real*4	Latitude
4	LON_n	° W	Real*4	Longitude
5	$IMON_n$		Integer*4	Month as *MM*
6	$TEMP_n$	°C	Real*4	Averaged air temperature
7	SST_n	°C	Real*4	Averaged sea surface temperature
8	$DELT_n$	°C	Real*4	Averaged ((sea surface temp.) - (air temp.))
9	$IREP_n$		Integer*4	Number of reports averaged

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

STAT_RFMT outputs one nonstandard file, BUOY_STAT*YY*.DAT, that contains the data shown in Table 3.100. The file is input to preprocessor BUOY_RFMT. The WRITE and FORMAT statements are listed below.

```
    WRITE (14, 300) ITIME, BUOYID, LAT, LON, ADATA, BDATA,
   &                TEMP, SST, DELT
300 FORMAT (I8, 1X, I5, 1X, F5.2, 1X, F6.2, A44, A20, 2F7.1,
   &        1X, F7.1)
```

TABLE 3.100. BUOY_STATYY.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ITIME	GMT	Integer*4	First hour of the month as *YYMMDDHH*
2	$BUOYID_n$		Integer*4	Buoy identifier in WBAN format
3	LAT_n	° N	Real*4	Latitude
4	LON_n	° W	Real*4	Longitude
5	$ADATA_n$		Char*44	Data stream representing cloud data
6	$BDATA_n$		Char*20	Data stream representing pressure and wind data
7	$TEMP_n$	°C	Real*4	Monthly average air temperature
8	SST_n	°C	Real*4	Monthly average sea surface temperature
9	$DELT_n$	°C	Real*4	Monthly average (sea surface temp.) - (air temp.)

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

III. CONTROL CARDS

One control card is used in the format shown below. Table 3.101 defines the control card variable.

IYR

TABLE 3.101. CONTROL CARD VARIABLE

Variable Name	Unit	Description
IYR		Year as *YY*

Example:

88

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	6	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	6	blocks
Executable file:	1	file	9	blocks
	3	files	21	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:00:10
Buffered I/O count:	195
Direct I/O count:	278
Peak working-set size:	603
Peak virtual size:	3755

C. Space Requirements: Log and Print Files

STAT_RFMT.LOG	3 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.102 shows the input file and output file space requirements.

TABLE 3.102. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	BUOY_STAT.LOG	Nonstd	±1300	7 months
Output	BUOY_AVGYY.DAT	Nonstd	±100	7 months

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

STAT_RFMT.LNK follows:

```
$ ! ACCOUNTING CODE
$ LINK          USER2$DISK:[DAO.BUOY]STAT_RFMT.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

STAT_RFMT.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_STAT.LOG          INPUT
$ ASSIGN MET27:[SCRATCH.DAO]BUOY_STAT88.DAT        OUTPUT
$ RUN USER2$DISK:[DAO.BUOY]STAT_RFMT.EXE
  88
```

(end of run stream)

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

A. Main Program

STAT_RFMT

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

3.28 PREPROCESSOR XTRACTLL

I. PREPROCESSOR FUNCTION

This preprocessor spatially windows buoy data.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

One nonstandard file, BUOYYY.DAT, that contains the data shown in Table 3.103 is required as input. The file is output by preprocessor OFFTAPE. The READ and FORMAT statements for the file are listed below.

```
1059  READ (IDEV, 1103, IOSTAT=IOST) BUFF
1103  FORMAT (A<BUFLEN>)

      READ (BUFF(10:14), 1105, ERR=2001 ) BUOYID
1105  FORMAT (A5)

      READ (BUFF(17:20), 1107, ERR=2001 ) YEAR
1107  FORMAT (I4)

      READ (BUFF(21:22), 1107, ERR=2001 ) MONTH
      READ (BUFF(23:24), 1107, ERR=2001 ) DAY
      READ (BUFF(25:26), 1107, ERR=2001 ) HOUR
      READ (BUFF(29:31), 1009, ERR=2001 ) LAT
1009  FORMAT (F4.0)

      READ (BUFF(32:32), 1011) ALAT
1011  FORMAT (A1)

READ  (BUFF(33:36), 1009, ERR=2001 ) LON
READ  (BUFF(37:37), 1011) ALON
```

TABLE 3.103. BUOYYY.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	BUFF		Char*80	Data stream; the following parameters are read:
	$BUOYID_n$		Char*5	Buoy identifier
	YEAR		Integer*4	Year as *YY*
	MONTH		Integer*4	Month as *DD*
	DAY		Integer*4	Day as *DD*
	HOUR		Integer*4	Hour as *HH*
	LAT_n	° N	Real*4	Latitude
	$ALAT_n$		Char*1	Latitude hemisphere indicator
	LON_n	° W	Real*4	Longitude
	$ALON_n$		Char*1	Longitude hemisphere indicator

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

XTRACTLL produces one nonstandard file, XTR_BUOYYY.DAT, that contains the parameter shown in Table 3.104. The file is input to preprocessor BUOY_WNDW. The WRITE and FORMAT statements for this file are listed below.

```
      WRITE(OUTDEV, 1005, IOSTAT=IOST) BUFF
 1005 FORMAT(A100)
```

TABLE 3.104. XTR_BUOYYY.DAT PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	BUFF		Char*100	Buoy data record

III. CONTROL CARDS

Two control cards are used in the format shown below. Table 3.105 defines the control card variables.

CLABEL
NELAT, NELON, SWLAT, SWLON

TABLE 3.105. CONTROL CARD VARIABLES

Variable Name	Unit	Description
CLABEL		Descriptive label of the following control card
NELAT	° N	Northern latitude boundary
NELON	° W	Eastern longitude boundary
SWLAT	° N	Southern latitude boundary
SWLON	° W	Western longitude boundary

Example:

NELAT	NELON	SWLAT	SWLON
50.0	64.0	20.0	99.0

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	9	blocks
FORTRAN INCLUDE files:	2	files	6	blocks
Object files:	1	file	6	blocks
Executable file:	1	file	15	blocks
	5	files	36	blocks

B. Execution Time Requirements (per file):

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:05:00
Buffered I/O count:	847
Direct I/O count:	10630
Peak working-set size:	597
Peak virtual size:	3660

C. Space Requirements: Log and Print Files

XTRACTLL.LOG	15 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 3.106 shows the input and output file space requirements.

TABLE 3.106. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	BUOYYY.DAT	Nonstd	±180000	1 year
Output	XTR_BUOYYY.DAT	Nonstd	±80000	1 year

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

XTRACTLL.LNK follows:

```
$ ! ACCOUNTING CODE
$ ! link in USER2$DISK:[DAO.BUOY]XTRACTLL.LNK
$ ASSIGN / USER USER2$DISK:[DAO.BUOY]XTRACTLL.OBJ          A01
$ ASSIGN / USER USER2$DISK:[DAO.BUOY]RDBUOY.OBJ            A02
$ ASSIGN / USER USER2$DISK:[DAO.BUOY]YULIAN.OBJ            A03
$ ASSIGN / USER USER2$DISK:[DAO.BUOY]WINDOW.OBJ            A04
$ ASSIGN / USER DISK$VAXSET:[ROMOBJ.UTILIO]JFILE2.OBJ      A05
$ ASSIGN / USER DISK$VAXSET:[ROMOBJ.UTILIO]JUNIT.OBJ       A06
$ ASSIGN / USER DISK$VAXSET:[ROMOBJ.UTILIO]IOCL.OBJ        A07
$ LINK -
  / EXECUTABLE = USER2$DISK:[DAO.BUOY]XTRACTLL.EXE -
  A01, -
  A02, -
  A03, -
  A04, -
  A05, -
  A06, -
  A07
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

XTRACTLL.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM in USER2$DISK:[DAO.BUOY]XTRACTLL.COM
$ ASSIGN MET27:[SCRATCH.DAO]BUOY88.DAT              RAWFILE
$ ASSIGN MET27:[SCRATCH.DAO]XTR_BUOY88.DAT          XTRFILE
$ RUN USER2$DISK:[DAO.BUOY]XTRACTLL.EXE
NELAT NELON SWLAT SWLON
50.0  64.0  20.0  99.0
```

(end of run stream)

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

A. Main Program

XTRACTLL

B. Subroutines

None

C. Functions

USER2$DISK:[DAO.BUOY]

RDBUOY
YULIAN
WINDOW

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JUNIT

IX. INCLUDE FILES

USER2$DISK:[DAO.BUOY]

LLWBUF
RECBUF

SECTION 4

THE ANTHROPOGENIC EMISSIONS DATABASE

The anthropogenic emissions data that we use in the Regional Oxidant Model (ROM) are derived from and are a part of the 1985 NAPAP (National Acid Precipitation Assessment Program) Modelers' Emissions Inventory Version 2. The 1985 NAPAP emissions inventory coverage is shown in Table 4.1.

TABLE 4.1. 1985 NAPAP ANTHROPOGENIC EMISSIONS INVENTORY COVERAGE

Coverage	Description	
Anthropogenic point sources	Stationary sources emitting ≥100 tons of criteria pollutants in 1985	
Anthropogenic area sources	County-level area sources, stationary sources emitting <100 tons of criteria pollutants in 1985	
Natural area sources	Emissions of sulfur and nitrogen compounds, hydrocarbon species, and ammonia from biogenic and geophysical processes; TSP from natural sources.	
Geographic domain	The 48 contiguous states; Canada south of 60 °N and east of 125 °W.	
	Annual Emissions Inventory	Regional Modeling Emissions Inventory
Pollutants	SO_2, SO_4, CO, TSP, HCL, HF, NO_x, NH_3, VOC, THC	SO_2, SO_4, CO, TSP, HCL, HF, NO, NO_2, NH_3, VOC, THC, 30 VOC species classes
Temporal resolution	Annual	Hourly emissions values for typical weekday, Saturday, and Sunday for all four seasons.
Spatial resolution	Latitude and longitude coordinates for point sources; area sources at the county level.	Point and area sources assigned to grid cells of $1/6°$ latitude by $1/4°$ longitude (≈19 x 19 km).

Source: Modica *et al.*, 1989

Although one of the primary functions of the 1985 NAPAP Modelers' Emissions Inventory Version 2 is to support regional- and national-scale acid deposition modeling efforts, these same data are used in the ROM to model ozone concentrations. The Flexible Regional Emissions Data System (FREDS) processes U.S. and

Canadian point- and area-source data from 1985 annual inventories into a format suitable for the ROM, providing CO, NO_x, and VOC data.[1] Note that area-source emissions are comprised of nonmobile and mobile sources of pollutants.

4.1 AN OVERVIEW OF FREDS

FREDS consists of seven primary modules written in SAS and FORTRAN; these are outlined in Sections 4.1.1 to 4.1.7. FREDS extracts data from the annual inventories, allocates temporal and spatial factors, speciates selected pollutants, converts data to modeling format, and performs quality control checks at various stages of processing. Control Options Files are associated with each module; using these, we can specify the parameters we want to retrieve. We can also vary the allocation methodology or speciation chemistry by changing or replacing one or more of the allocation factor files. This design flexibility allows FREDS to be used for processing emission inventories for a wide variety of applications. Point- and area-source data are processed separately. Figure 4.1 shows a simplified flowchart of FREDS processing for the ROM.[2]

4.1.1 The Hydrocarbon Preprocessor

The first module is the Hydrocarbon Preprocessor (HCPREP). The HCPREP is run on the preliminary data, and can interconvert VOC (volatile organic compounds) and THC (total hydrocarbons). The ROM uses the nonmethane VOC emissions output by HCPREP. HCPREP also compensates for the missing mass of aldehydes in emissions estimates.

4.1.2 The Model Data Extraction Module

The second module in FREDS is the Model Data Extraction Module (MDEM). MDEM reformats and reduces the U.S. and Canadian point- and area-source emissions data into the FREDS processing format.

1. Further information on, or permission to access, the NAPAP database and the FREDS can be obtained from the Emissions and Modeling Branch of the Air and Energy Engineering Research Laboratory, U.S. Environmental Protection Agency, MD-62, Research Triangle Park, NC 27711. (919)541-2612.
2. One difference between FREDS processing for the ROM and acid deposition models is that the ROM does not require TSP as input.

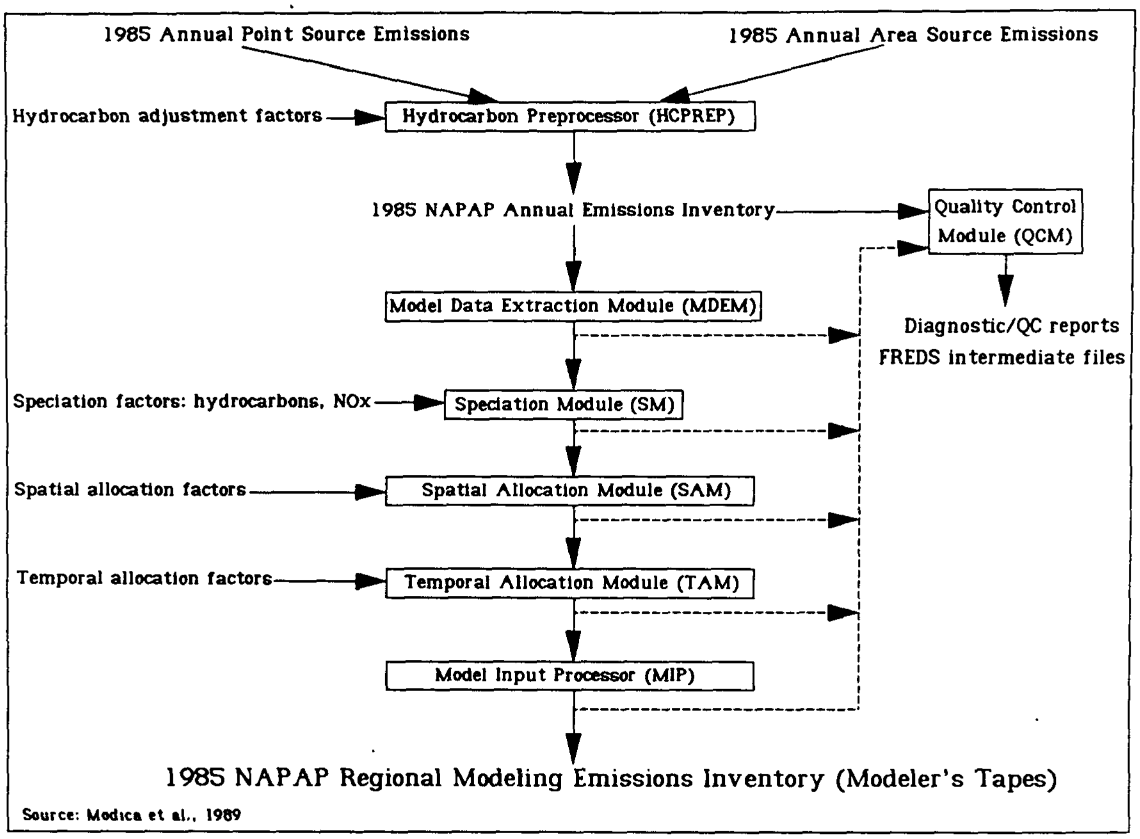

Figure 4.1. Simplified flowchart of FREDS processing for the ROM.

4.1.3 The Speciation Module

After HCPREP and MDEM have been executed, three more modules must be run; these split the annual emissions data into discrete pollutant species, grid cells, and time periods.

The speciation module splits NO_x into NO (expressed as weight of NO_2) and NO_2 and splits the VOCs into specific hydrocarbon classes such as olefins, paraffins, toluene, xylene, ethene (ethylene), formaldehyde, isoprene, higher aldehydes, and nonreactive hydrocarbons.

4.1.4 The Spatial Allocation Module

The Spatial Allocation Module (SAM) allocates emissions data into the ROM grid cells ($^1/_6$ degree latitude by $^1/_4$ degree longitude). For area sources, emissions are merged with spatial allocation factors to disaggregate county level emissions to individual grid cells. The spatial allocation factors are based on Landsat and census data, and take the form of fractional multipliers. Point-source emissions are assigned to grid cells based on latitude and longitude or UTM coordinates.

4.1.5 The Temporal Allocation Module

The Temporal Allocation Module (TAM) resolves the annual emissions into 12 types of days (weekdays, Saturday, and Sunday for each of 4 seasons) and 288 hours (24 hours for each of the 12 types of days). Point-source factors are usually based on plant operating data or on the type of pollutant source. Area-source factors are based on various sources, e.g. climatology (as it affects residential heating combustion), U.S. Department of Transportation records (related to auto emissions), EPA Industrial Fuel use guidelines, etc. For both point and area sources, if no temporal data of any kind are available, uniform factors are assigned to divide the annual emissions equally between seasons, days, and hours.

4.1.6 The Model Input Preprocessor

The Model Input Preprocessor (MIP) accepts fully resolved, area-source or point-source files that include the appropriate spatial, temporal, and speciation factors. The MIP concatenates input files if necessary, and performs the appropriate sorts to yield a complete point- or area-source modeler's file in SAS format. The SAS modeler's tapes are converted to standard ASCII or EBCDIC format prior to input to the ROM.

4.1.7 The Quality Control Module

In addition to the diagnostic checks performed within each of the FREDS modules, data sets at all stages of FREDS processing are examined for consistency by the Quality Control Module (QCM). The QCM accepts any SAS-formatted emissions file and compares emissions totals with values calculated from an earlier stage of FREDS to ensure that pollutant data are not accidentally altered during the apportioning process. The QCM checks national and state level sums for ten major pollutants, as well as national emissions totals for up to 20 user selected source category codes. The QCM also checks the temporal allocation factors to ensure that they sum to unity.

4.2 FREDS DATA OUTPUT

The data tapes that are output from FREDS are input into the ROM emissions preprocessors (Section 5).

SECTION 5

ANTHROPOGENIC EMISSIONS DATA PREPROCESSING

The National Acid Precipitation Assessment Program (NAPAP) raw emissions data for 1980 and 1985, as received by the EPA on magnetic tapes from Alliance Technologies, Inc., require preprocessing prior to input to the ROM processor network. Preprocessing involves spatial and temporal allocations of emissions data from one of three sources: (1) area (or nonmobile), (2) mobile, and (3) point.

The *area-* and *mobile-source* data are already in a format of hourly data (GMT) for three day types[1] (Saturday, Sunday, weekday) for each of the four seasons (winter, spring, summer, fall). Thus, there are three data files for each season, each containing 24 hours of data.

The *point-source* data consist of one file containing annual data and seasonal, daily, and hourly allocation factors. Allocation factors are applied to obtain an hourly-data format that is identical to the format of area- and mobile-source data. These data also contain species allocation factors to be applied to (1) NO_x, to calculate NO and NO_2 concentrations, and (2) total hydrocarbons, to calculate the individual VOC hydrocarbon concentrations (e.g., OLE and PAR).

Sections 5.1 to 5.4 describe how the raw anthropogenic emissions data are manipulated and checked prior to input to the ROM processor network. The anthropogenic emissions preprocessors, as used for the NAPAP 1985 and 2005 data, are alphabetically documented starting in Section 5.5; a chart showing the flow of programs is shown in Figure 5.1. Section 5.16 is pertinent only if you receive data directly from the FREDS database.

---

1. There is an assumption that all Saturdays, Sundays, or weekdays within a season are identical.

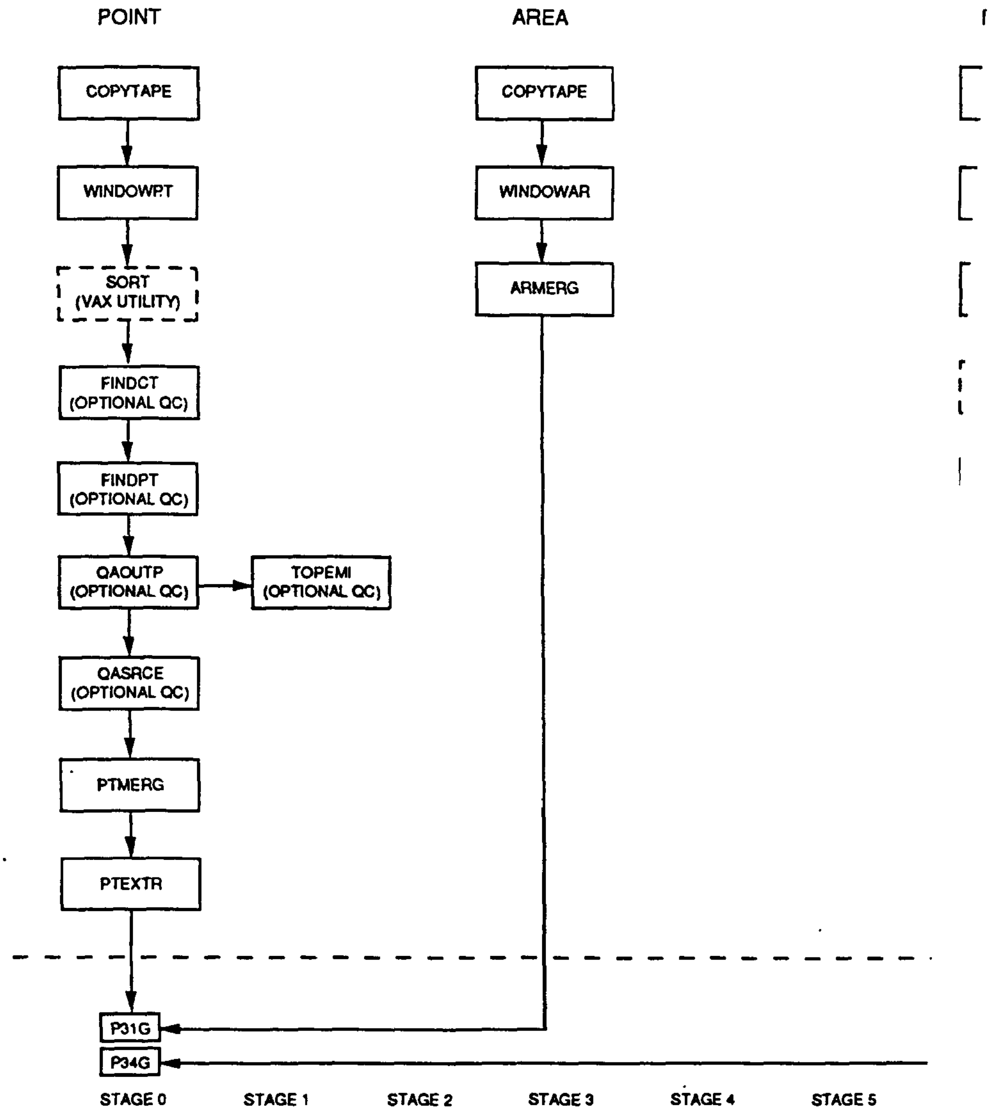

Figure 5.1. Flow of programs for emissions input data. VAX-specific programs are shown in broken-line boxes.

5.1 DATA MERGING AND SORTING

The first step in emissions preprocessing is to copy the raw data tapes to VAX system tapes. This is accomplished using the tape-to-tape copy program COPYTAPE. The data on the tapes usually cover a larger area than the particular region to be modeled, so only the data within the specified region require extraction to the disk. This procedure is designated as "windowing" the necessary data for a given region, and is accomplished by the programs WINDOWPT, WINDOWAR, and WINDOWMB. WINDOWPT is designed to run on the annual point-source data, WINDOWAR on the area-source data, and WINDOWMB on the mobile-source data. *Depending on the record length and block size of the data on tape, these programs may require slight modifications (i.e., changing the RECL and BLKSZE parameters).*

1985 and 2005 Canadian and United States data are received as separate data sets, so the two sets require merging.[2] This is done for the three data types, using three different programs:

- preprocessor PTMERG combines the Canadian and U.S. annual point-source emissions into one file;
- preprocessor MBMERG does the same for the mobile-source emissions; and
- preprocessor ARMERG combines the Canadian and U.S. area-source emissions data, and sums those grid cells that overlap both countries.[3]

The point-source and mobile-source emissions are sorted *after* the merge, using the VAX system's SORT utility:

- mobile-source data are sorted by county identification code and hour;
- point-source data are sorted by stack identification (state/county/plant/point) code.

Area-source data are sorted by hour, column, and row during the execution of preprocessor ARMERG. All sorting facilitates processing during later stages. Section 5.4 contains a table that shows the sequence of preprocessing, sorting, and merging for each of the source types.

5.2 EMISSIONS INVENTORY REDUCTION

The ROM requires NO_x, VOC, and CO as emissions data input. The emissions inventory contains various other unneeded species. To reduce file sizes, these extraneous species are eliminated from the emis-

2. For the 1980 emissions, no merging is necessary; only sorting is required. Also, there are no separate mobile-source emissions for 1980, because they are incorporated into the area emissions.
3. Area-source emissions data reported by both the U.S. and Canada stop at the international border. Thus, summing of the two nations' data is necessary to obtain the correct area-source emissions for grid cells that are bisected by the border.

sions data files. The species remaining are NO, NO_2, CO, ALD2, ETH, FORM, ISOP, OLE, PAR, TOL, XYL, NONR, METH, and MTHL. Abbreviations and descriptions of all chemical species used in the ROM are defined in Table 5.1.

TABLE 5.1. ROM CHEMICAL SPECIES

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

- The reduction in the number of species is accomplished for the mobile- and area-sources by the programs MBEXTR (mobile extractions) and AREXTR (area extractions). The program AREXTR is not required for 1985 or 2005 data (it is run only if using 1980 area source data), since the 1985 and 2005 area-source data are already in a reduced form.
- The extraneous point-source species are eliminated by the program PTEXTR (point extractions). Program PTEXTR also removes the temporal allocation factors for the seasons not currently modeled by the ROM (i.e., winter, spring, and fall), and calculates the allocations for the total hydrocarbon and NO_x species. In addition to these tasks, PTEXTR removes the stack parameters (stack height, plume height, latitude, longitude, etc.) from each individual record and places these parameters in a separate file. Again, this is done to aid in later processing.

This completes the raw emissions preprocessing stage. At this point, the area- and mobile-source emissions are hour-explicit day-type data (e.g., weekday, 17:00 LST, summer). The point-source emissions are annual data with seasonal, daily, and hourly allocation factors for each species. For example, a point *source* may have an annual SO_2 emission of 64.42 tons, with a *summer* factor of 0.25, a *weekday* factor of 0.011, and an *hourly* factor of 0.042. The hourly data that are input to ROM consist of the product of these factors and the annual emission, i.e., (64.42 x 0.25 x 0.011 x 0.042).

5.3 QUALITY CONTROL

Anthropogenic emissions processing involves several quality control preprocessors that are used to identify erroneous or missing data. Most of this processing is performed on the point-source emissions because these are where the majority of errors are likely to occur. We thoroughly check the annual point source data for (1) missing stack parameters (stack height, temperature, diameter, flow rate, etc.), (2) those parameters that appear unrealistic (e.g., a plume rise of 10,000 feet or flow rate of 300 $m \cdot s^{-1}$), and (3) erroneous emissions data values; if one point far exceeds all of the points in the region, it is likely that this point is in error; points that are high in NO_x emissions should not generally be high in VOC emissions. The quality control programs for point-source data (FINDCT, FINDPT, QAOUTP, QASRCE, QASTACK, and TOPEMI) are used to catch these types of errors.

The area- and mobile-source emissions data errors are usually found as mistakes in the emissions magnitude, or as spatial or temporal allocation errors. Color tiling of these data on an hourly basis highlights inconsistencies in these areas; e.g., all of Vermont registering emissions that are ten times higher than surrounding states implies a spatial allocation error, while higher emissions in a cell at 2 a.m. than at midday reveals a temporal allocation error.

5.4 SUMMARY FOR 1985 AND 2005 NAPAP DATA

Table 5.2 summarizes the flow of programs that read U.S. and Canadian anthropogenic emissions data, perform quality control procedures on the point-source data, and produce the final anthropogenic emissions data that are input to the ROM. Note that preprocessor AREXTR is not required for 1985 and 2005 data.

TABLE 5.2. PREPROCESSOR PROGRAMS FOR ANTHROPOGENIC EMISSIONS DATA

United States Data				Canadian Data			
PROGRAM (.FOR)	INPUT	OUTPUT	COMMENTS	PROGRAM (.FOR)	INPUT	OUTPUT	COMMENTS
COPYTAPE	MAGNETIC TAPE	VAX SYSTEM TAPE	tape backup	COPYTAPE	MAGNETIC TAPE	VAX SYSTEM TAPE	tape backup
WINDOWPT	VAX SYSTEM TAPE	USPOINTZ.RAW		WINDOWPT	VAX SYSTEM TAPE	CAPOINTZ.RAW	
WINDOWAR	VAX SYSTEM TAPE	USARSAZ.RAW USARSUZ.RAW USARWKZ.RAW		WINDOWAR	VAX SYSTEM TAPE	CAARSAZ.RAW CAARSUZ.RAW CAARWKZ.RAW	
WINDOWMB	VAX SYSTEM TAPE	USMBSAZ.RAW USMBSUZ.RAW USMBWKZ.RAW		WINDOWMB	VAX SYSTEM TAPE	CAMBSAZ.RAW CAMBSUZ.RAW CAMBWKZ.RAW	

PROGRAM (.FOR)	INPUT	OUTPUT	COMMENTS
PTMERG	USPOINTZ.RAW; CAPOINTZ.RAW	POINTZ.RAW	
SORT	POINTZ.RAW	POINTZ.DAT	VAX system sort
FINDCT FINDPT QAOUTP QASRCE QASTACK TOPEMI	POINTZ.DAT	various reports	optional quality control
ARMERG	USARSAZ.RAW USARSUZ.RAW USARWKZ.RAW CAARSAZ.RAW CAARSUZ.RAW CAARWKZ.RAW	ARSAZ.DAT ARSUZ.DAT ARWKZ.DAT	also sorts data
MBMERG	USMBSAZ.RAW USMBSUZ.RAW USMBWKZ.RAW CAMBSAZ.RAW CAMBSUZ.RAW CAMBWKZ.RAW	MBSAZ.RAW MBSUZ.RAW MBWKZ.RAW	
SORT	MBSAZ.RAW MBSUZ.RAW MBWKZ.RAW	MBSAZ.DAT MBSUZ.DAT MBWKZ.DAT	VAX system sort
PTEXTR	POINTZ.DAT	SEANPTZ.CS*x* STACKS.CS*x*	*x* is variable
MBEXTR	MBSAZ.DAT MBSUZ.DAT MBWKZ.DAT	MOBLSAZ.CS*x* MOBLSUZ.CS*x* MOBLWKZ.CS*x*	*x* is variable

5.5 PREPROCESSOR ARMERG (AREA-SOURCE EMISSIONS)

I. PREPROCESSOR FUNCTION

The area-source emissions inventory on disk is generated from VAX system tapes by WINDOWAR and exists as three files of U.S. data and three files of Canadian data. This preprocessor combines the Canadian data and the U.S. data files following a VAX system sort command.

II. I/O COMPONENTS (using summer 1985 data as an example)

A. Input Files

1. Standard Input Files

   None

2. Nonstandard Input Files

   a. The three Canadian data sets that are generated by WINDOWAR are used as input to ARMERG. The READ and FORMAT statements for the files are listed below, and the files' parameters are shown in Table 5.3.

```
      READ(UNIT2, 4005, IOSTAT=IOST2)CCOL, CROW, CHOUR,
     &                 (CANEM(ISPC), ISPC = 1,17)
4005  FORMAT(2I4, I2, 17(E10.3))
```

   b. The three U.S. data sets that are generated by WINDOWAR are also input to ARMERG. The READ and FORMAT statements for the files are listed below, and the files' parameters are shown in Table 5.4.

```
      READ(UNIT1, 1005, IOSTAT=IOST)ICOL, IROW, IHOUR,
     &                 (ANEM(ISPC), ISPC = 1,17)
1005  FORMAT(2I4, I2, 17(E10.3))
```

TABLE 5.3. CANADIAN AREA-SOURCE FILE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CCOL		Integer*4	Grid column number
2	CROW		Integer*4	Grid row number
3	CHOUR	GMT	Integer*4	Hour as *HH*
4	$CANEM(1)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO_x emissions
5	$CANEM(2)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	VOC emissions
6	$CANEM(3)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	THC emissions
7	$CANEM(4)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	CO emissions
8	$CANEM(5)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Olefin emissions
9	$CANEM(6)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Paraffin emissions
10	$CANEM(7)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Toluene emissions
11	$CANEM(8)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Xylene emissions
12	$CANEM(9)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Formaldehyde emissions
13	$CANEM(10)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Other aldehydes emissions
14	$CANEM(11)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Ethylene emissions
15	$CANEM(12)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Isoprene emissions
16	$CANEM(13)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Nonreactive THC emissions
17	$CANEM(14)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Methane emissions
18	$CANEM(15)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO emissions
19	$CANEM(16)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO_2 emissions
20	$CANEM(17)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Methanol emissions

Note: i = column number; j = row number; h = hour. Units (tons) are English short tons. NO emissions are reported as weight of NO_2.

TABLE 5.4. U.S. AREA-SOURCE FILE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	ICOL		Integer*4	Grid column number
2	IROW		Integer*4	Grid row number
3	IHOUR	GMT	Integer*4	Hour as *HH*
4	$ANEM(1)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO_x emissions
5	$ANEM(2)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	VOC emissions
6	$ANEM(3)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	THC emissions
7	$ANEM(4)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	CO emissions
8	$ANEM(5)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Olefin emissions
9	$ANEM(6)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Paraffin emissions
10	$ANEM(7)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Toluene emissions
11	$ANEM(8)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Xylene emissions
12	$ANEM(9)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Formaldehyde emissions
13	$ANEM(10)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Other aldehydes emissions
14	$ANEM(11)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Ethylene emissions
15	$ANEM(12)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Isoprene emissions
16	$ANEM(13)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Nonreactive THC emissions
17	$ANEM(14)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Methane emissions
18	$ANEM(15)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO emissions
19	$ANEM(16)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO_2 emissions
20	$ANEM(17)_{i,j,h}$	$mol \cdot h^{-1}$	Real*4	Methanol emissions

Note: i = column number; j = row number; h = hour. Units (tons) are English short tons. NO emissions are reported as weight of NO_2.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

ARMERG generates three sequential disk files. If no future inventory-reduction process is needed, these output files are named AREAWKZ.CS0, AREASAZ.CS0, and AREASUZ.CS0, and are the input files for processor P34G. In the event that the area-source emissions files contain extraneous species that must be eliminated prior to input into P34G, the output files produced by ARMERG are named ARWKZ.DAT, ARSAZ.DAT and ARSUZ.DAT. This latter case occurred, e.g., for the 1980 emissions inventory. Each record of the output files is produced by the following WRITE and FORMAT statements, and the files' parameters are shown in Table 5.5.

```
      WRITE (UNIT3, 6005, IOSTAT=IOST) M, N, OLDHR,
                          (OUTPUT(I, J, K) K=1, 17)
6005 FORMAT (I4, I4, I2, 17(0PE10.3))
```

TABLE 5.5. MERGED AREA-SOURCE FILE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	M		Integer*4	Grid column number + 160
2	N		Integer*4	Grid row number + 52
3	$OLDHR_{i,j,s}$	GMT	Integer*4	Hour as *HH*
4	OUTPUT(1)	$tons{\cdot}h^{-1}$	Real*4	NO_x emissions
5	OUTPUT(2)	$tons{\cdot}h^{-1}$	Real*4	VOC emissions
6	OUTPUT(3)	$tons{\cdot}h^{-1}$	Real*4	THC emissions
7	OUTPUT(4)	$tons{\cdot}h^{-1}$	Real*4	CO emissions
8	OUTPUT(5)	$mol{\cdot}h^{-1}$	Real*4	Olefin emissions
9	OUTPUT(6)	$mol{\cdot}h^{-1}$	Real*4	Paraffin emissions
10	OUTPUT(7)	$mol{\cdot}h^{-1}$	Real*4	Toluene emissions
11	OUTPUT(8)	$mol{\cdot}h^{-1}$	Real*4	Xylene emissions
12	OUTPUT(9)	$mol{\cdot}h^{-1}$	Real*4	Formaldehyde emissions
13	OUTPUT(10)	$mol{\cdot}h^{-1}$	Real*4	Other aldehydes emissions
14	OUTPUT(11)	$mol{\cdot}h^{-1}$	Real*4	Ethylene emissions
15	OUTPUT(12)	$mol{\cdot}h^{-1}$	Real*4	Isoprene emissions
16	OUTPUT(13)	$mol{\cdot}h^{-1}$	Real*4	Nonreactive emissions
17	OUTPUT(14)	$mol{\cdot}h^{-1}$	Real*4	Methane emissions
18	OUTPUT(15)	$tons{\cdot}h^{-1}$	Real*4	NO emissions
19	OUTPUT(16)	$tons{\cdot}h^{-1}$	Real*4	NO_2 emissions
20	OUTPUT(17)	$mol{\cdot}h^{-1}$	Real*4	Methanol emissions

Note: i = column number; j = row number; s = species. Units (tons) are English short tons. NO emissions are reported as weight of NO_2.

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	14	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	8	blocks
Executable file:	1	file	14	blocks
	3	files	36	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:31:32
Buffered I/O count:	1725
Direct I/O count:	17719
Peak working-set size:	14328
Peak virtual size:	18011

C. Space Requirements: Log and Print Files

ARMERG.LOG:	37 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.6 shows the input file and output file space requirements for summer 1985 data; note that these numbers are subject to change.

TABLE 5.6. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Before system sort				
Input	USARWKZ.RAW	Nonstd	19017	24 hours
	USARSAZ.RAW	Nonstd	19017	24 hours
	USARSUZ.RAW	Nonstd	22617	24 hours
	CAARWKZ.RAW	Nonstd	2261	24 hours
	CAARSAZ.RAW	Nonstd	2261	24 hours
	CAARSUZ.RAW	Nonstd	2261	24 hours
			67434	
After system sort				
Input	USARWKZ.DAT	Nonstd	19017	24 hours
	USARSAZ.DAT	Nonstd	19017	24 hours
	USARSUZ.DAT	Nonstd	22617	24 hours
	CAARWKZ.DAT	Nonstd	2261	24 hours
	CAARSAZ.DAT	Nonstd	2261	24 hours
	CAARSUZ.DAT	Nonstd	2261	24 hours
			67434	
Output	AREAWKZ.CS0	Nonstd	28392	24 hours
	AREASAZ.CS0	Nonstd	28392	24 hours
	AREASUZ.CS0	Nonstd	28392	24 hours
			85176	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

ARMERG.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE MET53:[EMISSIONS.SOURCE]ARMERG.LNK
$ ON WARNING THEN EXIT
$ EXEFILE = "MET53:[EMISSIONS.SOURCE]ARMERG.EXE"
$ ASSIGN/USER_MODE MET53:[EMISSIONS.SOURCE]ARMERG.OBJ          A1
$ ASSIGN/USER_MODE MET24:[ROMOBJ.UTILIO]JFILEX.OBJ             A2
$ ASSIGN/USER_MODE MET24:[ROMOBJ.UTILIO]IOCL.OBJ               A3
$ ASSIGN/USER_MODE MET24:[ROMOBJ.UTILIO]ADATE.OBJ              A4
$ ASSIGN/USER_MODE MET24:[ROMOBJ.UTILIO]JUNIT.OBJ              A5
$ LINK -
$ !  /DEBUG-
  /EXECUTABLE = MET53:[EMISSIONS.SOURCE]ARMERG.EXE -
  A1, -
  A2, -
  A3, -
  A4, -
  A5
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

ARMERG.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS $PRINT
$ !
$ ! RUN STREAM IS IN MET53:[EMISSIONS.SOURCE]ARMERG.COM
$ !
$ !  ************** SYSTEM SORT FILES BY HOUR **************
$ !
$ ASSIGN MET14: SORTWORK0
$ ASSIGN MET16: SORTWORK1
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARWKZ.RAW INFILE
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARWKZ.DAT OUTFILE
$ SORT/KEY=(POSITION=9,SIZE=2) -
  INFILE -
$ DIR/FULL INFILE
$ DIR/FULL OUTFILE
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARSAZ.RAW INFILE
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARSAZ.DAT OUTFILE
$ SORT/KEY=(POSITION=9,SIZE=2) -
  INFILE -
$ DIR/FULL INFILE
$ DIR/FULL OUTFILE
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARSUZ.RAW INFILE
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARSUZ.DAT OUTFILE
$ SORT/KEY=(POSITION=9,SIZE=2) -
  INFILE -
$ DIR/FULL INFILE
$ DIR/FULL OUTFILE
$ !
```

```
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]CAARWKZ.RAW INFILE
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]CAARWKZ.DAT OUTFILE
$ SORT/KEY=(POSITION=9,SIZE=2) -
  INFILE -
$ DIR/FULL INFILE
$ DIR/FULL OUTFILE
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]CAARSAZ.RAW INFILE
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]CAARSAZ.DAT OUTFILE
$ SORT/KEY=(POSITION=9,SIZE=2) -
  INFILE -
$ DIR/FULL INFILE
$ DIR/FULL OUTFILE
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]CAARSUZ.RAW INFILE
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]CAARSUZ.DAT OUTFILE
$ SORT/KEY=(POSITION=9,SIZE=2) -
  INFILE -
$ DIR/FULL INFILE
$ DIR/FULL OUTFILE
$ !
$ !
$ !  ********** MERGE SORTED U.S. AND CANADIAN FILES **********
$ !
$ ! assign USA sorted area input files
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARWKZ.DAT       USARWKD
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARSAZ.DAT       USARSAT
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARSUZ.DAT       USARSUN
$ !
$ ! assign Canada sorted area input files
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]CAARWKZ.DAT       CAARWKD
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]CAARSAZ.DAT       CAARSAT
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]CAARSUZ.DAT       CAARSUN
$ !
$ ! assign merged area output files
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]AREAWKZ.CSO       ARWKD
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]AREASAZ.CSO       ARSAT
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]AREASUZ.CSO       ARSUN
$ !
$ RUN MET53:[EMISSIONS.SOURCE]ARMERG.EXE
```

(end of run stream)

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

A. Main Program

ARMERG

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

JFILEX

B. Functions

DISK$VAXSET:[ROMLIB.ROMNET]

ADATE
IOCL
JUNIT

IX. INCLUDE FILES

None

5.6 PREPROCESSOR COPYTAPE (DATA TAPE TO VAX-SYSTEM TAPE COPY)

I. PREPROCESSOR FUNCTION

This preprocessor copies a raw data tape onto a VAX-system tape in order to ensure that valuable data are not lost. You need to understand the limitations of this preprocessor. One feature of this program is that a parity error will not stop the copying of a tape - the bad tape block on the original will not be written to the copy. However, no message stating that this has occurred is written to the log file. Therefore, we recommend that you use a scanning routine to compare the data on the original tape to the data on the copy. If such a routine is unavailable, you should monitor the error counts on the tape drives. Although an increase in the error count does not necessarily indicate a bad copy, no increase in the error count guarantees a good copy.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

Any ASCII tape can be used as input.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

The output is written to a VAX-system tape. The program is hard-coded to skip the label of the system tape.

III. CONTROL CARDS

One control card is used in the format shown below. Table 5.7 defines the control card variable.

IFILE

TABLE 5.7. CONTROL CARD VARIABLE

Variable Name	Unit	Description
IFILE		Number of files to be copied; this number represents the minimum number of files to be copied. A number greater than the actual number of files on the original tape guarantees that the entire original tape is copied.

Example:

99

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	12	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	2	files	9	blocks
Executable file:	1	file	9	blocks
	5	files	30	blocks

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

	VAX 8650
Charged CPU time (hh:mm:ss):	± 00:00:15 per tape
Buffered I/O count:	392
Direct I/O count:	27219
Peak working-set size:	530
Peak virtual size:	4564

C. Space Requirements: Log and Print Files

COPYTAPE.LOG	24 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

None

E. Space Requirements: Tape Files

Two tapes are required, an original and a blank VAX-system tape. Table 5.8 shows the input file and output file space requirements for a representative tape.

TABLE 5.8. TAPE FILE SPACE REQUIREMENTS

File Group	Tape Number	File Number	Record Length (bytes)	Block Size (bytes)
Input	B20012	1	495	24750
Output	102479	1 (label)	80	80
		2	495	24750

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

COPYTAPE.LNK follows:

```
$ ASSIGN USER2$DISK:[DAO.METTAPE]COPYTAPE.OBJ                A1
$ ASSIGN USER2$DISK:[DAO.METTAPE]VATAPE.OBJ                  A2
$ LINK -
  /EXECUTABLE = USER2$DISK:[DAO.METTAPE]COPYTAPE.EXE -
  /MAP = MET1:[S01S]COPYTAPE.MAP -
  A1, -
  A2
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

COPYTAPE.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ TAPE SELECT ADRIVE BTAPE
$ TAPE LOAD ADRIVE B20012
$ MOUNT/NOWRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=12000  -
  ADRIVE B20012 OLDTAP:
$ TAPE SELECT BDRIVE 102479
$ TAPE LOAD/RING BDRIVE 102479
$ MOUNT/WRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=12000  -
  BDRIVE 102479 COPTAP:
$ SHOW DEV/FULL ADRIVE
$ SHOW DEV/FULL BDRIVE
$ SHOW LOGICAL OLDTAP
$ SHOW LOGICAL COPTAP
$ ! SET MAGTAPE/SKIP=FILES:1 OLDTAP:      ! Use this line only if
$ !                                       ! old tape has label
$ RUN USER2$DISK:[DAO.METTAPE]COPYTAPE.EXE
  99
$ RUN USER2$DISK:[BRO]TSCAN.EXE
ADRIVE
$ RUN USER2$DISK:[BRO]TSCAN.EXE
BDRIVE
$ @REWIND OLDTAP:
$ @REWIND COPTAP:
$ SHOW DEV/FULL ADRIVE
$ SHOW DEV/FULL BDRIVE
$ DISMOUNT OLDTAP:
$ DEALLOCATE ADRIVE
$ DISMOUNT COPTAP:
$ DEALLOCATE TAPEDRIVE
$ DEALLOCATE BDRIVE
```

(end of run stream)

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

A. Main Program

COPYTAPE

B. Subroutines

USER2$DISK:[DAO.METTAPE]

VATAPE

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

5.7 PREPROCESSOR FINDCT (POINT-SOURCE EMISSIONS QUALITY CONTROL)

I. PREPROCESSOR FUNCTION

This preprocessor extracts all point-source records within a county that you specify. FINDCT is used for quality control purposes only; the extracted records are written to the log file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

POINTZ.DAT is a sequential data file that contains the annual point-source emissions data; running the VAX SORT utility on the point-source data. Each record is read as a single string of characters using the following READ and FORMAT statements, and the file parameter is shown in Table 5.9.

```
    READ(INDEV,100,IOSTAT = IOSTR) LINE
100 FORMAT(A2148)
```

TABLE 5.9. POINTZ.DAT PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$LINE_p$		Char*2148	Annual emissions record for point source p

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 5.10 defines the control card variable.

TARGET

TABLE 5.10. CONTROL CARD VARIABLE

Variable Name	Unit	Description
TARGET		AEROS state/county identification code

Note: 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.

Example:

184140

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	3	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	3	blocks
Executable file:	1	file	21	blocks
	3	files	27	blocks

B. Execution Time Requirements (Representative Values for a Single County)

	VAX 8650
Charged CPU time (hh:mm:ss):	00:01:02
Buffered I/O count:	197
Direct I/O count:	4535
Peak working-set size:	603
Peak virtual size:	3755

C. Space Requirements: Log and Print Files

FINDCT.LOG:	69 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.11 shows the input file and output file space requirements.

TABLE 5.11. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	POINTZ.DAT	Nonstd	94065	annual
Output	none		0	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

FINDCT.LNK follows:

```
$ LINK FINDCT, -
  USER2$DISK:[ROMNET.UTILIO]JFILE2, -
  USER2$DISK:[ROMNET.UTILIO]JUNIT, -
  USER2$DISK:[ROMNET.UTILIO]JFILE3, -
  USER2$DISK:[ROMNET.UTILIO]JFILE4, -
  USER2$DISK:[ROMNET.UTILIO]IOCL, -
  USER2$DISK:[ROMNET.UTILIO]PROGID, -
  USER2$DISK:[ROMNET.UTILIO]ADATE
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

FINDCT.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ !
$ ! RUN STREAM IS IN [WAY.ROM21.EMISQA.1985]FINDCT.COM
$ !
$ ASSIGN MET53:[EMISSIONS.RNSUM85P]POINTZ.DAT INFILE
$ !
$ ! Control card: AEROS state/county code
$ !
$ RUN USER2$DISK:[WAY.ROM21.EMISQA.1985]FINDCT.EXE
184140
```

(end of run stream)

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

A. Main Program

FINDCT

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
JFILE4
PROGID

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JFILE3
JUNIT

IX. INCLUDE FILES

None

5.8 PREPROCESSOR FINDPT (POINT-SOURCE EMISSIONS QUALITY CONTROL)

I. PREPROCESSOR FUNCTION

This preprocessor extracts an individual point-source record that you specify. FINDPT is used for quality control purposes only; the extracted record is written to the log file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

POINTZ.DAT is a sequential data file that contains the annual point-source emissions data; it is generated by running the VAX SORT utility on the point-source data. Each record is read as a single string of characters using the following READ and FORMAT statements, and the file parameter is shown in Table 5.12.

```
     READ(INDEV,100,IOSTAT = IOSTR) LINE
100  FORMAT(A2148)
```

TABLE 5.12. POINTZ.DAT PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$LINE_p$		Char*2148	Annual emissions record for point source *p*

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 5.13 defines the control card variable.

TARGET

TABLE 5.13. CONTROL CARD VARIABLE

Variable Name	Unit	Description
TARGET		AEROS state/county/plant/point identification code

Note: 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.

Example:

184140000907

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	3	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	3	blocks
Executable file:	1	file	21	blocks
	3	files	27	blocks

B. Execution Time Requirements (Representative Values for a Single County)

	VAX 8650
Charged CPU time (hh:mm:ss):	00:00:03
Buffered I/O count:	175
Direct I/O count:	186
Peak working-set size:	514
Peak virtual size:	2321

C. Space Requirements: Log and Print Files

FINDPT.LOG:	9 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.14 shows the input file and output file space requirements.

TABLE 5.14. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	POINTZ.DAT	Nonstd	94065	annual
Output	none		0	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

FINDPT.LNK follows:

```
$ LINK FINDPT, -
  USER2$DISK:[ROMNET.UTILIO]JFILE2, -
  USER2$DISK:[ROMNET.UTILIO]JUNIT, -
  USER2$DISK:[ROMNET.UTILIO]JFILE3, -
  USER2$DISK:[ROMNET.UTILIO]JFILE4, -
  USER2$DISK:[ROMNET.UTILIO]IOCL, -
  USER2$DISK:[ROMNET.UTILIO]PROGID, -
  USER2$DISK:[ROMNET.UTILIO]ADATE
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

FINDPT.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ !
$ ! RUN STREAM IS IN [WAY.ROM21.EMISQA.1985]FINDPT.COM
$ !
$ ASSIGN MET53:[EMISSIONS.RNSUM85P]POINTZ.DAT INFILE
$ !
$ ! Control card: AEROS state/county/plant/point code
$ !
$ RUN USER2$DISK:[WAY.ROM21.EMISQA.1985]FINDPT.EXE
184140000907
```

(end of run stream)

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

A. Main Program

FINDPT

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
JFILE4
PROGID

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JFILE3
JUNIT

IX. INCLUDE FILES

None

5.9 PREPROCESSOR MBEXTR (MOBILE-SOURCE EMISSIONS)

I. PREPROCESSOR FUNCTION

MBEXTR reads the raw mobile-source day type (Saturday, Sunday, and weekday) emissions data files; it extracts from these files only those chemical species required by the ROM. The program creates new data files for each of the three day types.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

MBEXTR reads three nonstandard files: MBSAZ, MBSUZ, and MBWKZ. These three files contain the raw 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 5.15.

```
      READ(UNIT1, 1005, IOSTAT=IOST) ICOL, IROW, IHOUR,
     &    (ANEM(ISPC), ISPC=1,6), NO, NO2, CNTYCODE
1005  FORMAT(2I4, I2, 10X, 6(E10.3), 100X, 2(E10.3), 10X, A6)
```

TABLE 5.15. MBSAZ, MBSUZ, AND MBWKZ 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	IHOUR	GMT	Integer*4	Hour
4	ANEM(1)	tons·h^{-1}	Real*4	Mobile VOC emissions
5	ANEM(2)	tons·h^{-1}	Real*4	Mobile THC emissions
6	ANEM(3)	tons·h^{-1}	Real*4	Mobile CO emissions
7	ANEM(4)	tons·h^{-1}	Real*4	Evaporative emissions of THC
8	ANEM(5)	tons·h^{-1}	Real*4	Gasoline exhaust emissions of THC
9	ANEM(6)	tons·h^{-1}	Real*4	Diesel exhaust emissions of THC
10	NO	tons·h^{-1}	Real*4	Mobile NO emissions
11	NO_2	tons·h^{-1}	Real*4	Mobile NO_2 emissions
12	CNTYCODE		Char*6	AEROS State/County code [b]

a. Units are in English short tons·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. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

MBEXTR produces three nonstandard output files called MOBLSASZ, MOBLSUZ, and MOBLWKZ. These three files each contain 24 hours of hourly mobile-source emissions data for five species: CO, NO, NO_2, VOC, and THC. Note that the hour parameter remains as Greenwich Mean Time (GMT). The WRITE and FORMAT statements used to write these files follow, and the files' parameters are shown in Table 5.16.

```
      WRITE(UNIT2, 2005, IOSTAT=IOST) ICOL, IROW, IHOUR, ANEM(3), NO, NO2,
     &      ANEM(4), ANEM(5), ANEM(6), ANEM(1), ANEM(2), CNTYCODE
 2005  FORMAT(2I4, I2, 8(1PE10.3), A6)
```

TABLE 5.16. MBSAZ, MBSUZ, AND MBWKZ 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	IHOUR	GMT	Integer*4	Hour
4	ANEM(3)	$tons{\cdot}h^{-1}$	Real*4	Mobile CO emissions
5	NO	$tons{\cdot}h^{-1}$	Real*4	Mobile NO emissions
6	NO_2	$tons{\cdot}h^{-1}$	Real*4	Mobile NO_2 emissions
7	ANEM(4)	$tons{\cdot}h^{-1}$	Real*4	Evaporative emissions of THC
8	ANEM(5)	$tons{\cdot}h^{-1}$	Real*4	Gasoline exhaust emissions of THC
9	ANEM(6)	$tons{\cdot}h^{-1}$	Real*4	Diesel exhaust emissions of THC
10	ANEM(1)	$tons{\cdot}h^{-1}$	Real*4	Mobile VOC emissions
11	ANEM(2)	$tons{\cdot}h^{-1}$	Real*4	Mobile THC emissions
12	CNTYCODE		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.

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	8	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	6	blocks
Executable file:	1	file	16	blocks
	3	files	30	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:30:13
Buffered I/O count:	1156
Direct I/O count:	17529
Peak working-set size:	542
Peak virtual size:	2365

C. Space Requirements: Log and Print Files

MBEXTR.LOG:	7 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.17 shows the input file and output file space requirements.

TABLE 5.17. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	MBSAZ.DAT	Nonstd	57195	24 hours
	MBSUZ.DAT	Nonstd	57195	24 hours
	MBWKZ.DAT	Nonstd	57195	24 hours
			171585	
Output	MOBLSAZ.CS0	Nonstd	26076	24 hours
	MOBLSUZ.CS0	Nonstd	26076	24 hours
	MOBLWKZ.CS0	Nonstd	26076	24 hours
			78228	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

MBEXTR.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER62$DISK:[FUG.MOBILE.PROCS]MBEXTR.LNK
$ ON WARNING THEN EXIT
$ EXEFILE = "USER62$DISK:[FUG.MOBILE.PROCS]MBEXTR.EXE"
$ ASSIGN/USER_MODE USER62$DISK:[FUG.MOBILE.PROCS]MBEXTR.OBJ        A1
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JFILEX.OBJ            A2
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JFILE2.OBJ            A3
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]IOCL.OBJ              A4
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JUNIT.OBJ             A5
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]PROGID.OBJ            A6
```

```
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]ADATE.OBJ              A7
$ LINK -
  /EXECUTABLE = USER62$DISK:[FUG.MOBILE.PROCS]MBEXTR.EXE -
  A1, -
  A2, -
  A3, -
  A4, -
  A5, -
  A6, -
  A7
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

MBEXTR.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! run stream is in USER62$DISK:[FUG.MOBILE.PROCS]MBEXTR.COM
$ ASSIGN MLAB: SYS$PRINT
$ !assign mobile input files
$ !
$ ASSIGN WORK_SCRATCH:MBSAZ.DAT          MOSATZ
$ ASSIGN WORK_SCRATCH:MBSUZ.DAT          MOSUNZ
$ ASSIGN WORK_SCRATCH:MBWKZ.DAT          MOWKDZ
$ !assign mobile output
$ !
$ ASSIGN WORK_SCRATCH:MOBLSAZ.CSO        MOSAT
$ ASSIGN WORK_SCRATCH:MOBLSUZ.CSO        MOSUN
$ ASSIGN WORK_SCRATCH:MOBLWKZ.CSO        MOWKD
$ !
$ RUN USER62$DISK:[FUG.PREP.MOBILE]MBEXTR.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

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

A. Main Program

MBEXTR

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

USER2$DISK:[ROMLIB.UTILIO]

JFILEX
PROGID

B. Functions

USER2$DISK:[ROMLIB.UTILIO]

JFILE2

IX. INCLUDE FILES

None

5.10 PREPROCESSOR MBMERGE (MOBILE-SOURCE EMISSIONS)

I. PREPROCESSOR FUNCTION

The mobile-source emissions inventory on disk is generated from VAX system tapes by WINDOWMB, and exists as three files of U.S. data and three files of Canadian data. This preprocessor appends the Canadian data to the U.S. data files; the resultant output files maintain their U.S. designations.

II. I/O COMPONENTS (using summer 1985 data as an example)

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

The three Canadian data sets that are generated by WINDOWMB (CAMBWKZ.RAW, CAMBSAZ.RAW, AND CAMBSUZ.RAW) are used as input to MBMERGE. The READ and FORMAT statements for the files are listed below, and the files' parameters are shown in Table 5.18.

```
      READ(UNIT2, 4005, IOSTAT=IOST2)CCOL, CROW, CHOUR,
     &                 (CANEM(ISPC), ISPC = 1,20)
 4005 FORMAT(2I4, I2, 20(E10.3))
```

TABLE 5.18. CANADIAN MOBILE-SOURCE FILE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CCOL		Integer*4	Grid column number
2	CROW		Integer*4	Grid row number
3	CHOUR	GMT	Integer*4	Hour as *HH*
4	$CANEM(1)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO_x emissions
5	$CANEM(2)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	VOC emissions
6	$CANEM(3)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	THC emissions
7	$CANEM(4)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	CO emissions
8	$CANEM(5)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Evaporative emissions
9	$CANEM(6)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Gasoline vehicle exhaust
10	$CANEM(7)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Diesel vehicle exhaust
11	$CANEM(8)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Olefin emissions
12	$CANEM(9)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Paraffin emissions
13	$CANEM(10)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Toluene emissions
14	$CANEM(11)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Xylene emissions
15	$CANEM(12)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Formaldehyde emissions
16	$CANEM(13)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Other aldehydes emissions
17	$CANEM(14)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Ethylene emissions
18	$CANEM(15)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Isoprene emissions
19	$CANEM(16)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Nonreactive HC emissions
20	$CANEM(17)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Methane emissions
21	$CANEM(18)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO emissions
22	$CANEM(19)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO_2 emissions
23	$CANEM(20)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Methanol emissions

Note: i = column number; j = row number; h = hour. Units are English short tons, HC = hydrocarbon.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

MBMERGE does not generate any independent output files, but the preprocessor produces new USMBWKZ.RAW, USMBSAZ.RAW, and USMBSUZ.RAW files. Each file contains 24 hours of U.S. and Canadian mobile-source data expressed in GMT, and each is produced by the following WRITE and FORMAT statements:

```
      WRITE(UNIT1, 3005, IOSTAT = IOST3)CCOL, CROW, CHOUR,
     &             (CANEM(ISPC), ISPC = 1, 20), CNTYCODE
3005  FORMAT(2I4, I2, 20(OPE10.3), A6)
```

The preprocessor assigns a state/county code of 750000 to all Canadian data. The files' parameters are shown in Table 5.19.

TABLE 5.19. MERGED MOBILE-SOURCE DATA FILE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	CCOL		Integer*4	Grid column number
2	CROW		Integer*4	Grid row number
3	CHOUR	GMT	Integer*4	Hour as *HH*
4	$CANEM(1)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO_x emissions
5	$CANEM(2)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	VOC emissions
6	$CANEM(3)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	THC emissions
7	$CANEM(4)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	CO emissions
8	$CANEM(5)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Evaporative emissions
9	$CANEM(6)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Gasoline vehicle exhaust
10	$CANEM(7)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Diesel vehicle exhaust
11	$CANEM(8)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Olefin emissions
12	$CANEM(9)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Paraffin emissions
13	$CANEM(10)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Toluene emissions
14	$CANEM(11)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Xylene emissions
15	$CANEM(12)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Formaldehyde emissions
16	$CANEM(13)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Other aldehydes emissions
17	$CANEM(14)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Ethylene emissions
18	$CANEM(15)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Isoprene emissions
19	$CANEM(16)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Nonreactive HC emissions
20	$CANEM(17)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Methane emissions
21	$CANEM(18)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO emissions
22	$CANEM(19)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	NO_2 emissions
23	$CANEM(20)_{i,j,h}$	$tons \cdot h^{-1}$	Real*4	Methanol emissions
24	$CNTYCODE_{i,j}$		Char*6	AEROS state/county code

i = column number; *j* = row number; *h* = hour; units are English tons; HC = hydrocarbon. NO reported as weight of NO_2. For information on AEROS, refer to *AEROS Manual of Codes*, EPA-450/2-76-005a.

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	5	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	5	blocks
Executable file:	1	file	11	blocks
	3	files	23	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:02:06
Buffered I/O count:	272
Direct I/O count:	1012
Peak working-set size:	542
Peak virtual size:	2361

C. Space Requirements: Log and Print Files

MBMERGE.LOG:	6 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.20 shows the input file and output file space requirements for summer 1985 data; note that these numbers are subject to change.

TABLE 5.20. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	CAMBWKZ.RAW	Nonstd	1901	24 hours
	CAMBSAZ.RAW	Nonstd	1901	24 hours
	CAMBSUZ.RAW	Nonstd	1901	24 hours
			5703	
Output	USMBWKZ.RAW	Nonstd	57195	24 hours
	USMBSAZ.RAW	Nonstd	57195	24 hours
	USMBSUZ.RAW	Nonstd	57195	24 hours
			171585	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

MBMERGE.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! Command file is in USER62$DISK:[FUG.PREP.MOBILE]MBMERGE.LNK
$ ON WARNING THEN EXIT
$ ASSIGN/USER_MODE USER62$DISK:[FUG.PREP.MOBILE]MBMERGE.OBJ          A1
$ ASSIGN/USER_MODE MET24:[ROMOBJ.UTILIO]JFILEX.OBJ                    A2
$ ASSIGN/USER_MODE MET24:[ROMOBJ.UTILIO]IOCL.OBJ                      A3
$ ASSIGN/USER_MODE MET24:[ROMOBJ.UTILIO]ADATE.OBJ                     A4
$ ASSIGN/USER_MODE MET24:[ROMOBJ.UTILIO]JUNIT.OBJ                     A5
$ LINK -
  /EXECUTABLE = USER62$DISK:[FUG.PREP.MOBILE]MBMERGE.EXE -
  A1, -
  A2, -
  A3, -
  A4, -
  A5
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

MBMERGE.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! Command file is in USER2$DISK:[FUG.PREP.MOBILE]MBMERGE.COM
$ ASSIGN MLAB: SYS$PRINT
$ !assign USA area input files
$ ASSIGN WORK_SCRATCH:USMBSAZ.RAW        USMBSAT
$ ASSIGN WORK_SCRATCH:USMBSUZ.RAW        USMBSUN
$ ASSIGN WORK_SCRATCH:USMBWKZ.RAW        USMBWKD
$ !
$ !assign Canada area input files
$ ASSIGN MET52:[EMISSIONS.RNSUM05M]CAMBSAZ.RAW      CAMBSAT
$ ASSIGN MET52:[EMISSIONS.RNSUM05M]CAMBSUZ.RAW      CAMBSUN
$ ASSIGN MET52:[EMISSIONS.RNSUM05M]CAMBWKZ.RAW      CAMBWKD
$ !assign area output
$ !Canadian files are appended to US files and the files are
$ !called the same as the US files.
$ RUN USER2$DISK:[VMQ.PREP.ARMBSORT]MBMERGE.EXE
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

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

A. Main Program

MBMERGE

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.ROMNET]

JFILEX

B. Functions

None

IX. INCLUDE FILES

None

5.11 PREPROCESSOR PTEXTR (POINT-SOURCE EMISSIONS)

I. PREPROCESSOR FUNCTION

This preprocessor extracts specified seasonal point-source emissions data from the sorted annual point-source data file that is generated by preprocessor PTMERG. PTEXTR defines each point source that it extracts as either major or minor; you must declare the boundary between these types of point sources in the control cards. PTEXTR also separates the stack parameters (stack ID, stack height, plume height, etc.) from the emissions values and temporal factors. The temporal factors are hourly allocation factors over three day types: weekdays, Saturdays, and Sundays (see Section 5.2 for an example). The resultant output files are (1) a stack parameters file, with each point defined as a major or minor source, and (2) an annual emissions data file with allocation factors for a specific season, the three day types, and for every hour in a day. *Note that we believe that the ROM is only effective when simulating summer pollution episodes.*

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

POINTZ.DAT is a sequential data file that contains the annual point-source emissions data; it is generated by running the VAX SORT utility on the point-source data. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 5.21.

```
      READ(UNITFR, 1003, IOSTAT = IOSTR)
     &    SCC, STKID, ANOX, AVOC, ATHC, ACO, LAT, LON, COL, ROW,
     &    FLOWR, PLUME, DIAM, HGHT, TEMP, VELOC,
     &    (S_AFAC(ISET), D_SFAC(ISET),
     &    (H_DFAC(IHR, ISET), IHR = 1, 24), ISET = 1, 12),
     &    (FACTRS(ISP), ISP = 1, 13)

1003  FORMAT(I8, A12, 4F8.1, 7X, F8.5, F9.5, 2I4,
     &          2X, F7.0, F4.0, F4.0, 2F4.0, F9.2, 6X,
     &          12(F5.3, F6.4, 24F6.4),
     &          13E11.3)
```

(For an explanation of ISET, see Note for Table 5.21.)

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

b. SEANPT.CS0

SEANPT.CS0 (the seasonal annual point-source emissions file) contains the 15 chemical species that are shown in Table 5.22, and the seasonal, day-type, hourly allocation factors. The WRITE and FORMAT statements for this file are listed below.

```
      WRITE(UNITSA, 1005, IOSTAT = IOSTW)
     &    SSTKID,
     &    NSPECS2, (TCLASS(ISP), TSEAS_ANEM(ISP), ISP = 1, NSPECS2),
     &    MXDAY, (SSAFAC(IDAY), SDSFAC(IDAY),
     &          (SHDFAC(IHR, IDAY), IHR = 1, 24), IDAY = 1, MXDAY)
1005  FORMAT(A20, I3, <NSPECS2>(A5, E12.5), I2,
     &          <MXDAY>(F5.3, F6.4, 24F6.4))
```

TABLE 5.21. POINTZ.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	SCC_p		Integer*4	Source classification code
2	$STKID_p$		Char*12	AEROS state/county/plant/point ID
3	$ANOX_p$	tons·y^{-1}	Real*4	Annual NO_x emissions
4	$AVOC_p$	tons·y^{-1}	Real*4	Annual VOC emissions
5	$ATHC_p$	tons·y^{-1}	Real*4	Annual THC emissions
6	ACO_p	tons·y^{-1}	Real*4	Annual CO emissions
7	LAT_p	° N	Real*4	Latitude (fractional)
8	LON_p	° W	Real*4	Longitude (fractional)
9	COL_p		Integer*4	Column number
10	ROW_p		Integer*4	Row number
11	$FLOWR_p$	ft^3·s^{-1}	Real*4	Exhaust flow rate
12	$PLUME_p$	ft	Real*4	Plume height
13	$DIAM_p$	ft	Real*4	Stack diameter
14	$HGHT_p$	ft	Real*4	Stack height
15	$TEMP_p$	°F	Real*4	Exhaust gas temperature
16	$VELOC_p$	ft·s^{-1}	Real*4	Gas exit velocity
17	ISET		Integer*4	No. of day types in season
18	S_AFAC_p		Real*4	Seasonal allocation factor
19	D_SFAC_p		Real*4	Daily allocation factor
20	H_DFAC_p		Real*4	Hourly allocation factor
21	ISP		Integer*4	No. of speciation factors
22				Speciation factor for:
	$FACTRS_s$	mol·kg^{-1}	Real*4	olefins
	$FACTRS_s$	mol·kg^{-1}	Real*4	paraffins
	$FACTRS_s$	mol·kg^{-1}	Real*4	toluene
	$FACTRS_s$	mol·kg^{-1}	Real*4	xylene
	$FACTRS_s$	mol·kg^{-1}	Real*4	formaldehyde
	$FACTRS_s$	mol·kg^{-1}	Real*4	higher aldehydes
	$FACTRS_s$	mol·kg^{-1}	Real*4	ethene
	$FACTRS_s$	mol·kg^{-1}	Real*4	isoprenes
	$FACTRS_s$	mol·kg^{-1}	Real*4	nonreactive hydrocarbons
	$FACTRS_s$	mol·kg^{-1}	Real*4	methane
	$FACTRS_s$	%	Real*4	NO
	$FACTRS_s$	%	Real*4	NO_2
	$FACTRS_s$	mol·kg^{-1}	Real*4	methanol

Note: p = point source; s = species; units of tons are English short tons. NO reported as weight of NO_2.
Allocation factors for each point are repeated as (season/day type/hour), specifically:

winter/weekday/(1 - 24)	ISET = 1
winter/Saturday/(1 - 24)	ISET = 2
winter/Sunday/(1 - 24)	ISET = 3
spring/weekday/(1 - 24)	ISET = 4
spring/Saturday/(1 - 24)	ISET = 5
spring/Sunday/(1 - 24)	ISET = 6
summer/weekday/(1 - 24)	ISET = 7
summer/Saturday/(1 - 24)	ISET = 8
summer/Sunday/(1 - 24)	ISET = 9
fall/weekday/(1 - 24)	ISET = 10
fall/Saturday/(1 - 24)	ISET = 11
fall/Sunday/(1 - 24)	ISET = 12

TABLE 5.22. SEANPT.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	SSTKID		Char*20	Stack ID
2	NSPECS2		Integer*4	Number of species
3	TCLASS(1)		Char*5	AEROS code for VOC
4	TSEAS_ANEM(1)$_p$	tons·y^{-1}	Real*4	Annual emission rate, VOC
5	TCLASS(2)		Char*5	AEROS code for CO
6	TSEAS_ANEM(2)$_p$	tons·y^{-1}	Real*4	Annual emission rate, CO
7	TCLASS(3)		Char*5	AEROS code for olefins
8	TSEAS_ANEM(3)$_p$	tons·y^{-1}	Real*4	Annual emission rate, olefins
9	TCLASS(4)		Char*5	AEROS code for paraffins
10	TSEAS_ANEM(4)$_p$	tons·y^{-1}	Real*4	Annual emission rate, paraffins
11	TCLASS(5)		Char*5	AEROS code for toluene
12	TSEAS_ANEM(5)$_p$	tons·y^{-1}	Real*4	Annual emission rate, toluene
13	TCLASS(6)		Char*5	AEROS code for xylene
14	TSEAS_ANEM(6)$_p$	tons·y^{-1}	Real*4	Annual emission rate, xylene
15	TCLASS(7)		Char*5	AEROS code for formaldehyde
16	TSEAS_ANEM(7)$_p$	tons·y^{-1}	Real*4	Annual emission rate, formald.
17	TCLASS(8)		Char*5	AEROS code for aldehydes
18	TSEAS_ANEM(8)$_p$	tons·y^{-1}	Real*4	Annual emission rate, aldehydes
19	TCLASS(9)		Char*5	AEROS code for ethene
20	TSEAS_ANEM(9)$_p$	tons·y^{-1}	Real*4	Annual emission rate, ethene
21	TCLASS(10)		Char*5	AEROS code for nonr.
22	TSEAS_ANEM(10)$_p$	tons·y^{-1}	Real*4	Annual emission rate, nonr.
23	TCLASS(11)		Char*5	AEROS code for isoprenes
24	TSEAS_ANEM(11)$_p$	tons·y^{-1}	Real*4	Annual emission rate, isoprenes
25	TCLASS(12)		Char*5	AEROS code for methane
26	TSEAS_ANEM(12)$_p$	tons·y^{-1}	Real*4	Annual emission rate, methane
27	TCLASS(13)		Char*5	AEROS code for NO
28	TSEAS_ANEM(13)$_p$	tons·y^{-1}	Real*4	Annual emission rate, NO
29	TCLASS(14)		Char*5	AEROS code for NO_2
30	TSEAS_ANEM(14)$_p$	tons·y^{-1}	Real*4	Annual emission rate, NO_2
31	TCLASS(15)		Char*5	AEROS code for methanol
32	TSEAS_ANEM(15)$_p$	tons·y^{-1}	Real*4	Annual emission rate, methanol
33	MXDAY		Integer*4	No. of day types per season
34	SSAFAC$_p$		Real*4	Seasonal allocation factor
35	SDSFAC$_p$		Real*4	Weekday allocation factor
36	SHDFAC(1)$_p$		Real*4	Hour 1 allocation factor
⋮	⋮		⋮	⋮
60	SHDFAC(24)$_p$		Real*4	Hour 24 allocation factor
61	SSAFAC$_p$		Real*4	Seasonal allocation factor
62	SDSFAC$_p$		Real*4	Saturday allocation factor
63	SHDFAC(1)$_p$		Real*4	Hour 1 allocation factor
⋮	⋮		⋮	⋮
87	SHDFAC(24)$_p$		Real*4	Hour 24 allocation factor
88	SSAFAC$_p$		Real*4	Seasonal allocation factor
89	SDSFAC$_p$		Real*4	Sunday allocation factor
90	SHDFAC(1)$_p$		Real*4	Hour 1 allocation factor
⋮	⋮		⋮	⋮
114	SHDFAC(24)$_p$		Real*4	Hour 24 allocation factor

p = point source; nonr. = nonreactive hydrocarbons; formald. = formaldehyde; units are English tons. NO reported as weight of NO_2. For information on AEROS, see *AEROS Manual of Codes*, EPA-450/2-76-005a.

b. STACKS.CS0

STACKS.CS0 contains the point-source stack and locational parameters. Note that PTEXTR converts units from English to metric. The WRITE and FORMAT statements for this file are listed below, and its parameters are shown in Table 5.23.

```
      WRITE(UNITST, 1007, IOSTAT = IOSTW) STKFLG, LSTKID, LLON,
     &                               LLAT, LFLOW, LPLUME, LDIAM,
     &                               LTEMP, LVELOC, LHGHT
1007  FORMAT(I1, T4, A12, T18, 2F10.5, 6F10.4)
```

TABLE 5.23. STACKS.CS0 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$STKFLG_p$		Integer*4	Major/minor point source flag: 0 = major; 1 = minor
2	$LSTKID_p$		Char*12	AEROS state/county/plant/point ID
3	$LLON_p$	° W	Real*4	Longitude (fractional)
4	$LLAT_p$	° N	Real*4	Latitude (fractional)
5	$LFLOW_p$	$m^3 \cdot s^{-1}$	Real*4	Exhaust flow rate
6	$LPLUME_p$	m	Real*4	Plume height
7	$LDIAM_p$	m	Real*4	Stack diameter
8	$LTEMP_p$	°C	Real*4	Exhaust gas temperature
9	$LVELOC_p$	$m \cdot s^{-1}$	Real*4	Exhaust gas exit velocity
10	$LHGHT_p$	m	Real*4	Stack height

Note: p = point source. For information on AEROS, see *AEROS Manual of Codes*, EPA-450/2-76-005a.

III. CONTROL CARDS

Three control cards are used to input control data in the format shown below. Table 5.24 defines the control card variables.

YEAR
MXEMIS
SELSEA

TABLE 5.24. CONTROL CARD VARIABLES

Variable Name	Unit	Description
YEAR		Emissions base year as *YYYY* (1980 or 1985)
MXEMIS	$tons \cdot y^{-1}$	Major or minor point cut-off (for THC and NO_x)
SELSEA		Season for extraction

Note: Units are English short tons.

Example:

'1985'
500
'SUMMER'

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	4	files	75	blocks
FORTRAN INCLUDE files:	4	files	7	blocks
Object files:	4	files	42	blocks
Executable file:	1	file	48	blocks
	13	files	172	blocks

B. Execution Time Requirements (Representative Values for One Season's Extractions):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:41:51
Buffered I/O count:	614
Direct I/O count:	9052
Peak working-set size:	6950
Peak virtual size:	8931

C. Space Requirements: Log and Print Files

PTEXTR.LOG:	12 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.25 shows the input file and output file space requirements.

TABLE 5.25. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	POINTZ.DAT	Nonstd	94065	annual
Output	SEANPT.CS0	Nonstd	26204	seasonal
	STACKS.CS0	Nonstd	3021	
			29225	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

PTEXTR.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[WAY.PTEXTR]PTEXTR.LNK
$ ON WARNING THEN EXIT
$ ASSIGN/USER_MODE USER2$DISK:[WAY.PTEXTR]PTEXTR.OBJ              A1
$ ASSIGN/USER_MODE USER2$DISK:[WAY.PTEXTR]RDFRPT.OBJ              A2
$ ASSIGN/USER_MODE USER2$DISK:[WAY.PTEXTR]RD85PT.OBJ              A3
$ ASSIGN/USER_MODE USER2$DISK:[WAY.PTEXTR]SRTREC.OBJ              A4
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]ADATE.OBJ            A5
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]GETMSG.OBJ           A6
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]IOCL.OBJ             A7
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]INDEX1.OBJ           A8
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JFILE2.OBJ           A9
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JFILEX.OBJ           A10
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JUNIT.OBJ            A11
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]PROGID.OBJ           A12
$ LINK -
  /EXECUTABLE = USER2$DISK:[WAY.PTEXTR]PTEXTR.EXE -
  A1, -
  A2, -
  A3, -
   .
   .
   .
  A10, -
  A11, -
  A12
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

PTEXTR.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ !
$ ! Run stream is in [WAY.PTEXTR]PTEXTR.COM
$ !
$ ASSIGN MET20:[SCRATCH] SORTWORK0
$ ASSIGN MET27:[SCRATCH] SORTWORK1
$ !
$ ! Input file:
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]POINTZ.DAT        FREDSPT
$ !
$ ! Output files:
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]SEANPT.CSO        SEANPT
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]STACKS.CSO        STACK
$ !
$ RUN USER2$DISK:[WAY.PTEXTR]PTEXTR.EXE
'1985'
500
'SUMMER'
$ !
$ DIRECTORY/FULL FREDSPT
$ DIRECTORY/FULL SEANPT
$ DIRECTORY/FULL STACK
```

(end of run stream)

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

A. Main Program

PTEXTR

B. Subroutines

RD85PT
RDFRPT
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
JFILEX
PROGID

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETMSG
INDEX1
IOCL
JFILE2
JUNIT

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.PROCES]

PARAMS.EXT
SEANFL.EXT
SPCLIS.EXT
SRFILE.EXT

5.12 PREPROCESSOR PTMERG (POINT-SOURCE EMISSIONS)

I. PREPROCESSOR FUNCTION

This preprocessor merges the two sequential data files (produced by WINDOWPT) that contain the U.S. and Canadian point-source emissions. The U.S. data are in numerical order by the point-source AEROS State/County/plant/point codes; the Canadian data are in numerical order by the point-source AEROS Province/County/plant/point codes, and are appended to the U.S. data. The merged data are then sorted with the VAX SORT routine.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

Two nonstandard input files, USPOINTZ.RAW and CAPOINTZ.RAW, are read by PTMERG using the following common READ and FORMAT statement:

```
    READ (UNIT1,200,IOSTAT=IOST) STKSCC, BUFFER
200 FORMAT (A22,A2126)
```

Table 5.26 shows the files' parameters.

TABLE 5.26. USPOINTZ.RAW AND CAPOINTZ.RAW PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description [a]
1	$STKSCC_p$		Char*22	AEROS State (Province) /County/plant/ point identifier combined with the Source Classification Code.
2	$BUFFER_p$		Char*2126	Point-source emissions data, and associated temporal and speciation factors.

a. 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.
The NEDS (National Emissions Data System) Source Classification Codes pertain to the type of emitter for the point source (e.g., an oil-fired boiler). The thousands of NEDS codes can be located in *Criteria Pollutant Emission Factors for the 1985 NAPAP Emission Inventory*, EPA-600/7-87/015.

Note: p = point source.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

PTMERG generates the sequential file POINTZ.RAW, which is produced by the WRITE and FORMAT statement below; Table 5.27 shows the file's parameters.

```
    WRITE (UNIT3,290) NSTKSC, BUFFER
290 FORMAT (A20,A2126)
```

TABLE 5.27. POINTZ.RAW PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description [a]
1	$NSTKSC_p$		Char*20	AEROS State/County/plant/point identifier (embedded blank spaces removed) combined with the Source Classification Code.
2	$BUFFER_p$		Char*2126	Point-source emissions data, and associated temporal and speciation factors.

a. 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.
The NEDS (National Emissions Data System) Source Classification Codes pertain to the type of emitter for the point source (e.g., an oil-fired boiler). The thousands of NEDS codes can be located in *Criteria Pollutant Emission Factors for the 1985 NAPAP Emission Inventory*, EPA-600/7-87/015.

Note: p = point source.

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	9	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	6	blocks
Executable file:	1	file	12	blocks
	3	files	27	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:02:05
Buffered I/O count:	1404
Direct I/O count:	22672
Peak working-set size:	13245
Peak virtual size:	15076

C. Space Requirements: Log and Print Files

PTMERG.LOG:	6 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.28 shows the input file and output file space requirements.

TABLE 5.28. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	USPOINTZ.RAW	Nonstd	91095	annual
	CAPOINTZ.RAW	Nonstd	3060	annual
			94155	
Output	POINTZ.RAW	Nonstd	94065	annual

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

PTMERG.LNK follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ LINK PTMERG, -
  USER2$DISK:[ROMNET.UTILIO]JUNIT.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

PTMERG.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ !
$ ! RUN STREAM IS IN MET53:[EMISSIONS.SOURCE]PTMERG.COM
$ !
$ ! ************ FOR ROMNET REGION *****************
$ !
$ ASSIGN MLAB: SYS$PRINT
$ ! Assign sorting workspace:
$ ASSIGN MET20: SORTWORK0
$ ASSIGN MET35: SORTWORK1
$ ! Assign U.S. and Canada input files:
$ ASSIGN MET52:[EMISSIONS.RNSUM05P]USPOINTZ.RAW STATES
$ ASSIGN MET52:[EMISSIONS.RNSUM05P]CAPOINTZ.RAW CANADA
$ ! Assign output file:
$ ASSIGN WORK_SCRATCH:POINTZ.RAW FILEOT
$ ON ERROR THEN EXIT
$ !
$ RUN MET53:[EMISSIONS.SOURCE]PTMERG.EXE
$ !
$ SET VERIFY
$ ASSIGN WORK_SCRATCH:POINTZ.RAW INFILE
$ ASSIGN MET53:[EMISSIONS.SCRATCH]POINTZ.DAT OUTFILE
$ SORT/KEY=(POSITION=9,SIZE=12) -
  INFILE -
  OUTFILE
$ DIR/FULL OUTFILE
```

(end of run stream)

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

A. Main Program

PTMERG

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

DISK$VAXSET:[ROMLIB.ROMNET]

JUNIT

IX. INCLUDE FILES

None

5.13 PREPROCESSOR QAOUTP (POINT-SOURCE EMISSIONS QUALITY ASSURANCE)

I. PREPROCESSOR FUNCTION

This preprocessor extracts the four major emissions categories (NO_x, CO, THC, and VOC) from the sorted annual point-source data file that is generated by preprocessor PTMERG. The output file from QAOUTP is used by other preprocessors for quality control purposes only.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

POINTZ.DAT is a sequential data file that contains the annual point-source emissions data; it is generated by running the VAX SORT utility on the point-source data. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 5.29.

```
      READ(UNIT2,100,IOSTAT=IOST1)SCC,STKID,(EMIS(I),I=1,4),
     &    LAT,LON,COL,ROW,EXFLOW,PLUME,STKDIA,STKHT,STKTMP,STKVEL,
     &    (SEATMP(I),DAYTMP(I),(TMPHR(J,I),J=0,23),I=1,12)
100   FORMAT(I8, A12, 4F8.1, 7X, F8.5, F9.5, 2I4, 2X,
     &       I7, I4, F4.1, 2I4, F9.2, 6X,
     &       12(F5.3, F6.4, 24F6.4))
```

TABLE 5.29. POINTZ.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	SCC_p		Integer*4	Source classification code
2	$STKID_p$		Char*12	AEROS state/county/plant/point ID
3	$EMIS(1)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual NO_x emissions
4	$EMIS(2)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual VOC emissions
5	$EMIS(3)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual THC emissions
6	$EMIS(4)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual CO emissions
7	LAT_p	° N	Real*4	Latitude (fractional)
8	LON_p	° W	Real*4	Longitude (fractional)
9	COL_p		Integer*4	Column number
10	ROW_p		Integer*4	Row number
11	$EXFLOW_p$	$ft^3{\cdot}s^{-1}$	Real*4	Exhaust flow rate
12	$PLUME_p$	ft	Real*4	Plume height
13	$STKDIA_p$	ft	Real*4	Stack diameter
14	$STKHT_p$	ft	Real*4	Stack height
15	$STKTMP_p$	°F	Real*4	Exhaust gas temperature
16	$STKVEL_p$	$ft{\cdot}s^{-1}$	Real*4	Gas exit velocity
18	$SEATMP_p$		Real*4	Seasonal allocation factor
19	$DAYTMP_p$		Real*4	Daily allocation factor
20	$TMPHR_p$		Real*4	Hourly allocation factor

p = point source; units (tons) are English tons. For information on AEROS, see *AEROS Manual of Codes*; EPA-450/2-76-005a.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

QAOUTP generates QAPOINT.DAT, a binary sequential file used by preprocessor TOPEMI. The WRITE statement for this file appears below, and its parameters are shown in Table 5.30.

```
WRITE(IUNIT)STKSV, COLSV, ROWSV, POLLSV
```

TABLE 5.30. QAPOINT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	STKSV		Char*12	AEROS State/County/plant/point code
2	COLSV		Integer*4	Modeling region column
3	ROWSV		Integer*4	Modeling region row
4	$POLLSV(1)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, NO_x
5	$POLLSV(2)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, VOC
6	$POLLSV(3)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, THC
7	$POLLSV(4)_p$	$tons \cdot y^{-1}$	Real*4	Annual emission rate, CO

p = point source; units are English tons. For information on AEROS, see *AEROS Manual of Codes*; EPA-450/2-76-005a.

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 5.31 defines the control card variables.

ACOL, AROW, DAYID

TABLE 5.31. CONTROL CARD VARIABLES

Variable Name	Unit	Description
		Translation from NAPAP grid to ROMNET grid origin (SW corner):
ACOL		Column translation
AROW		Row translation
DAYID		Number of day types per season to skip (see note)

Note: Day types 1 - 3 = winter; 4 - 6 = spring; 7-9 = summer; 10 - 12 = fall.

Example:

161, 69, 7

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	18	blocks
FORTRAN INCLUDE files:	1	files	2	blocks
Object files:	2	files	9	blocks
Executable file:	1	file	21	blocks
	6	files	50	blocks

B. Execution Time Requirements (Representative Values for One Season's Extractions):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:27:21
Buffered I/O count:	214
Direct I/O count:	6306
Peak working-set size:	506
Peak virtual size:	2514

C. Space Requirements: Log and Print Files

QAOUTP.LOG:	6 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.32 shows the input file and output file space requirements.

TABLE 5.32. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	POINTZ.DAT	Nonstd	94065	annual
Output	QAPOINT.DAT	Nonstd	2676	annual

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

QAOUTP.LNK follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ LINK/EXE=QAOUTP -
  QAOUTP,QAINPT, -
  USER2$DISK:[ROMNET.UTILIO]JFILE2, -
  USER2$DISK:[ROMNET.UTILIO]JFILE4, -
  USER2$DISK:[ROMNET.UTILIO]JUNIT, -
  USER2$DISK:[ROMNET.UTILIO]IOCL, -
  USER2$DISK:[ROMNET.UTILIO]PROGID, -
  USER2$DISK:[ROMNET.UTILIO]ADATE, -
  USER2$DISK:[ROMNET.UTILIO]INDEX1
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

QAOUTP.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ !
$ !  RUN STREAM IS IN [WAY.ROM21.EMISQA.1985]QAOUTP.COM
$ !
$ !  NAPAP 5.3 DATA
$ !------------------------------------------------------------
$ !  User related input (control cards)
$ !     ACOL  = NUMBER OF COLUMNS TO SHIFT TO SET LOWER LEFT
$ !             GRID CELL TO  (1,1)
$ !     AROW  = SAME AS ACOL BUT FOR ROWS
$ !     DAYID = NUMBER OF CHARACTER "SETS" TO PASS IN POINTZ.DAT
$ !             FILE TO ARRIVE AT THE CORRECT SEASON
$ !             (1=WINTER, 4=SPRING, 7=SUMMER, 10=FALL)
$ !    FORMAT:  ACOL,AROW,DAYID
$ !____________________________________________________________
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]POINTZ.DAT       ANNUAL_FILE
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]QAPOINT.DAT      OUTFILE
$ RUN USER2$DISK:[WAY.ROM21.EMISQA.1985]QAOUTP.EXE
161,69,7
```

(end of run stream)

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

A. Main Program

QAOUTP

B. Subroutines

QAINPT.FOR

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
JFILE4
PROGID

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

INDEX1
IOCL
JFILE2
JUNIT

IX. INCLUDE FILES

QAINFL.EXT

5.14 PREPROCESSOR QASRCE (POINT-SOURCE EMISSIONS QUALITY ASSURANCE)

I. PREPROCESSOR FUNCTION

This preprocessor reduces the annual point-source emissions data file POINTZ.DAT into the four major emissions categories (NO_x, VOC, THC, CO) for quality control purposes only. (The annual point source file contains these four major emissions species as well as thirteen subspecies [11 hydrocarbon and 2 NO_x].)

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

POINTZ.DAT is a sequential data file that contains the annual point-source emissions data; it is generated by the VAX SORT utility. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 5.33.

```
      READ(UNIT2,100,IOSTAT=IOST1)SCC,STKID,(EMIS(I),I=1,4),
     &     LAT,LON,COL,ROW,EXFLOW,PLUME,STKDIA,STKHT,STKTMP,STKVEL,
     &     (SEATMP(I),DAYTMP(I),(TMPHR(J,I),J=0,23),I=1,12)
100   FORMAT(I8, A12, 4F8.1, 7X, F8.5, F9.5, 2I4, 2X,
     &        I7, I4, F4.1, 2I4, F9.2, 6X,
     &        12(F5.3, F6.4, 24F6.4))
```

TABLE 5.33. POINTZ.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	SCC_p		Integer*4	Source classification code
2	$STKID_p$		Char*12	AEROS state/county/plant/point ID
3	$EMIS(1)_p$	$tons \cdot y^{-1}$	Real*4	Annual NOx emissions
4	$EMIS(2)_p$	$tons \cdot y^{-1}$	Real*4	Annual VOC emissions
5	$EMIS(3)_p$	$tons \cdot y^{-1}$	Real*4	Annual THC emissions
6	$EMIS(4)_p$	$tons \cdot y^{-1}$	Real*4	Annual CO emissions
7	LAT_p	° N	Real*4	Latitude (fractional)
8	LON_p	° W	Real*4	Longitude (fractional)
9	COL_p		Integer*4	Column number
10	ROW_p		Integer*4	Row number
11	$EXFLOW_p$	$ft^3 \cdot s^{-1}$	Real*4	Exhaust flow rate
12	$PLUME_p$	ft	Real*4	Plume height
13	$STKDIA_p$	ft	Real*4	Stack diameter
14	$STKHT_p$	ft	Real*4	Stack height
15	$STKTMP_p$	°F	Real*4	Exhaust gas temperature
16	$STKVEL_p$	$ft \cdot s^{-1}$	Real*4	Gas exit velocity
18	$SEATMP_p$		Real*4	Seasonal allocation factor
19	$DAYTMP_p$		Real*4	Daily allocation factor
20	$TMPHR_p$		Real*4	Hourly allocation factor

p = point source; units (tons) are English tons. For information on AEROS, see *AEROS Manual of Codes*; EPA-450/2-76-005a.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

QASRCE generates QASOURCE.DAT, a binary sequential file. The WRITE statement for this file is listed below, and its parameters are shown in Table 5.34.

```
WRITE(IUNIT)SCC, STKID, LAT, LON, COL, ROW, POLLSV
```

TABLE 5.34. QASOURCE.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	SCC_p		Integer*4	Source classification code
2	$STKID_p$		Char*12	AEROS state/county/plant/point code
3	LAT_p	° N	Real*4	Stack latitude
4	LON_p	° W	Real*4	Stack longitude
5	COL_p		Integer*4	Modeling region column
6	ROW_p		Integer*4	Modeling region row
7	$POLLSV(1)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual emission rate, NO_x
8	$POLLSV(2)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual emission rate, VOC
9	$POLLSV(3)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual emission rate, THC
10	$POLLSV(4)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual emission rate, CO

p = point source; units (tons) are English tons. For information on AEROS, see *AEROS Manual of Codes*; EPA-450/2-76-005a.

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 5.35 defines the control card variables.

ACOL, AROW, DAYID

TABLE 5.35. CONTROL CARD VARIABLES

Variable Name	Unit	Description
		Translation from NAPAP grid to ROMNET grid origin (SW corner):
ACOL		Column translation
AROW		Row translation
DAYID		Number of day types per season to skip (see note)

Note: Day types 1 - 3 = winter; 4 - 6 = spring; 7-9 = summer; 10 - 12 = fall.

Example:

161, 69, 7

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	18	blocks
FORTRAN INCLUDE files:	1	files	2	blocks
Object files:	2	files	9	blocks
Executable file:	1	file	21	blocks
	6	files	50	blocks

B. Execution Time Requirements (Representative Values for One Season's Extractions):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:27:21
Buffered I/O count:	214
Direct I/O count:	6306
Peak working-set size:	506
Peak virtual size:	2514

C. Space Requirements: Log and Print Files

QASRCE.LOG:	6 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.36 shows the input file and output file space requirements.

TABLE 5.36. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	POINTZ.DAT	Nonstd	94065	annual
Output	QASOURCE.DAT	Nonstd	3176	annual

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

QASRCE.LNK follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ LINK/EXE=QASRCE -
  QASRCE,QAINPT, -
  USER2$DISK:[ROMNET.UTILIO]JFILE2, -
  USER2$DISK:[ROMNET.UTILIO]JFILE4, -
  USER2$DISK:[ROMNET.UTILIO]JUNIT, -
  USER2$DISK:[ROMNET.UTILIO]IOCL, -
  USER2$DISK:[ROMNET.UTILIO]PROGID, -
  USER2$DISK:[ROMNET.UTILIO]ADATE, -
  USER2$DISK:[ROMNET.UTILIO]INDEX1
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

QASRCE.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ !
$ !  RUN STREAM IS IN [WAY.ROM21.EMISQA.1985]QASRCE.COM
$ !
$ !  NAPAP 5.3 DATA
$ !---------------------------------------------------------------
$ !  User related input (control cards)
$ !     ACOL = NUMBER OF COLUMNS TO SHIFT TO SET LOWER LEFT
$ !            GRID CELL TO  (1,1)
$ !     AROW = SAME AS ACOL BUT FOR ROWS
$ !     DAYID = NUMBER OF CHARACTER "SETS" TO MOVE IN ON POINT.CSO
$ !             FILE TO GET CORRECT SEASON (1=WINTER, 4=SPRING,
$ !             7=SUMMER, 10=FALL)
$ !    FORMAT: ACOL,AROW,DAYID
$ !_______________________________________________________________
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]POINTZ.DAT   ANNUAL_FILE
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]QASOURCE.DAT OUTFILE
$ RUN USER2$DISK:[WAY.ROM21.EMISQA.1985]QASRCE.EXE
161,69,7
```

(end of run stream)

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

A. Main Program

QASRCE

B. Subroutines

QAINPT

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
JFILE4
PROGID

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

INDEX1
IOCL
JFILE2
JUNIT

IX. INCLUDE FILES

QASCFL.CBK

5.15 PREPROCESSOR QASTACK (POINT-SOURCE EMISSIONS QUALITY ASSURANCE)

I. PREPROCESSOR FUNCTION

This preprocessor reads the annual point-source emissions data file and identifies those point sources that have stack parameter information missing. This is performed as a quality assurance measure to make sure the necessary data are available for input to the ROM processor network. The output is printed to the log file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

POINTZ.DAT is a sequential data file that contains the annual point-source emissions data; it is generated by the VAX SORT utility. The READ and FORMAT statements for this file are listed below, and its parameters are shown in Table 5.37.

```
      READ(UNIT2,100,IOSTAT=IOST1)SCC,STKID,(EMIS(I),I=1,4),
     &     LAT,LON,COL,ROW,EXFLOW,PLUME,STKDIA,STKHT,STKTMP,STKVEL
100   FORMAT(I8, A12, 4F8.1, 7X, F8.5, F9.5, 2I4, 2X,
     &       I7, I4, F4.1, 2I4, F9.2)
```

TABLE 5.37. POINTZ.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	SCC_p		Integer*4	Source classification code
2	$STKID_p$		Char*12	AEROS state/county/plant/point ID
3	$EMIS(1)_p$	$tons \cdot y^{-1}$	Real*4	Annual NO_x emissions
4	$EMIS(2)_p$	$tons \cdot y^{-1}$	Real*4	Annual VOC emissions
5	$EMIS(3)_p$	$tons \cdot y^{-1}$	Real*4	Annual THC emissions
6	$EMIS(4)_p$	$tons \cdot y^{-1}$	Real*4	Annual CO emissions
7	LAT_p	° N	Real*4	Latitude (fractional)
8	LON_p	° W	Real*4	Longitude (fractional)
9	COL_p		Integer*4	Column number
10	ROW_p		Integer*4	Row number
11	$EXFLOW_p$	$ft^3 \cdot s^{-1}$	Real*4	Exhaust flow rate
12	$PLUME_p$	ft	Real*4	Plume height
13	$STKDIA_p$	ft	Real*4	Stack diameter
14	$STKHT_p$	ft	Real*4	Stack height
15	$STKTMP_p$	°F	Real*4	Exhaust gas temperature
16	$STKVEL_p$	$ft \cdot s^{-1}$	Real*4	Gas exit velocity

p = point source; units (tons) are English tons. For information on AEROS, see *AEROS Manual of Codes*; EPA-450/2-76-005a.

B. Output Files

1. Standard Output Files

None

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	36	blocks
FORTRAN INCLUDE files:	2	files	6	blocks
Object files:	3	files	24	blocks
Executable file:	1	file	27	blocks
	9	files	93	blocks

B. Execution Time Requirements (Representative Values for One Season's Extractions):

	VAX 8650
Charged CPU time (hh:mm:ss):	00:16:13
Buffered I/O count:	214
Direct I/O count:	6306
Peak working-set size:	506
Peak virtual size:	2514

C. Space Requirements: Log and Print Files

QASTACK.LOG:	45 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.38 shows the input file and output file space requirements.

TABLE 5.38. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	POINTZ.DAT	Nonstd	94065	annual
Output	None			

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

QASTACK.LNK follows:

```
LINK/EXE=QASTACK -
QASTACK,QASTRT,QASTPT, -
$2$DUA24:[ROMOBJ.UTILIO]JFILE2, -
$2$DUA24:[ROMOBJ.UTILIO]JFILE4, -
$2$DUA24:[ROMOBJ.UTILIO]JUNIT, -
$2$DUA24:[ROMOBJ.UTILIO]IOCL, -
$2$DUA24:[ROMOBJ.UTILIO]PROGID, -
$2$DUA24:[ROMOBJ.UTILIO]ADATE, -
$2$DUA24:[ROMOBJ.UTILIO]INDEX1, -
$2$DUA24:[ROMOBJ.UTILIO]GETMSG
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

QASTACK.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ !
$ !  RUN STREAM IS IN [WAY.ROM21.EMISQA.1985]QASTACK.COM
$ !
$ ASSIGN MET23: SORTWORKO
$ ASSIGN MET53:[EMISSIONS.RNSUM85P]POINTZ.DAT ANNUAL_FILE
$ RUN USER2$DISK:[WAY.ROM21.EMISQA.1985]QASTACK.EXE
```

(end of run stream)

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

A. Main Program

QASTACK

B. Subroutines

QASTRT
QASTPT

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
JFILE4
PROGID

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

INDEX1
IOCL
JFILE2
JUNIT

IX. INCLUDE FILES

QASTFL.CBK
QAOSTFL.CBK

5.16 PROGRAM RETFAC (REFORMATS THE TEMPERATURE ADJUSTMENT FILE)

RETFAC reformats the temperature adjustment factor file (ultimately used by processor P26G), and needs to be run only when you receive new data from FREDS. The factor file is received as a sequential ASCII file; the file number on the FREDS data tape will be identified on the accompanying documentation. RETFAC reads in the factor file, determines how the temperature factors are distributed for each year on the file, and creates header records that contain this information. RETFAC outputs a new factor file (RETFAC.CS*n* or RETFAC.C*nn*, where *n* and *nn* are the control strategy number) that is a direct access ASCII file. P26G can now read this file and directly access any record by using the information in the header. A sample header follows; please refer to processor P26G for further details concerning this file.

```
          REFORMATTED TEMPERATURE ADJUSTMENT FACTOR FILE
             CREATION DATE/TIME: 01-24-89 / 10:46:42
 1ST DATA AT RECORD:   44    INPUT RECORDS:    181   OUTPUT RECORDS:   223
    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
             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
             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
```

5.17 PREPROCESSOR TOPEMI (POINT-SOURCE EMISSIONS QUALITY ASSURANCE)

I. PREPROCESSOR FUNCTION

This preprocessor reads the abbreviated annual point-source emissions file from preprocessor QAOUTP, and sorts the point sources by your choice (specified in the control cards) of one of four major emissions species (NO_x, VOC, THC, CO). The listing is output to an ASCII print file.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

One nonstandard file is required, QAPOINT.DAT, a binary sequential file created by preprocessor QAOUTP. The READ statement for this file is listed below, and its parameters are shown in Table 5.39.

```
READ(IUNIT, IOSTAT=IOST1))STKID, COL, ROW, POLL
```

TABLE 5.39. QAPOINT.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	STKID		Char*12	AEROS state/county/plant/point code
2	COL		Integer*4	Modeling region column
3	ROW		Integer*4	Modeling region row
4	$POLL(1)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual emission rate, NO_x
5	$POLL(2)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual emission rate, VOC
6	$POLL(3)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual emission rate, THC
7	$POLL(4)_p$	$tons{\cdot}y^{-1}$	Real*4	Annual emission rate, CO

p = point source; units are English tons. For information on AEROS, see *AEROS Manual of Codes*; EPA-450/2-76-005a.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

None

III. CONTROL CARDS

Two control cards are used to input control data in the format shown below. Table 5.40 defines the control card variables.

PARM, NREC

TABLE 5.40. CONTROL CARD VARIABLES

Variable Name	Unit	Description
PARM		Name of the primary pollutant species to be sorted (NO_x, VOC, THC, CO)
NREC		Maximum number of point sources to be listed

Example:

NOX, 200

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:	4	files	12	blocks
Object files:	2	files	12	blocks
Executable file:	1	file	21	blocks
	9	files	60	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:12:10
Buffered I/O count:	214
Direct I/O count:	6306
Peak working-set size:	506
Peak virtual size:	2514

C. Space Requirements: Log and Print Files

TOPEMI.LOG:	12 blocks
Print Files:	36 blocks

D. Space Requirements: Input and Output Files

Table 5.41 shows the input file and output file space requirements.

TABLE 5.41. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	QAPOINT.DAT	Nonstd	2676	annual
Output	None			

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

TOPEMI.LNK follows:

```
$ LINK TOPEMI,TOPSRT, -
  $2$DUA24:[ROMOBJ.UTILIO]JFILE3, -
  $2$DUA24:[ROMOBJ.UTILIO]JFILE4, -
  $2$DUA24:[ROMOBJ.UTILIO]JUNIT, -
  $2$DUA24:[ROMOBJ.UTILIO]IOCL, -
  $2$DUA24:[ROMOBJ.UTILIO]PROGID, -
  $2$DUA24:[ROMOBJ.UTILIO]ADATE, -
  $2$DUA24:[ROMOBJ.UTILIO]INDEX1, -
  $2$DUA24:[ROMOBJ.UTILIO]GETMSG
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

TOPEMI.COM follows:

```
$ !ACCOUNTING CODE
$ SET VERIFY
$ ASSIGN MLAB: SYS$PRINT
$ !
$ !   RUN STREAM IS IN [WAY.ROM21.EMISQA.1985]TOPEMI.COM
$ !
$ ASSIGN USER1$DISK: SORTWORKO
$ ASSIGN MET52:[EMISSIONS.RNSUM85P]QAPOINT.DAT INFILE
$ ASSIGN MET1:[S01S]PTEMIS85A.DAT PRINTFILE
$ RUN USER2$DISK:[WAY.ROM21.EMISQA.1985]TOPEMI.EXE
'NOX',200
$ NETPRINT/NODE=MADMAD PRINTFILE
```

(end of run stream)

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

A. Main Program

TOPEMI

B. Subroutines

TOPSRT

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
JFILE4
PROGID

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETMSG
INDEX1
IOCL
JFILE2
JFILE3
JUNIT

IX. INCLUDE FILES

PRID.EXT
QATOPIN.CBK
QATOPOT.CBK
TOPLIS.EXT

5.18 PREPROCESSOR WINDOWAR (AREA-SOURCE EMISSIONS)

I. PREPROCESSOR FUNCTION

This preprocessor extracts (windows) the modeling domain's area-source emissions data from the VAX system tapes that contain the National Acid Precipitation Assessment Program (NAPAP) raw emissions data, and stores the windowed data on the VAX disk. Raw area-emissions are comprised of weekday, Saturday, and Sunday data for the U.S. and Canada, and are thus stored as six separate ASCII files (on two VAX system tapes). To create the corresponding disk files, you must execute WINDOWAR six times.

II. I/O COMPONENTS (using summer 1985 data as an example)

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

A VAX system tape, created by the COPYTAPE preprocessor, contains the U.S. area-source emissions data for weekdays, Saturday, and Sunday (referenced as Tape 100286 in the run stream below). File number 1 on this tape contains the header information, which WINDOWAR skips. Files 2, 3, and 4 contain the emissions data as listed above, and are read sequentially for each tape by running WINDOWAR three times successively. A second tape (referenced as Tape 101210 in the run stream below) similarly contains the Canadian area-source emissions data. Each data file contains 24 hours of data expressed in GMT. The LRECL for the tapes = 180 bytes, and the BLKSIZE = 3600 bytes (20 records per block). Each record is read from the tape using the following READ and FORMAT statement:

```
    READ (UNIT1,200,IOSTAT=IOST) BUFFER
200 FORMAT (A180)
```

Table 5.42 shows the tape parameter.

TABLE 5.42. TAPE PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	BUFFER		Char*180	One record: positions 1-4 are the column number of the grid cell; positions 5-8 are the row number of the grid cell; positions 9-180 contain the area-source emissions data.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

WINDOWAR generates six sequential disk files: USARWKZ.RAW, USARSAZ.RAW, and USARSUZ.RAW for the U.S.; CAARWKZ.RAW, CAARSAZ.RAW, and CAARSUZ.RAW for Canada. Each file contains 24 hours of data expressed in GMT, and each is produced by the following WRITE and FORMAT statements:

```
    WRITE (UNIT2,200) BUFFER
200 FORMAT (A180)
```

The characteristics and description of BUFFER are identical to those of the nonstandard input files.

III. CONTROL CARDS

One control card is used to input the region boundaries. The boundaries for your particular modeling domain (if it is not the ROMNET region) can be ascertained from Chapter 3 of Sellars *et al.* Table 5.43 defines the control card variables for the ROMNET region, and the free-format READ statement is listed below:

```
READ (5,*) WEST, EAST, SOUTH, NORTH
```

TABLE 5.43. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Data Type	Description
1	WEST		Integer*4	Western column boundary
2	EAST		Integer*4	Eastern column boundary
3	SOUTH		Integer*4	Southern row boundary
4	NORTH		Integer*4	Northern row boundary

Example (for ROMNET):

161,224,69,120

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	5	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	4	blocks
Executable file:	1	file	8	blocks
	3	files	17	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:03:06
Buffered I/O count:	1039
Direct I/O count:	13235
Peak working-set size:	530
Peak virtual size:	2684

C. Space Requirements: Log and Print Files

WINDOWAR.LOG: 21 blocks per tape read.
Print Files: None

D. Space Requirements: Input and Output Files

Table 5.44 shows the input file and output file space requirements.

TABLE 5.44. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	None		0	
Output	USARWKZ.RAW	Nonstd	19017	24 hours
	USARSAZ.RAW	Nonstd	19017	24 hours
	USARSUZ.RAW	Nonstd	19017	24 hours
	CAARWKZ.RAW	Nonstd	2261	24 hours
	CAARSAZ.RAW	Nonstd	2261	24 hours
	CAARSUZ.RAW	Nonstd	2261	24 hours
			63834	

E. Space Requirements: Tape Files

Table 5.45 shows the input tape file space requirements.

TABLE 5.45. INPUT TAPE FILE SPACE REQUIREMENTS

File Group	Tape No.	File No.	Record Length (in bytes)	Block Size (in bytes)
Input	100286	2, 3, 4	180	3600
	101210	2, 3, 4	180	3600

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

WINDOWAR.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! LINK STREAM IS IN USER0$DISK:[LBW.PREP.AREA]WINDOWAR.LNK
$ LINK USER0$DISK[LBW.PREP.AREA]WINDOWAR, -
  USER2$DISK:[IOLIB1.WTSIOLIB1]JUNIT
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

WINDOWAR.COM for windowing the U.S. data follows; WINDOWAR.COM for Canada is identical except for the tape numbers, file numbers, and the output file names.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ !
$ ! RUN STREAM IS IN USER0$DISK:[LBW.PREP.AREA]WINDOWAR.COM
$ !
$ !   **************** FOR ROMNET REGION **************
$ !
```

```
$ ASSIGN MLAB: SYS $PRINT
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARWKZ.RAW FILEOT
$ ON ERROR THEN EXIT
$ !
$ !ALLOCATE  $3 $MUA3 TAPEDRIVE
$ TAPE SELECT TAPEDRIVE 100286
$ TAPE LOAD TAPEDRIVE 100286
$ !
$ MOUNT/NOWRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=3600 -
  /RECORDSIZE=180 TAPEDRIVE 100286 TAPEIN
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ ! Skip over header if using a system tape
$ !
$ SET MAGTAPE/SKIP=FILES:1 TAPEIN
$ !
$ ! User input control cards to determine windowed region
$ ! western column boundary, eastern column boundary,
$ ! southern row boundary, northern row boundary
$ !
$ RUN USER0$DISK:[LBW.PREP.AREA]]WINDOWAR.EXE
161,224,69,120
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ DIR/FULL FILEOT
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARSAZ.RAW FILEOT
$ ! Skip over header if using a system tape
$ !
$ ! User input control cards to determine windowed region
$ ! western column boundary, eastern column boundary,
$ ! southern row boundary, northern row boundary
$ !
$ RUN USER0$DISK:[LBW.PREP.AREA]]WINDOWAR.EXE
161,224,69,120
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ DIR/FULL FILEOT
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USARSUZ.RAW FILEOT
$ ! Skip over header if using a system tape
$ !
$ ! User input control cards to determine windowed region
$ ! western column boundary, eastern column boundary,
$ ! southern row boundary, northern row boundary
$ !
$ RUN USER0$DISK:[LBW.PREP.AREA]]WINDOWAR.EXE
161,224,69,120
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ DIR/FULL FILEOT
$ @REWIND TAPEIN
$ DISMOUNT TAPEIN
$ DEALLOCATE TAPEDRIVE
```

(end of run stream)

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

A. Main Program

WINDOWAR

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

5.19 PREPROCESSOR WINDOWMB (MOBILE-SOURCE EMISSIONS)

I. PREPROCESSOR FUNCTION

This preprocessor extracts (windows) the modeling domain's mobile-source emissions data from the VAX system tapes that contain the National Acid Precipitation Assessment Program (NAPAP) raw emissions data, and stores the windowed data on the VAX disk. Raw mobile-emissions are comprised of weekday, Saturday, and Sunday data for the U.S. and Canada, and are thus stored as six separate ASCII files (on two VAX system tapes). To create the corresponding disk files, you must execute WINDOWMB six times.

II. I/O COMPONENTS (using summer 1985 data as an example)

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

A VAX system tape, created by the COPYTAPE preprocessor, contains the U.S. mobile-source emissions data for weekdays, Saturday, and Sunday (referenced as Tape 101076 in the run stream below). File number 1 on this tape contains the header information, which WINDOWMB skips. Files 2, 3, and 4 contain the emissions data as listed above, and are read sequentially for each tape by running WINDOWMB three times successively. A second tape similarly contains the Canadian mobile-source emissions data. Each data file contains 24 hours of data expressed in GMT. The LRECL for the tapes = 216 bytes, and the BLKSIZE = 4320 bytes (20 records per block). Each record is read from the tape using the following READ and FORMAT statement:

```
      READ (UNIT1,200,IOSTAT=IOST) BUFFER
200   FORMAT (A216)
```

Table 5.46 shows the tape parameter.

TABLE 5.46. TAPE PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	BUFFER		Char*216	One record: positions 1-4 are the column number of the grid cell; positions 5-8 are the row number of the grid cell; positions 9-216 contain the mobile-source emissions data.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

WINDOWMB generates six sequential disk files: USMBWKZ.RAW, USMBSAZ.RAW, and USMBSUZ.RAW for the U.S.; CAMBWKZ.RAW, CAMBSAZ.RAW, and CAMBSUZ.RAW for Canada. Each file contains 24 hours of data expressed in GMT, and each is produced by the following WRITE and FORMAT statements:

```
    WRITE (UNIT2,200) BUFFER
200 FORMAT (A216)
```

The characteristics and description of BUFFER are identical to those of the nonstandard input files.

III. CONTROL CARDS

One control card is used to input the region boundaries. The boundaries for your particular modeling domain (if it is not the ROMNET region) can be ascertained from Chapter 3 of Sellars *et al.* Table 5.47 defines the control card variables for the ROMNET region, and the free-format READ statement is listed below:

```
READ (5,*) WEST, EAST, SOUTH, NORTH
```

TABLE 5.47. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Data Type	Description
1	WEST		Integer*4	Western column boundary
2	EAST		Integer*4	Eastern column boundary
3	SOUTH		Integer*4	Southern row boundary
4	NORTH		Integer*4	Northern row boundary

Example (for ROMNET):

161,224,69,120

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	5	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	4	blocks
Executable file:	1	file	8	blocks
	3	files	17	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:07:22
Buffered I/O count:	2385
Direct I/O count:	34419
Peak working-set size:	533
Peak virtual size:	2684

C. Space Requirements: Log and Print Files

WINDOWMB.LOG:	21 blocks per tape
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.48 shows the input file and output file space requirements.

TABLE 5.48. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input			0	
Output	USMBWKZ.RAW	Nonstd	55294	24 hours
	USMBSAZ.RAW	Nonstd	55294	24 hours
	USMBSUZ.RAW	Nonstd	55294	24 hours
	CAMBWKZ.RAW	Nonstd	1901	24 hours
	CAMBSAZ.RAW	Nonstd	1901	24 hours
	CAMBSUZ.RAW	Nonstd	1901	24 hours
			171585	

E. Space Requirements: Tape Files

Table 5.49 shows the input tape file space requirements for Summer 1985. Note that any of the tape parameters are subject to change.

TABLE 5.49. INPUT TAPE FILE SPACE REQUIREMENTS

File Group	Tape No.	File No.	Record Length (in bytes)	Block Size (in bytes)
Input	101076	2, 3, 4	216	4320

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

WINDOWMB.LNK follows:

```
$ SET VERIFY
$ LINK WINDOWMB, -
  [ROMNET.UTILIO]JUNIT
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

WINDOWMB.COM for windowing the U.S. data follows; WINDOWMB.COM for Canada is identical except for the tape numbers, file numbers, and the output file names.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ !
$ ! RUN STREAM IS IN [FUG.PREP.MOBILE]WINDOWMB.COM
$ !
$ ASSIGN MLAB: SYS$PRINT
$ ASSIGN MET52:[EMISSIONS.RNSUM05M]USMBWKZ.RAW FILEOT
$ ON ERROR THEN EXIT
$ !
$ TAPE SELECT TAPEDRIVE 101076
$ TAPE LOAD TAPEDRIVE 101076
$ !
$ MOUNT/NOWRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=4320 -
  /RECORDSIZE=216 TAPEDRIVE 101076 TAPEIN
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ ! Skip over header if using a system tape
$ !
$ SET MAGTAPE/SKIP=FILES:1 TAPEIN
$ !
$ ! User input control cards to determine windowed region
$ ! western column boundary, eastern column boundary,
$ ! southern row boundary, northern row boundary
$ !
$ RUN USER2$DISK:[WAY.ROM21.EMISQA]WINDOWMB.EXE
161,224,69,120
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ DIR/FULL FILEOT
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM05M]USMBSAZ.RAW FILEOT
$ ! Skip over header if using a system tape
$ !
$ !SET MAGTAPE/SKIP=FILES:5 TAPEIN
$ !
$ ! User input control cards to determine windowed region
$ ! western column boundary, eastern column boundary,
$ ! southern row boundary, northern row boundary
$ !
$ RUN USER2$DISK:[WAY.ROM21.EMISQA]WINDOWMB.EXE
161,224,69,120
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ DIR/FULL FILEOT
$ !
$ ASSIGN MET52:[EMISSIONS.RNSUM05M]USMBSUZ.RAW FILEOT
$ ! Skip over header if using a system tape
$ !
$ !SET MAGTAPE/SKIP=FILES:1 TAPEIN
$ !
```

```
$ ! User input control cards to determine windowed region
$ ! western column boundary, eastern column boundary,
$ ! southern row boundary, northern row boundary
$ !
$ RUN USER2$DISK:[WAY.ROM21.EMISQA]WINDOWMB.EXE
161,224,69,120
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ DIR/FULL FILEOT
$ @REWIND TAPEIN
$ DISMOUNT TAPEIN
$ DEALLOCATE TAPEDRIVE
```

(end of run stream)

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

A. Main Program

WINDOWMB

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

DISK$VAXSET:[ROMLIB.ROMNET]

JUNIT

IX. INCLUDE FILES

None

5.20 PREPROCESSOR WINDOWPT (POINT-SOURCE EMISSIONS)

I. PREPROCESSOR FUNCTION

This preprocessor extracts (windows) the modeling domain's point-source emissions data from the VAX system tapes that contain the National Acid Precipitation Assessment Program (NAPAP) raw emissions data, and stores the windowed data on the VAX disk. Unlike the area- and mobile-source emissions data, the point-source data are an annual quantity, with associated seasonal, daily, and hourly temporal factors (see Section 5.2 for a full description of these factors). Point-source emissions for the U.S. and Canada are stored as two separate ASCII files (on two VAX system tapes). To create the corresponding disk files, you must execute WINDOWPT twice.

II. I/O COMPONENTS (using summer 1985 data as an example)

A. Input Files

1. Standard Input Files

   None

2. Nonstandard Input Files

   A VAX system tape, created by the COPYTAPE preprocessor, contains the U.S. point-source emissions data (referenced as Tape 100622 in the run stream below). A second tape (referenced as 101210) similarly contains the Canadian point-source emissions data. The LRECL for the tapes = 2148 bytes, and the BLKSIZE = 21480 bytes (10 records per block). Each record is read from the tape using the following READ and FORMAT statement:

```
    READ (UNIT1,200,IOSTAT=IOST) BUFFER
200 FORMAT (A2148)
```

   Table 5.50 shows the tape parameter.

TABLE 5.50. TAPE PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	BUFFER		Char*2148	One record: positions 1-8 contain the point-source classification code; positions 9-20 contain the AEROS State/County/plant/point code of each source;[a] positions 21-2148 contain the point-source emissions data, and associated temporal and speciation factors.

a. 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. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

WINDOWPT generates two sequential disk files in two independent runs: USPOINTZ.RAW for the U.S., and CAPOINTZ.RAW for Canada. Each file is produced by the following WRITE and FORMAT statements:

```
    WRITE (UNIT2,200) BUFFER
200 FORMAT (A2148)
```

The characteristics and description of BUFFER are identical to those of the nonstandard input files.

III. CONTROL CARDS

One control card is used to input the region boundaries. The boundaries for your particular modeling domain (if it is not the ROMNET region) can be ascertained from Chapter 3 of Sellars *et al.* Table 5.51 defines the control card variables for the ROMNET region, and the free-format READ statement is listed below:

```
READ (5,*) WEST, EAST, SOUTH, NORTH
```

TABLE 5.51. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Data Type	Description
1	WEST		Integer*4	Western column boundary
2	EAST		Integer*4	Eastern column boundary
3	SOUTH		Integer*4	Southern row boundary
4	NORTH		Integer*4	Northern row boundary

Example (for ROMNET):

161,224,69,120

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	6	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	6	blocks
Executable file:	1	file	12	blocks
	3	files	24	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:01:05
Buffered I/O count:	1405
Direct I/O count:	10222
Peak working-set size:	527
Peak virtual size:	2684

C. Space Requirements: Log and Print Files

WINDOWPT.LOG:	12 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 5.52 shows the input file and output file space requirements.

TABLE 5.52. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	None		0	
Output	USPOINTZ.RAW	Nonstd	91095	annual
	CAPOINTZ.RAW	Nonstd	3060	annual
			94155	

E. Space Requirements: Tape Files

Table 5.53 shows the input tape file space requirements.

TABLE 5.53. INPUT TAPE FILE SPACE REQUIREMENTS

File Group	Tape No.	File No.	Record Length (in bytes)	Block Size (in bytes)
Input	100622	1	2148	21480
	101210	1	2148	21480

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

WINDOWPT.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! LINK STREAM IS IN MET53:[EMISSIONS.SOURCE]WINDOWPT.LNK
$ LINK MET53:[EMISSIONS.SOURCE]WINDOWPT, -
  USER2$DISK:[ROMNET.UTILIO]JUNIT
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

WINDOWPT.COM for windowing the U.S. data follows; WINDOWPT.COM for Canada is identical except for the tape numbers, file numbers, and the output file names.

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ !
$ ! RUN STREAM IS IN MET53:[EMISSIONS.SOURCE]WINDOWPT.COM
$ !
$ !  **************** FOR ROMNET REGION **************
$ !
$ ASSIGN MLAB: SYS $PRINT
$ ASSIGN MET52:[EMISSIONS.RNSUM85A]USPOINTZ.RAW FILEOT
$ ON ERROR THEN EXIT
$ !
$ TAPE SELECT TAPEDRIVE BTAPE
$ TAPE LOAD TAPEDRIVE 100622
```

```
$ !
$ MOUNT/NOWRITE/FOREIGN/DENSITY=6250/BLOCKSIZE=21480 -
  /RECORDSIZE=2148 TAPEDRIVE 100622 TAPEIN
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ ! Skip over header if using a system tape
$ ! (plus any additional files on the tape)
$ !
$ SET MAGTAPE/SKIP=FILES:7 TAPEIN
$ !
$ ! User input control cards to determine windowed region
$ ! western column boundary, eastern column boundary,
$ ! southern row boundary, northern row boundary
$ !
$ RUN MET53:[EMISSIONS.SOURCE]WINDOWPT.EXE
161,224,69,120
$ !
$ SHOW DEV/FULL TAPEDRIVE
$ DIR/FULL FILEOT
$ @REWIND TAPEIN
$ DISMOUNT TAPEIN
$ DEALLOCATE TAPEDRIVE
```

(end of run stream)

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

A. Main Program

WINDOWPT

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

JUNIT

IX. INCLUDE FILES

None

This page is intentionally left blank.

SECTION 6

BIOGENIC EMISSIONS: SOURCES AND PREPROCESSING OF DATA

Hydrocarbon emissions influence the formation of ozone through the intricately-coupled atmospheric chemistry of nitrogen oxides and reactive organic hydrocarbons. Biogenic hydrocarbon emissions emanate from living surface vegetation--trees, shrubs, grasses, and agricultural crops--and from decaying leaf litter and vegetation in fresh and salt water. Hydrocarbon emissions from biogenic sources have been estimated to equal or exceed those from anthropogenic sources on a total-mass basis. Thus, biogenic hydrocarbon emission rates have become an important input requirement for regional oxidant models such as the ROM.

The calculation of biogenic hydrocarbon emission rates requires four basic components:

- estimates of the biomass density of each vegetative class in each grid cell;
- an adjustment of biomass density to account for season;
- emission factors for the vegetation classes in the modeling region; and
- empirical relationships that allow for adjusting the emission factors based on the values of specific environmental parameters, such as temperature, solar intensity, soil conditions, and elevation.

We show the procedure we use to calculate the hourly grid-specific biogenic emissions for the ROM in Figure 6.1, and describe it below. Our procedure provides the flexibility to update vegetation-specific emission factors, and allows us to evaluate the importance of an individual vegetative species in the modeling domain. We calculate the hourly emission rate for an individual grid cell and a specific hydrocarbon compound (or group of compounds) by adjusting the vegetation-specific emission factors for canopy (forest) and noncanopy (nonforest) areas to reflect variations in the meteorological episode being modeled, and then summing the canopy and noncanopy emissions.

6.1 BIOMASS DENSITY BY VEGETATION CLASS

Data from the Oak Ridge National Laboratory (ORNL) Geoecology Database (Olson, 1980) form the basis of the U. S. biogenic emissions inventory for the ROM. The database contains county-level land cover data for classes of natural vegetation, agricultural crops, urban areas, and water. Table 6.1 lists examples of vegetative species included in the biogenic emissions inventory system by vegetation class.

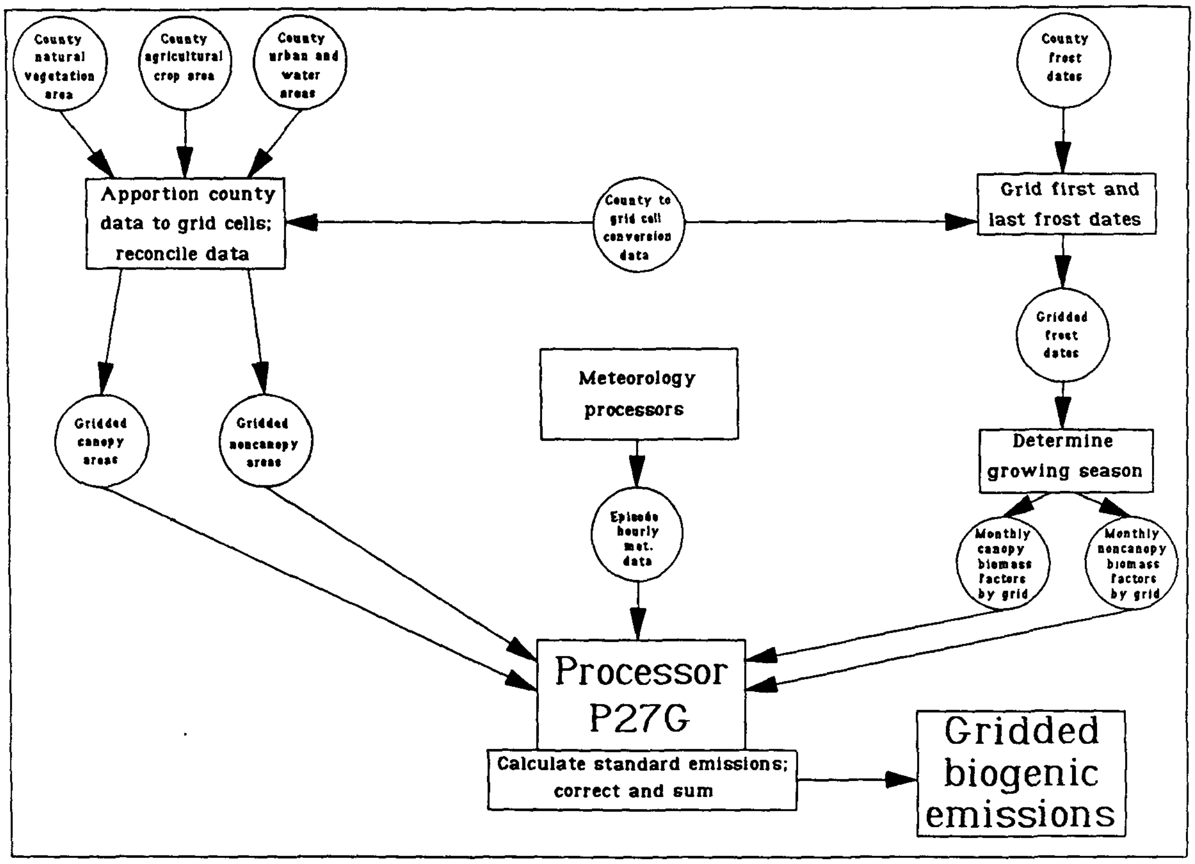

Figure 6.1. The biogenic emissions inventory system.

The Landsat data set (Page, 1980), which reports data in standard NAPAP grid cells, and vegetation data from Matthews (1984) form the basis of the Canadian biogenic emissions inventory. For the portion of Canada south of 55° N latitude, we used the Landsat data set to determine the types of vegetation present by land cover class (Page, 1980). However, the Landsat database contains no data for areas north of 55° N latitude; if you require data north of this latitude, you can use the vegetation, land use, and seasonal albedo data sets of Matthews (1984). Note that the Matthews data specify only one vegetation type for each 1° latitude by 1° longitude square.

TABLE 6.1. VEGETATION CLASSES IN THE BIOGENIC EMISSIONS INVENTORY SYSTEM

Vegetation class	Examples
	CANOPY (FOREST)
Natural vegetation:	
Oak	Oregon oakwoods, oak savanna, oak-hickory
Other deciduous	Elm-ash, northern hardwoods, beech-maple
Coniferous	Cypress savanna, Douglas fir, conifer bog
	NONCANOPY (NONFOREST)
Natural vegetation:	
Scrubland	Creosote bush, chaparral, coastal sagebrush
Grassland	Fescue-oatgrass, northern cordgrass, prairie
Agricultural crops:	Alfalfa, barley, corn, cotton, hay, oats, peanuts, potatoes, rice, rye, sorghum, soybeans, tobacco, wheat, miscellaneous crops
Urban area:	Urban grass, urban trees
Water (fresh and salt):	Inland lakes
Barren area:	Tundra, ice, alpine meadows, desert

6.1.1 Natural Vegetation Area

The ORNL Geoecology Potential and Adjusted Vegetation Data File uses Kuchler's vegetation codes (001-106) to identify natural vegetation. We categorized these into five natural vegetation classes: oak forests, other deciduous forests, coniferous forests, scrubland, and grassland.

For Canada, we assigned the vegetation types within the Matthews (1984) and the Landsat (Page, 1980) data sets to one of the above five natural vegetation classes. We calculated the area allocated to each class in each NAPAP grid cell. However, there was no direct correspondence to the oak class for either data sets. Therefore, we allocated a zero area to this class.[1]

1. A percentage of the deciduous areas could be allocated to the oak class if that percentage is known.

Oak, other deciduous, and coniferous forests are categorized as canopy (forest) vegetation classes. Canopy emissions are determined by biomass density, a measure of the dry leaf biomass per unit area (kg/ha). Table 6.2 presents the biomass density for the three canopy vegetation classes.

TABLE 6.2. FOREST BIOMASS DENSITY ESTIMATES

	Forest biomass density (kg/ha) by canopy vegetation class		
Forest biomass class	Oak	Other deciduous	Coniferous
Deciduous high isoprene	1,850	600	390
Deciduous low isoprene	600	1,850	260
Deciduous nonisoprene	600	900	260
Coniferous nonisoprene	700	1,350	5,590

Source: Lamb *et al.*, 1987.

For a single canopy class, we use four forest biomass classes to describe the mix of forest vegetation, including undergrowth, within that class. All oaks (and some other deciduous tree species) that emit $>10\ \mu g_{isoprene}/(g_{biomass} \cdot h)$ at temperatures near 30°C are grouped together as high isoprene emitters. All deciduous tree species with an emission rate $<10\ \mu g_{isoprene}/(g_{biomass} \cdot h)$ are considered low isoprene emitters. Deciduous and coniferous tree species that do not emit isoprene make up the two remaining forest biomass classes.

Natural vegetation areas of scrubland and grassland are noncanopy (nonforest) vegetation classes. For these vegetation classes, as well as the agricultural crop class, we determine hydrocarbon emissions using emission factors expressed as a function of land area. Thus, biomass density is not calculated directly.

6.1.2 Agricultural Crop Area

The agricultural crop data are from the ORNL Geoecology Crop Areas and Yields Data File. The crops included in the biogenic emissions inventory are: alfalfa, barley, corn, cotton, hay, oats, peanuts, potatoes, rice, rye, sorghum, soybeans, tobacco, wheat, and miscellaneous crops.

For Canada, neither the Matthews (1984) data set nor the Landsat (Page, 1980) data set assigned specific agricultural classes. Therefore, we made agricultural class assignments along latitude and longitude lines (see Figure 6.2), using cash crop data by Province from atlases. Where only the broad crop category "grain" was listed, the area was assigned 25% wheat and 75% oats, since oats, barley, and rye all have the same emission factor. Where wheat was specifically listed, the area was assigned 75% wheat and 25% oats. Where no specific crop was listed, the area was assigned to the miscellaneous crops class.

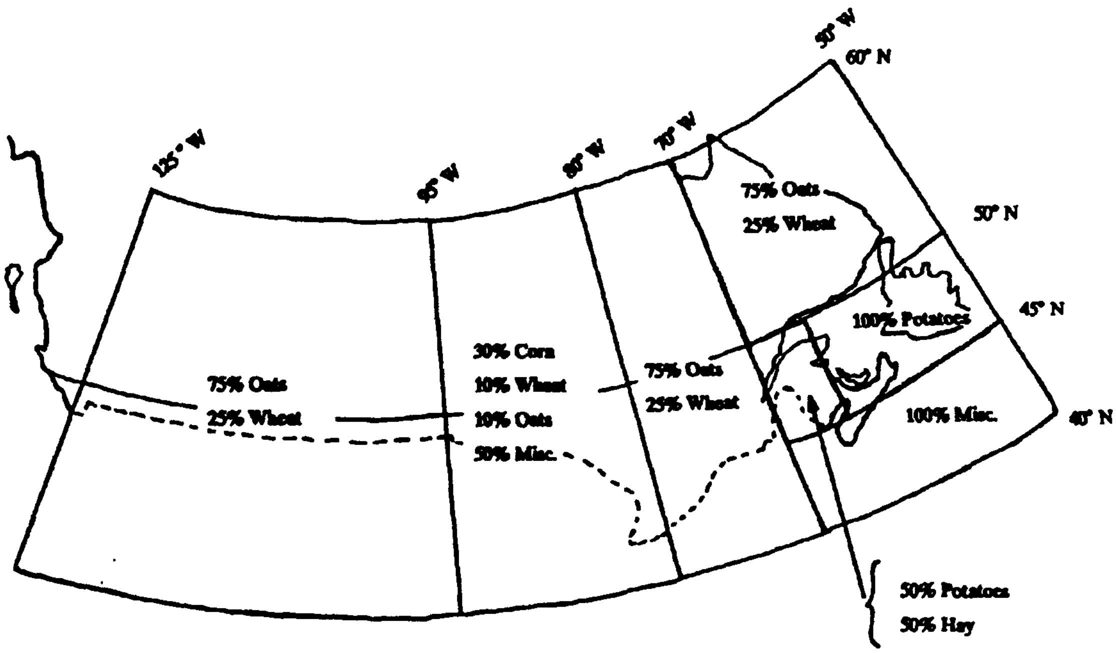

Figure 6.2. Canadian agricultural class assignments.

6.1.3 Urban Area

The ORNL Geoecology Land Areas Data File specifies urban, rural, road, water, and federal land areas. Urban areas include suburban areas if the same area has not been included as agricultural crops or natural vegetation.

To account for hydrocarbon emissions from grass and trees in an urban area, we used the results from two studies. Zimmerman (1979) showed that residential areas made up 14.6% of an urban area. Winer *et al.* (1983) showed that trees covered 9.7% of an urban area, and that ground cover comprised more than 17.1% of the area. For purposes of ROM modeling, we assumed that 20% of an urban area is covered by grasses; a further 20% is covered by trees, where this area is evenly distributed among oak, other deciduous, and coniferous categories.

Note that the Matthews data set does not define urban areas.

6.1.4 Water and Barren Areas

Water areas are also determined from the ORNL Geoecology Land Areas Data File (for the United States) and the Landsat data set (for Canada). Oceans and the Great Lakes are not included in the biogenic emissions inventory. However, smaller water areas such as lakes and rivers are included if the grid cell that they are in is not 100% water. Also included in the biogenic emissions inventory are any barren areas, such as tundra, ice, alpine meadows, and desert. Areas of water and barren land are used only in the reconciliation of the total area for the county and the grid cell.

6.2 ADJUSTMENT OF BIOMASS DENSITY

6.2.1 Growing Season

We used first and last frost dates to determine the growing season for vegetation. We acquired these data from (1) the ORNL Geoecology Growing Season Data File for counties in the United States, and (2) seasonal data for Canada (Kaplan, N., U.S. Environmental Protection Agency, personal communication, January 4, 1989). For simplicity, we assume that deciduous (i.e., nonconiferous) vegetation is at full biomass between the last frost date and the first frost date, and at zero biomass for the rest of the year. We assume that coniferous vegetation is at full biomass over the entire year.

6.2.2 Layering of Forest Biomass

We vary canopy biomass as a function of canopy height to simulate vertical forest structure. We assume that deciduous forest biomass classes (including high isoprene, low isoprene, and nonisoprene) have a canopy height of 15 m, while the coniferous nonisoprene biomass class has a canopy height of 20 m.

Table 6.3 presents the height and the estimated fraction of biomass for each layer (B. Lamb, Washington State University, personal communication, 1989).

TABLE 6.3. LAYERS FOR FOREST BIOMASS CLASSES

Layer variable name	Layer height, m	Fraction of biomass by layer
	DECIDUOUS (High Isoprene, Low Isoprene, Nonisoprene)	
LA1	3.75 - 5.25	0.00
LA2	5.25 - 6.75	0.00
LA3	6.75 - 8.25	0.02
LA4	8.25 - 9.75	0.11
LA5	9.75 - 11.25	0.22
LA6	11.25 - 12.75	0.35
LA7	12.75 - 14.25	0.22
LA8	14.25 - 15.00	0.09
	CONIFEROUS	
LA9	5.0 - 7.0	0.025
LA10	7.0 - 9.0	0.050
LA11	9.0 - 11.0	0.150
LA12	11.0 - 12.0	0.215
LA13	12.0 - 15.0	0.215
LA14	15.0 - 17.0	0.165
LA15	17.0 - 19.0	0.120
LA16	19.0 - 20.0	0.050

Source: B. Lamb, Washington State University, personal communication, 1989.

6.3 EMISSION FACTORS

Vegetation-specific emission factors are available for the following hydrocarbon compounds: isoprene, α-pinene, other identified monoterpenes (excluding α-pinene), and other unidentified hydrocarbons. Emission rates of the unidentified hydrocarbons can be estimated. The reactivity of the unidentified hydrocarbons is uncertain; we assume that about 95% of the unidentified compounds are reactive, and are evenly split between terpenoid and oxygenated compounds.

6.3.1 Canopy

Table 6.4 lists the compound-specific emission factors [in units of $\mu g_{compound}/(g_{biomass} \cdot h)$] for the forest biomass classes.

TABLE 6.4. CANOPY EMISSION FACTORS AT 30°C

Hydrocarbon compound	Forest biomass class	Canopy emission factor, [$\mu g_{compound}$/($g_{biomass}$ • h)]
Isoprene	Deciduous high isoprene	14.69
	Deciduous low isoprene	6.60
	Deciduous nonisoprene	0.00
	Coniferous nonisoprene	0.00
α-pinene	Deciduous high isoprene	0.13
	Deciduous low isoprene	0.05
	Deciduous nonisoprene	0.07
	Coniferous nonisoprene	1.13
Other identified monoterpenes	Deciduous high isoprene	0.11
	Deciduous low isoprene	0.05
	Deciduous nonisoprene	0.07
	Coniferous nonisoprene	1.29
Other unidentified hydrocarbons	Deciduous high isoprene	3.24
	Deciduous low isoprene	1.76
	Deciduous nonisoprene	1.91
	Coniferous nonisoprene	1.38

Source: Lamb *et al.*, 1987.

The canopy emission factors are standardized to 30°C using the temperature relationship of Tingey (1981). Each emission factor represents the geometric mean emission rate for a forest biomass class (Lamb *et al.*, 1990).

The compound-specific emission flux by vegetation class (oak, other deciduous, or coniferous) is the product of the forest biomass density for the vegetation class (from Table 6.2) and the canopy emission factor for the hydrocarbon compound (from Table 6.4), summed over the four forest biomass classes.

6.3.2 Noncanopy

Table 6.5 lists the noncanopy emission factors [$\mu g_{compound}/(m^2 \cdot h)$], and the hydrocarbon compound-specific emission composition (%) for the noncanopy vegetation classes.

Emission rates for a specific hydrocarbon compound can be calculated by multiplying the surface land area (for each vegetation class) by the appropriate emission factor and the fraction of hydrocarbon compound composition.

TABLE 6.5. NONCANOPY EMISSION FACTORS AT 30 °C AND ESTIMATED PERCENT COMPOSITION OF EMISSIONS

		Estimated emissions composition (%)			
Noncanopy vegetation class	Noncanopy emission factor, [$\mu g_{compound}/(m^2 \cdot h)$]	Isoprene	α-pinene	Other mono-terpenes	Other unidentified hydrocarbons
Natural Vegetation:					
Grass	281.0	20	25	25	30
Scrub*	189.0	20	25	25	30
Agricultural Crops:					
Alfalfa	37.9	50	10	10	30
Barley †	37.9	20	25	25	30
Corn	3,542.0	0	10	10	80
Cotton †	37.9	20	25	25	30
Hay	189.0	20	25	25	30
Oats †	37.9	20	25	25	30
Peanuts	510.0	20	25	25	30
Potatoes	48.1	20	25	25	50
Rice	510.0	20	25	25	30
Rye †	37.9	20	25	25	30
Sorghum	39.4	20	25	25	30
Soybeans	22.2	100	0	0	0
Tobacco	294.0	0	10	10	80
Wheat	30.0	50	10	10	30
Misc. crops †	37.9	20	25	25	30
Water: ‡					
Barren Area: ‡					

Source: Lamb *et al.*, 1987.
* Emission factor is assumed to equal the hay emission factor.
† Emission factor is assumed to equal the alfalfa emission factor.
‡ Used only in the reconciliation of land area.

6.4 ADJUSTMENT OF EMISSION FACTORS

6.4.1 Tingey Temperature and Solar Intensity Corrections

Several studies have shown the effects of temperature and solar intensity on hydrocarbon emissions. We adjust the gridded compound-specific emission factors for variations in temperature and solar intensity with Tingey's curves (Tingey, 1981). Tingey's laboratory work with slash pine and live oak has yielded logarithmic equations to describe the increase in isoprene emissions due to the combined effect of temperature and solar intensity, and the increase in nonisoprene emissions due to temperature only. These equations are listed below.

For isoprene emissions,

$$E_{adj} = E_{30} \cdot \frac{10^{\left\{\frac{a}{1+\exp[-b(T-c)]} - d\right\}}}{e}$$

where: E_{adj} is the adjusted emission factor at temperature T [$\mu g_{isoprene}/(g_{biomass} \cdot h)$], and

E_{30} is the emission factor at 30 °C [$\mu g_{isoprene}/(g_{biomass} \cdot h)$]

T is the hourly ambient temperature (°C), used as a surrogate for leaf temperature

Table 6.6 lists the equation coefficients *a*, *b*, *c*, *d*, and *e* for four levels of light intensity ($\mu E/m^2$, where μErepresents micro-einsteins, a unit of light energy). For light intensities not listed, we used linear interpolation to calculate the adjusted emission factors. Note that the biogenic emissions inventory system for the ROM uses cloud cover data to attenuate light intensity values on an hourly basis.

TABLE 6.6. ISOPRENE TEMPERATURE AND SOLAR INTENSITY ADJUSTMENT COEFFICIENTS

Light intensity [$\mu E/(m^2 \cdot s)$] *	Isoprene equation coefficient (unitless)				
	a	*b*	*c*	*d*	*e*
800	1.200	0.400	28.30	0.796	1.00
400	0.916	0.239	29.93	0.462	1.95
200	0.615	0.696	32.79	0.077	4.75
100	0.437	0.312	31.75	0.160	10.73

Sources: Tingey, 1981, and Pierce *et al.*, 1990a.
* μE represents micro-einsteins, a unit of light energy.

The coefficients for light intensity of 800 $\mu E/m^2$ were modified from Tingey's values to match a light intensity of 400 $\mu E/m^2$ for temperatures of less than 29°C. Also, the original Tingey equation expressed emissions in terms of $\mu g_{carbonmass}/dm^2$ leaf area; the isoprene equation presented above includes unit conversions (68/60 represents the ratio of isoprene mass to carbon mass; 1.205 is the number of grams of biomass per square decimeter of leaf area).

For nonisoprene emissions (α-pinene, other identified monoterpenes, and other unidentified hydrocarbons),

$$E_{adj} = E_{30} \cdot \exp(a\ [T - 30])$$

where: E_{adj} is the adjusted emission factor at temperature T [$\mu g_{nonisoprene}/(g_{biomass} \cdot h)$]

E_{30} is the emission factor at 30 °C [$\mu g_{nonisoprene}/(g_{biomass} \cdot h)$]

T is the hourly ambient temperature (°C), used as a surrogate for leaf temperature.

The emission factor in Tingey's equation was expressed in units of $\mu g_{carbonmass}/(g_{biomass} \cdot h)$. The nonisoprene equation presented above has been converted to units of $\mu g_{compound}/(g_{biomass} \cdot h)$; 136/120 represents the ratio of nonisoprene mass (as α-pinene) to carbon mass. Table 6.7 lists the coefficient a of the nonisoprene adjustment equation by hydrocarbon compound.

TABLE 6.7. NONISOPRENE TEMPERATURE ADJUSTMENT COEFFICIENTS

	Nonisoprene equation coefficient (unitless)
Hydrocarbon compound	a
α-pinene	0.067
Other identified monoterpenes (excluding α-pinene)	0.0739
Other unidentified hydrocarbons	0.0739

Source: Tingey, 1981; Pierce *et al.*, 1990a.

6.4.2 Layered Correction Factors for Forest Biomass Classes

A canopy model has been developed by the Laboratory for Atmospheric Research at Washington State University (Gay, 1987). It is used to adjust emission factors for the four forest biomass classes (deciduous high isoprene, deciduous low isoprene, deciduous nonisoprene, and coniferous nonisoprene).

Typical leaf biomass profiles are assumed for the deciduous and coniferous forest types (as discussed in Section 6.2.2); the leaf area indices corresponding to these biomass profiles are apportioned into eight vertical layers for each forest type.

The canopy model utilizes hourly meteorological data for the episode, including ambient temperature, solar radiation, relative humidity, and wind speed. Meteorological input data are assumed to represent the top of the canopy. Within each layer and for each of the two forest types, the canopy model uses an iterative approach to compute the leaf-radiation balance of a typical leaf's surface. Solar radiation is exponentially reduced through the layers with the rate being a function of the biomass distribution. The rate of solar attenuation increases more rapidly for the photosynthetically-active region of the solar spectrum than for the rest of the spectrum, since leaves preferentially absorb visible light (Baldocchi *et al.*, 1984).

Both the total solar spectrum and the visible spectrum subset are calculated over the eight levels of the hypothetical canopies. The calculated total solar radiation is used to compute the leaf temperature at each level using the radiation balance equation of Gates and Papian (1971).

The final output from this process consists of values of leaf temperatures and photosynthetically-active radiation for the eight layers in the two forest types. We then use these data when applying the Tingey correction factors.

6.5 CALCULATION OF BIOGENIC EMISSIONS

For the forest biomass classes, we multiply the layered biomass by the canopy emission factors to arrive at the layered standardized emissions. These emissions are then adjusted by the layered Tingey correction factors; we sum the results to produce canopy emissions.

For the noncanopy vegetation classes, we multiply the biomass area by the noncanopy emission factors to arrive at the standardized emissions. These emissions are then adjusted using the Tingey curves to produce noncanopy emissions. The canopy and noncanopy emissions are then summed for each grid cell.

6.6 QUALITY CONTROL

Quality control efforts by the EPA have focused on reconciling land area values. The sum of the areas allocated to all the vegetation classes in a county (natural vegetation, agricultural crops, urban, water, and barren areas) must equal the total area of the county. Similarly, the sum of the areas allocated to all the vegetation classes in one grid cell must equal the total area of the grid cell. Thus, we account for all the land area in a county or grid cell.

New or revised emission factors resulting from further studies of hydrocarbon emissions from vegetative species will be incorporated into the biogenic emissions inventory system.

6.7 BIOGENICS EMISSIONS PREPROCESSING: THE BIOGENIC EMISSIONS INVENTORY SYSTEM (BEIS)

There are 29 SAS® routines that render the data from the Geoecology database to a form suitable for input into the biogenic preprocessors for the ROM.[2] Documenting these routines is beyond the scope of this document; however, a User's Guide for the BEIS will be forthcoming.

6.7.1 SAS Routines to Process Biogenic Emissions

INIT.SAS	Sets up the SAS system options and defines the data set library to be used. Included into all of the SAS programs.
L08IN.SAS	Reads land-type area data (L08.DAT) into a SAS data set (L08.SSD).
REGIN.SAS	Creates a county list from the county-to-grid data set. Sums the pieces by county and stores along with the L08.DAT nonwater area. This data set is used to extract only those counties within the region.
INDEP.SAS	Corrects independent cities, mostly in Virginia, by removing the area from the county they are contained in.
A06IN.SAS	Reads crop data (A06.DAT) into a SAS data set (A06.SSD) and drops Alaska and Hawaii.
A06QA.SAS	Extracts and totals for counties in the region. Looks for zero agricultural areas, for agricultural areas greater than county areas, and for counties with yields but no matching areas.
C18IN.SAS	Reads frost dates (C18.DAT) into a SAS data set (C18.SSD).
C18QA.SAS	Extracts the counties in the region. Looks for missing dates or missing counties.
C18FIX.SAS	Sets up the frost dates for the Western U.S. counties that may be in the region.
D04IN.SAS	Reads natural vegetation data (D04.DAT) into a SAS data set (D04.SSD).
D04QA.SAS	Extracts and totals the counties in the region. Looks for counties with no natural vegetation (usually urban areas), and for counties with more natural vegetation than the area in the county.

2. SAS is a registered trademark of the SAS Institute, Inc., Cary, NC.

L08QA.SAS Extracts the counties in the region. Compares nonwater areas to county area from the FIPSCO.SSD data set; looks for counties with zero urban area.

REGROM.SAS Creates region data sets by extracting the counties in the region from A06.SSD, D04.SSD, L08.SSD, and C18.SSD - drops all unnecessary variables and merges western frost dates with C18.SSD data set.

DJQA.SAS Using the QA county total data sets (CTYA06.SSD, CTYD04.SSD, and CTYL08.SSD), compares nonwater area of county to the sum of the natural vegetation areas, the agricultural areas, and the urban areas. Looks for counties with large (>50%) amounts of land not accounted for; looks for counties with large (>50%) amounts of land over-accounted for; no corrections are made.

VEGIN.SAS Reads the Matthews vegetation data into a SAS data set (VEGTYPE.SSD).

CULTIN.SAS Reads the Matthews cultivation data into a SAS data set (CULTLND.SSD).

CANADA.SAS Using the gridded U.S. counties, determines the data cells that are partially in the U.S. and partially in Canada. This will be used to window out the Canadian data from the other data sets.

CANVEG1.SAS Calculates the area of cultivated land in the Canadian grid cells north of 55 degrees latitude from the Matthews data set. Assigns crop types to the cultivated land.

CANVEG2.SAS Calculates the area of natural vegetation in the Canadian grid cells north of 55 degrees latitude from the Matthews data set. Assigns the vegetation types to the forest and nonforest vegetation classes.

CANVEG3.SAS Calculates the amount of cultivated land and the amount of natural vegetation in the Canadian grid cells south of 55 degrees latitude from the Land Use data set. Assigns crop types to the cultivated land. Assigns the vegetation types to the forest and nonforest vegetation classes.

CANVEG4.SAS Combines the Canadian data sets to create gridded canopy and gridded noncanopy data sets for merging with the U.S. data sets.

FIXLND.SAS Adjusts vegetation areas: if land data is missing (or not accounted for), then save as "other" land ; if land is over accounted for, then reduces the natural vegetation evenly. Groups and totals vegetation areas into the following canopy types: oak, deciduous, coniferous, and into the following noncanopy types: scrub, grass, barren, water, urban, corn, wheat, oats, barley, sorghum, hay, soybeans,cotton, tobacco, potatoes, rice, peanuts, rye, and miscellaneous crops.

FIXQA.SAS	Compares adjusted vegetation areas to total area in the county. Looks for totals that remain over the county area. Reduces the following types evenly: oak, deciduous, coniferous, scrub, grass, and, if necessary, the crops.
GRDROM.SAS	Converts the county data for the U.S. to the NAPAP-size grid cells. Grids both the vegetation and the frost dates.
URBVEG.SAS	Using the gridded urban areas, adds 20% urban grass and 20% urban trees (evenly: oak, deciduous, and coniferous) to the gridded vegetation. Currently , this routine adds the urban trees to the other trees; later, these data will be processed as part of the noncanopy data.
CANC18.SAS	Creates gridded frost dates for the Canadian grid cells. Merges with the U.S. gridded frost dates.
GRWSEA.SAS	Determines growing months for nonconiferous vegetation.
CANVEG5.SAS	Merges the gridded Canadian canopy and noncanopy data sets with the gridded U.S. canopy and noncanopy data sets.
BIOFAC.SAS	Calculates monthly biomass factors for the canopy types (oak, deciduous, and coniferous) and calculates monthly biomass flags for the noncanopy types.

6.7.2 SAS Data Sets

A06.SSD	Crop data - entire U.S.
C18.SSD	Frost dates - Eastern U.S.
D04.SSD	Natural vegetation - entire U.S.
L08.SSD	Land areas - entire U.S.
CTYROM.SSD	County to grid cell conversion - entire U.S.
REGCTY.SSD	Counties in region

FIPSCO.SSD	County FIPS (Federal Information Processing Standards) codes and areas from various databases
CTYA06.SSD	QA - county total agriculture
CTYD04.SSD	QA - county total natural vegetation
CTYL08.SSD	QA - county areas
CHKLND.SSD	QA - combined total area in each county
WESTFRS.SSD	Frost dates - counties in Western U.S.
REGA06.SSD	Crop data - counties in region
REGC18.SSD	Frost dates - counties in region
REGD04.SSD	Natural vegetation - counties in region
REGL08.SSD	Land areas - counties in region
CANOPY.SSD	Canopy vegetation areas to be gridded
NONCNPY.SSD	Noncanopy vegetation areas to be gridded
REGGRD.SSD	County to grid cell conversion data for region
CANC18.SSD	Gridded Canadian frost dates
CULTLND.SSD	Matthews cultivation data
VEGTYPE.SSD	Matthews vegetation data
USA.SSD	Grid cells that are partially U.S., partially Canada
LNDUSE.SSD	Gridded landuse percentages - U.S. and Canada
CANVGG.SSD	Landuse data for grid cells in USA.SSD and for grid cells in Canada - south of 55 degrees latitude.
CANVGG2.SSD	Land use data for Canadian grid cells only - south of 55 degrees latitude.
CANCULT.SSD	Canadian portion of CULTLND.SSD - north of 55 degrees latitude
ANCROP.SSD	Canadian crops for cultivated areas - north of 55 degrees latitude

CANVEG.SSD	Canadian vegetation from VEGTYPE.SSD - north of 55 degrees latitude
CNATVEG.SSD	Canadian vegetation by vegetation class - north of 55 degrees latitude
CANCROP2.SSD	Canadian crops for agricultural areas - south of 55 degrees latitude
CNATVEG2.SSD	Canadian vegetation by vegetation class - south of 55 degrees latitude
CANNCPY.SSD	Canadian gridded noncanopy data
CANCNPY.SSD	Canadian gridded canopy data
GRDC18.SSD	Gridded frost dates - U.S. and Canada
GCNPY.SSD	Gridded canopy vegetation areas - U.S. and Canada
GNCNPY.SSD	Gridded noncanopy vegetation areas - U.S. and Canada
GURVEG.SSD	Gridded urban vegetation areas - U.S. and Canada
GRWFAC.SSD	Monthly growing season flags for nonconiferous vegetation
CNPBIOFC.SSD	Monthly canopy biomass factors
NCBIOFC.SSD	Monthly noncanopy biomass flags

6.8 PREPROCESSOR ROMREG

I. PREPROCESSOR FUNCTION

This preprocessor is a SAS routine that windows the National Acid Precipitation Assessment Program (NAPAP: Sellars *et al.*, 1985) canopy foliar biomass data to the boundaries of the ROMNET modeling domain, and renumbers the windowed grid cells to the local coordinates. The windowing code is as follows; it is shown so that you can window the NAPAP data to your modeling domain:

```
IF NAPAP COL<161 OR NAPAP COL>224 OR NAPAP ROW<69 OR NAPAP ROW>120 THEN DELETE;
COLADJ=160;     * romnet column adjustment;
ROWADJ=68;      * romnet row adjustment;
COL=COL-COLADJ;
ROW=ROW-ROWADJ;
```

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. CNPBIOFC.SSD

This file contains the NAPAP gridded monthly canopy foliar biomass density, classified as (1) the "biomass class" (high-, low-, or non-isoprenes-generating biomass density, as well as conifer biomass density), and (2) the forest class. CNPBIOFC.SSD is input using standard SAS statements; its parameters are described in Table 6.8.

TABLE 6.8. CNPBIOFC.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	MONTH		Real*8	Month as *MM*
4	$OAKHI_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	HI-b.d. in an oak forest
5	$OAKLI_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	LI-b.d. in an oak forest
6	$OAKNI_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	NI-b.d. in an oak forest
7	$OAKCF_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	Coniferous b.d. in an oak forest
8	$DECDHI_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	HI-b.d. in a deciduous forest
9	$DECDLI_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	LI-b.d. in a deciduous forest
10	$DECDNI_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	NI-b.d. in a deciduous forest
11	$DECDCF_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	Coniferous b.d. in a deciduous forest
12	$CONFHI_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	HI-b.d. in a coniferous forest
13	$CONFLI_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	LI-b.d. in a coniferous forest
14	$CONFNI_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	NI-b.d. in a coniferous forest
15	$CONFCF_{i,j,MM}$	$kg{\cdot}ha^{-1}$	Real*8	Coniferous b.d. in a coniferous forest

Note: *i* = column number, *j* = row number, *MM* = month; HI = high isoprenes, LI = low isoprenes, NI = nonisoprenes, b.d. = biomass density.

b. NCBIOFC.SSD

This file contains the NAPAP gridded monthly noncanopy (nonforest) biomass density as either "on" or "off." (If the noncanopy biomass density for a grid cell is "on," biogenic emissions for that grid cell are internally calculated in P27G.) NCBIOFC.SSD is input using standard SAS statements; its parameters are described in Table 6.9.

TABLE 6.9. NCBIOFC.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	MONTH		Real*8	Month as *MM*
4	$BIOFAC_{i,j,MM}$		Real*8	Noncanopy biomass density: 1 = on; 0 = off

Note: *i* =column number, *j* = row number, *MM* = month.

c. GCNPY.SSD

This file contains the gridded canopy (forest) land area. GCNPY.SSD is input using standard SAS statements; its parameters are described in Table 6.10.

TABLE 6.10. GCNPY.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	$OAK_{i,j}$	ha	Real*8	Oak forest
4	$DECD_{i,j}$	ha	Real*8	Deciduous forest
5	$CONF_{i,j}$	ha	Real*8	Coniferous forest

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

d. GNCNPY.SSD

This file contains the gridded noncanopy (nonforest) land area. GNCNPY.SSD is input using standard SAS statements; its parameters are described in Table 6.11.

TABLE 6.11. GNCNPY.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	$CRP\_MS_{i,j}$	ha	Real*8	Miscellaneous cropland
4	$CORN_{i,j}$	ha	Real*8	Corn
5	$WHEAT_{i,j}$	ha	Real*8	Wheat
6	$OATS_{i,j}$	ha	Real*8	Oats
7	$BARL_{i,j}$	ha	Real*8	Barley
8	$SORG_{i,j}$	ha	Real*8	Sorghum
9	$SOYBN_{i,j}$	ha	Real*8	Soybean
10	$COTT_{i,j}$	ha	Real*8	Cotton
11	$TOBAC_{i,j}$	ha	Real*8	Tobacco
12	$POTAT_{i,j}$	ha	Real*8	Potato
13	$HAY_{i,j}$	ha	Real*8	Hay
14	$ALFA_{i,j}$	ha	Real*8	Alfalfa
15	$PEANUT_{i,j}$	ha	Real*8	Peanut
16	$RICE_{i,j}$	ha	Real*8	Rice
17	$RYE_{i,j}$	ha	Real*8	Rye
18	$SCRUB_{i,j}$	ha	Real*8	Scrub land
19	$GRASS_{i,j}$	ha	Real*8	Grass land
20	$URB\_GRSS_{i,j}$	ha	Real*8	Urban grass land
21	$WATER_{i,j}$	ha	Real*8	Water

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

e. GUTREE.SSD

This file contains the gridded urban canopy (forest) land area. GUTREE.SSD is input using standard SAS statements; its parameters are described in Table 6.12.

TABLE 6.12. GUTREE.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	$OAK_{i,j}$	ha	Real*8	Oak forest
4	$DECD_{i,j}$	ha	Real*8	Deciduous forest
5	$CONF_{i,j}$	ha	Real*8	Coniferous forest

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

One nonstandard output file of windowed data is created for each nonstandard input file. Each output file contains identical parameters to the input file. The input and corresponding output file names are shown below; note that the output files are SAS data files.

Input file	Output file
CNPBIOFC.SSD	RMCNBFC.SSD
NCBIOFC.SSD	RMNCBFC.SSD
GCNPY.SSD	GCNPYRM.SSD
GNCNPY.SSD	GNCNPYRM.SSD
GUTREE.SSD	GUTREERM.SSD

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

SAS source files:	1	file	15	blocks
SAS INCLUDE files:	1	file	1	block
	2	files	16	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:24:53
Buffered I/O count:	336
Direct I/O count:	10347
Peak working-set size:	2585
Peak virtual size:	6164

C. Space Requirements: Log and Print Files

ROMREG.SAS	25 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 6.13 shows the input file and output file space requirements.

TABLE 6.13. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	CNPBIOFC.SSD	SAS data set	49656	1 year
	NCBIOFC.SSD	SAS data set	2984	1 year
	GCNPY.SSD	SAS data set	12520	1 year
	GNCNPY.SSD	SAS data set	13248	1 year
	GUTREE.SSD	SAS data set	2984	1 year
			81392	
Output	RMCNBFC.SSD	SAS data set	3616	1 year
	RMNCBFC.SSD	SAS data set	104	1 year
	GCNPYRM.SSD	SAS data set	416	1 year
	GNCNPYRM.SSD	SAS data set	968	1 year
	GUTREERM.SSD	SAS data set	104	1 year
			5208	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

None

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

ROMREG.COM follows:

```
$ ! ACCOUNTING CODE
$ SET DEFAULT MET53:[SCRATCH.BIOGEN]
$ !
$ ! ROMREG.COM
$ !
$ ASSIGN USER2$DISK:[VMQ.PREP]INIT.SAS   INIT
$ ASSIGN MET53:[SCRATCH.BIOGEN] SYS$SCRATCH
$ !
$ @USER2$DISK:[VMQ.PREP]SASSET ROMREG
$ SAS USER2$DISK:[VMQ.PREP]ROMREG
$ @USER2$DISK:[VMQ.PREP]SASTYPE
$ !
$ EXIT
```

(end of run stream)

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

A. Main Program

ROMREG.SAS

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

INIT.SAS

6.9 PREPROCESSOR NBIOMASS

I. PREPROCESSOR FUNCTION

This preprocessor is a SAS routine that generates monthly biomass statistics from the annual biomass density files that are produced by preprocessor ROMREG (Section 6.8). These biomass output data files are for three types of land cover: (1) forest canopy, with the biomass layered as a function of canopy height; (2) noncanopy biomass, e.g., agricultural lands, and (3) urban tree biomass.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. RMCNBFC.SSD

This file contains the NAPAP gridded monthly canopy foliar biomass density for the modeling domain, classified as (1) the "biomass class" (high-, low-, or non-isoprenes-generating biomass density, as well as conifer biomass density), and (2) the forest class. RMCNBFC.SSD is input using standard SAS statements; its parameters are described in Table 6.14.

TABLE 6.14. RMCNBFC.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	MONTH		Real*8	Month as *MM*
4	$OAKHI_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	HI-b.d. in an oak forest
5	$OAKLI_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	LI-b.d. in an oak forest
6	$OAKNI_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	NI-b.d. in an oak forest
7	$OAKCF_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	Coniferous b.d. in an oak forest
8	$DECDHI_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	HI-b.d. in a deciduous forest
9	$DECDLI_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	LI-b.d. in a deciduous forest
10	$DECDNI_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	NI-b.d. in a deciduous forest
11	$DECDCF_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	Coniferous b.d. in a deciduous forest
12	$CONFHI_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	HI-b.d. in a coniferous forest
13	$CONFLI_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	LI-b.d. in a coniferous forest
14	$CONFNI_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	NI-b.d. in a coniferous forest
15	$CONFCF_{i,j,MM}$	$kg \cdot ha^{-1}$	Real*8	Coniferous b.d. in a coniferous forest

Note: *i* = column number, *j* = row number, *MM* = month; HI = high isoprenes, LI = low isoprenes, NI = nonisoprenes, b.d. = biomass density.

b. RMNCBFC.SSD

This file contains the NAPAP gridded monthly noncanopy (nonforest) biomass density for the modeling domain, as either "on" or "off." (If the noncanopy biomass density for a grid cell is "on," i.e., there is noncanopy growth and metabolism, then biogenic emissions for that grid cell are internally calculated in P27G.) RMNCBFC.SSD is input using standard SAS statements; its parameters are described in Table 6.15.

TABLE 6.15. RMNCBFC.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	MONTH		Real*8	Month as *MM*
4	$BIOFAC_{i,j,MM}$		Real*8	Noncanopy biomass density: 1 = on; 0 = off

Note: *i* =column number, *j* = row number, *MM* = month.

c. GCNPYRM.SSD

This file contains the gridded canopy (forest) land area for the modeling domain. GCNPYRM.SSD is input using standard SAS statements; its parameters are described in Table 6.16.

TABLE 6.16. GCNPYRM.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	$OAK_{i,j}$	ha	Real*8	Oak forest
4	$DECD_{i,j}$	ha	Real*8	Deciduous forest
5	$CONF_{i,j}$	ha	Real*8	Coniferous forest

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

d. GNCNPYRM.SSD

This file contains the gridded noncanopy (nonforest) land area for the modeling domain. GNCNPYRM.SSD is input using standard SAS statements; its parameters are described in Table 6.17.

TABLE 6.17. GNCNPYRM.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	$CRP\_MS_{i,j}$	ha	Real*8	Miscellaneous cropland
4	$CORN_{i,j}$	ha	Real*8	Corn
5	$WHEAT_{i,j}$	ha	Real*8	Wheat
6	$OATS_{i,j}$	ha	Real*8	Oats
7	$BARL_{i,j}$	ha	Real*8	Barley
8	$SORG_{i,j}$	ha	Real*8	Sorghum
9	$SOYBN_{i,j}$	ha	Real*8	Soybean
10	$COTT_{i,j}$	ha	Real*8	Cotton
11	$TOBAC_{i,j}$	ha	Real*8	Tobacco
12	$POTAT_{i,j}$	ha	Real*8	Potato
13	$HAY_{i,j}$	ha	Real*8	Hay
14	$ALFA_{i,j}$	ha	Real*8	Alfalfa
15	$PEANUT_{i,j}$	ha	Real*8	Peanut
16	$RICE_{i,j}$	ha	Real*8	Rice
17	$RYE_{i,j}$	ha	Real*8	Rye
18	$SCRUB_{i,j}$	ha	Real*8	Scrub land
19	$GRASS_{i,j}$	ha	Real*8	Grass land
20	$URB\_GRSS_{i,j}$	ha	Real*8	Urban grass land
21	$WATER_{i,j}$	ha	Real*8	Water

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

e. GUTREERM.SSD

This file contains the gridded urban canopy (forest) land area for the modeling domain. GUTREERM.SSD is input using standard SAS statements; its parameters are described in Table 6.18.

TABLE 6.18. GUTREERM.SSD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Real*8	Column
2	ROW		Real*8	Row
3	$OAK_{i,j}$	ha	Real*8	Oak forest
4	$DECD_{i,j}$	ha	Real*8	Deciduous forest
5	$CONF_{i,j}$	ha	Real*8	Coniferous forest

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

a. GCNBIO*MM*.DAT

This file contains the monthly gridded canopy biomass data by layer; NBIOMASS merges data from GCNPYRM.SSD (the canopy land cover file) and RMCNBFC.SSD (the canopy biomass density file). The oak and deciduous forests are combined, and represented in terms of the magnitude of emissions of isoprenes. GCNBIO*MM*.DAT is written as an ASCII file; the SAS output statements are shown below, and the file's parameters are shown in Table 6.20. The leaf area indices that are used to calculate the layered biomass are shown in Table 6.19; these attempt to simulate an idealized canopy shapes for both a deciduous and a coniferous canopy.

```
FILE CNOUT;
PUT (COL ROW) (3.) (BIOHI1-BIOHI8 BIOLI1-BIOLI8 BIONI1-BIONI8
                    BIOCF1-BIOCF8) (12.3)
```

TABLE 6.19. LEAF AREA INDICES

Parm No.	Parm Name	Value	Data Type	Description
1	LAI1	0.00	Real*8	Deciduous/oak forest, layer 1, 3.75-5.25 m
2	LAI2	0.00	Real*8	Deciduous/oak forest, layer 2, 5.25-6.75 m
3	LAI3	0.02	Real*8	Deciduous/oak forest, layer 3, 6.75-8.25 m
4	LAI4	0.11	Real*8	Deciduous/oak forest, layer 4, 8.25-9.75 m
5	LAI5	0.22	Real*8	Deciduous/oak forest, layer 5, 9.75-11.25 m
6	LAI6	0.35	Real*8	Deciduous/oak forest, layer 6, 11.25-12.75 m
7	LAI7	0.22	Real*8	Deciduous/oak forest, layer 7, 12.75-14.25 m
8	LAI8	0.09	Real*8	Deciduous/oak forest, layer 8, 14.25-15.00 m
9	LAI9	0.025	Real*8	Coniferous forest, layer 1, 5.0-7.0 m
10	LAI10	0.050	Real*8	Coniferous forest, layer 2, 7.0-9.0 m
11	LAI11	0.150	Real*8	Coniferous forest, layer 3, 9.0-11.0 m
12	LAI12	0.215	Real*8	Coniferous forest, layer 4, 11.0-12.0 m
13	LAI13	0.215	Real*8	Coniferous forest, layer 5, 12.0-15.0 m
14	LAI14	0.165	Real*8	Coniferous forest, layer 6, 15.0-17.0 m
15	LAI15	0.120	Real*8	Coniferous forest, layer 7, 17.0-19.0 m
16	LAI16	0.050	Real*8	Coniferous forest, layer 8, 19.0-20.0 m

Source: B. Lamb, Washington State University, personal communication, 1989.

TABLE 6.20. GCNBIO*MM*.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Integer*4	Column
2	ROW		Integer*4	Row
3	$BIOHI1_{i,j}$	kg	Real*4	High isoprenes, layer 1
4	$BIOHI2_{i,j}$	kg	Real*4	High isoprenes, layer 2
5	$BIOHI3_{i,j}$	kg	Real*4	High isoprenes, layer 3
6	$BIOHI4_{i,j}$	kg	Real*4	High isoprenes, layer 4
7	$BIOHI5_{i,j}$	kg	Real*4	High isoprenes, layer 5
8	$BIOHI6_{i,j}$	kg	Real*4	High isoprenes, layer 6
9	$BIOHI7_{i,j}$	kg	Real*4	High isoprenes, layer 7
10	$BIOHI8_{i,j}$	kg	Real*4	High isoprenes, layer 8
11	$BIOLI1_{i,j}$	kg	Real*4	Low isoprenes, layer 1
12	$BIOLI2_{i,j}$	kg	Real*4	Low isoprenes, layer 2
13	$BIOLI3_{i,j}$	kg	Real*4	Low isoprenes, layer 3
14	$BIOLI4_{i,j}$	kg	Real*4	Low isoprenes, layer 4
15	$BIOLI5_{i,j}$	kg	Real*4	Low isoprenes, layer 5
16	$BIOLI6_{i,j}$	kg	Real*4	Low isoprenes, layer 6
17	$BIOLI7_{i,j}$	kg	Real*4	Low isoprenes, layer 7
18	$BIOLI8_{i,j}$	kg	Real*4	Low isoprenes, layer 8
19	$BIONI1_{i,j}$	kg	Real*4	Nonisoprenes, layer 1
20	$BIONI2_{i,j}$	kg	Real*4	Nonisoprenes, layer 2
21	$BIONI3_{i,j}$	kg	Real*4	Nonisoprenes, layer 3
22	$BIONI4_{i,j}$	kg	Real*4	Nonisoprenes, layer 4
23	$BIONI5_{i,j}$	kg	Real*4	Nonisoprenes, layer 5
24	$BIONI6_{i,j}$	kg	Real*4	Nonisoprenes, layer 6
25	$BIONI7_{i,j}$	kg	Real*4	Nonisoprenes, layer 7
26	$BIONI8_{i,j}$	kg	Real*4	Nonisoprenes, layer 8
27	$BIOCF1_{i,j}$	kg	Real*4	Coniferous forest, layer 1
28	$BIOCF2_{i,j}$	kg	Real*4	Coniferous forest, layer 2
29	$BIOCF3_{i,j}$	kg	Real*4	Coniferous forest, layer 3
30	$BIOCF4_{i,j}$	kg	Real*4	Coniferous forest, layer 4
31	$BIOCF5_{i,j}$	kg	Real*4	Coniferous forest, layer 5
32	$BIOCF6_{i,j}$	kg	Real*4	Coniferous forest, layer 6
33	$BIOCF7_{i,j}$	kg	Real*4	Coniferous forest, layer 7
34	$BIOCF8_{i,j}$	kg	Real*4	Coniferous forest, layer 8

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

b. GNCBIO*MM*.DAT

This file contains the monthly gridded noncanopy land cover data; NBIOMASS merges data from GNCPYRM.SSD (the noncanopy land cover file) and RMNCBFC.SSD (the noncanopy biofactor file), and calculates the noncanopy areas of growth (and therefore of biogenic emissions). GNCBIO*MM*.DAT is written as an ASCII file; the SAS output statements are shown below, and the file's parameters are shown in Table 6.21.

```
FILE NCOUT;
PUT (COL ROW) (3.) (ALFA SORG HAY SOYBN CORN POTAT TOBAC WHEAT COTT RYE
        RICE PEANUT BARL OATS SCRUB GRASS URB_GRASS CRP_MS WATER) (12.3)
```

TABLE 6.21. GNCBIO*MM*.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	NCCOL		Real*8	Column
2	NCROW		Real*8	Row
3	$ALFA_{i,j}$	ha	Real*8	Alfalfa
4	$SORG_{i,j}$	ha	Real*8	Sorghum
5	$HAY_{i,j}$	ha	Real*8	Hay
6	$SOYBN_{i,j}$	ha	Real*8	Soybean
7	$CORN_{i,j}$	ha	Real*8	Corn
8	$POTAT_{i,j}$	ha	Real*8	Potato
9	$TOBAC_{i,j}$	ha	Real*8	Tobacco
10	$WHEAT_{i,j}$	ha	Real*8	Wheat
11	$COTT_{i,j}$	ha	Real*8	Cotton
12	$RYE_{i,j}$	ha	Real*8	Rye
13	$RICE_{i,j}$	ha	Real*8	Rice
14	$PEANUT_{i,j}$	ha	Real*8	Peanut
15	$BARL_{i,j}$	ha	Real*8	Barley
16	$OATS_{i,j}$	ha	Real*8	Oats
17	$SCRUB_{i,j}$	ha	Real*8	Scrub land
18	$GRASS_{i,j}$	ha	Real*8	Grass land
19	$URB\_GRSS_{i,j}$	ha	Real*8	Urban grass land
20	$CRP\_MS_{i,j}$	ha	Real*8	Miscellaneous cropland
21	$WATER_{i,j}$	ha	Real*8	Water

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

c. GUTBIO*MM*.DAT

This file contains the monthly gridded urban tree biomass data; NBIOMASS merges data from GUTREERM.SSD (the urban tree land cover file) and RMCNBFC.SSD (the canopy biomass density file), and calculates the urban tree biomass in terms of the magnitude of emissions of isoprenes. GUTBIO*MM*.DAT is written as an ASCII file; the SAS output statements are shown below, and the file's parameters are shown in Table 6.22.

```
FILE UROUT;
PUT (COL ROW) (3.) (BIOHI BIOLI BIONI BIOCF) (12.3)
```

TABLE 6.22. GUTBIO*MM*.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	COL		Integer*4	Column
2	ROW		Integer*4	Row
3	$BIOHI_{i,j}$	kg	Real*4	High isoprenes, deciduous/oak trees
4	$BIOLI_{i,j}$	kg	Real*4	Low isoprenes, deciduous/oak trees
5	$BIONI_{i,j}$	kg	Real*4	Nonisoprenes, deciduous/oak trees
6	$BIOCF_{i,j}$	kg	Real*4	Coniferous trees

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

III. CONTROL CARDS

One control card (contained in file CARDIN.CPR) is used to input control data in the following format shown below. Table 6.23 defines the control card variables.

```
INFILE CARDIN;
LENGTH TIMEP 5. REGION $ 8.;
INPUT TIMEP REGION;
```

TABLE 6.23. CARDIN.CPR VARIABLES

Variable Name	Unit	Description
TIMEP		Julian scenario starting date as *YYDDD*
REGION		ROM modeling domain

Example:

88321 ROMNET1

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

SAS source files:	1	file	12	blocks
SAS INCLUDE files:	2	files	2	blocks
	3	files	14	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:05:28
Buffered I/O count:	449
Direct I/O count:	1716
Peak working-set size:	2909
Peak virtual size:	6307

C. Space Requirements: Log and Print Files

NBIOMASS.SAS	25 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 6.24 shows the input file and output file space requirements.

TABLE 6.24. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	RMCNBFC.SSD	SAS data set	3616	1 year
	RMNCBFC.SSD	SAS data set	104	1 year
	GCNPYRM.SSD	SAS data set	416	1 year
	GNCNPYRM.SSD	SAS data set	968	1 year
	GUTREERM.SSD	SAS data set	104	1 year
			5208	
Output	GCNBIO*MM*.DAT	Nonstd	1863	1 month
	GNCBIO*MM*.DAT	Nonstd	1138	1 month
	GUTBIO*MM*.DAT	Nonstd	172	1 month
			3173	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

None

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

NBIOMASS.COM for November follows:

```
$ ! ACCOUNTING CODE
$ !
$ ! *** run stream is in USER2$DISK:[VMQ.PREP]NBIOMASS.COM
$ !
$ SET VERIFY
$ ! assign input for NBIOMASS
$ SET DEFAULT MET53:[SCRATCH.BIOGEN]
$ ASSIGN WORK_SCRATCH:     SYS$SCRATCH
$ ASSIGN USER2$DISK:[VMQ.PREP]INIT.SAS                INIT
$ ASSIGN USER2$DISK:[VMQ.PREP]INIT2.SAS               INIT2
$ ! assign output for NBIOMASS
$ ASSIGN MET35:[SCRATCH.VMQ]GCNBIONO.DAT              CNOUT
$ ASSIGN MET35:[SCRATCH.VMQ]GNCBIONO.DAT              NCOUT
$ ASSIGN MET35:[SCRATCH.VMQ]GUTBIONO.DAT              UROUT
$ ! ASSIGN CONTROL CARD(S) FILE
$ ASSIGN USER2$DISK:[VMQ.PREP]CARDIN.CPR              CARDIN
$ !
$ TYPE CARDIN
$ SAS$STATUS:=="0"
$ !
$ SHOW SYMBOL SAS$STATUS
$ !
$ @USER2$DISK:[VMQ.PREP]SASSET NBIOMASS
$ SAS USER2$DISK:[VMQ.PREP]NBIOMASS
$ @USER2$DISK:[VMQ.PREP]SASTYPE
$ EXIT
```

(end of run stream)

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

A. Main Program

NBIOMASS.SAS

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

INIT.SAS
INIT2.SAS

SECTION 7

MISCELLANEOUS PREPROCESSORS AND OTHER PROGRAMS

This section describes seven preprocessing programs. The first two (ozone boundary conditions, Section 7.1, and CVCHEM, Section 7.2) pertain to the ROM's boundary conditions and initial conditions respectively. Section 7.3 describes how digital line data are retrieved from U.S. Geodata. Section 7.4 explains the inversion-decision paradigm (a component of processor P03G) as executed by preprocessor MASK. TEMPROC calculates the meteorology variables required by P26G, and is documented in Section 7.5. Section 7.6 shows you how to operate the PFMFCOPY program. Section 7.7 highlights SHOWMET, the graphics program that plots the output from the meteorology preprocessors.

7.1 PREPROCESSING OZONE BOUNDARY CONDITIONS

I. FUNCTION

These preprocessing steps generate boundary conditions of ozone concentrations. The input data consist of hourly ozone observations that you can obtain from the U.S. Environmental Protection Agency's (EPA) National Air Data Branch (Research Triangle Park, NC 27709; telephone (919)541-5583). In general, however, any valid set of ozone monitoring data will suffice.

We estimate daytime and nighttime average ozone concentrations for the southern and western boundaries of the modeling domain. Thus far, we have used *ad hoc* procedures to manipulate windowed data in a Lotus 1-2-3® spreadsheet.[1] We offer the methodology discussed below as a guideline to users who need to develop their own boundary conditions. We followed these steps for assembling ozone boundary data file in the ROMNET modeling domain. Note that we accomplish all the steps using 1-2-3; we then transfer the ASCII data from a PC to the VAX with Kermit-MS.[2]

---

1. Lotus and 1-2-3 are registered trademarks of Lotus Development Corporation.
2. Kermit is available from Kermit Distribution, Columbia University Center for Computing Activities, 612 West 115th St., New York, NY 10025.

- Identify the region of overlap along the model's boundary. This region includes ozone monitoring sites as shown in Figure 7.1. For the ROMNET domain, sites are selected from the following window: along the southern boundary, 35°N to 38°N; along the western boundary, 78.5°W to 87°W. Monitoring data for the northern boundary (Canada) and the eastern boundary (Atlantic Ocean) are not available for the ROMNET application. However, we encourage you to seek additional sources of ozone monitoring data.
- Select hourly observations from sites in the overlap region. We select only stations designated as rural in order to avoid nearby urban effects. We also limit our choice to stations below 1000 feet in elevation, because high elevation sites tend to represent ozone concentrations in layer 2 of the ROM. We obtain the raw data from EPA's Aerometric Information Retrieval System (AIRS). We recommend that you include data beginning a day prior to the start of the model simulation through the last day of the model simulation.
- Divide the hourly observations into daytime and nighttime groups. Daytime includes data from 1000 to 1559 local standard time (LST), while nighttime includes data from 2200 to 0359 LST. Data for other times are associated with transitional periods and are not used. For each site, we use concentrations for a given daytime or nighttime period only if at least five hours of data are valid.[3] This satisfies the EPA guidance of requiring 75% data recovery.
- Compute the nighttime average. The nighttime average at each site is the arithmetic mean of the valid nighttime hourly observations. Hourly concentrations below 20 ppb are considered invalid because they imply the occurrence of local scavenging processes. A nighttime average for each day is formed from the arithmetic mean of all valid site averages. Nighttime averages are not separated geographically because so few stations (usually fewer than 20) have mean nighttime concentrations <20 ppb. Consequently, nighttime averages are the same at each of the four boundaries.
- Compute the daytime average. This step is slightly more complex than the computation of nighttime averages because each boundary is considered separately. For the southern boundary of ROMNET, we average sites delimited by 35°N - 38°N and 75.8°W - 87°W for each daytime period. For the western boundary of ROMNET, we average sites that are delimited by 35°N - 44°N and 83°W - 87°W. Some overlap does occur in the southwestern corner of the domain, and we use sites in this area twice. We then assign the minimum value from the averages computed for the southern and western boundary to be the average for the northern and eastern boundaries. We assume the minimum value to be valid since these two boundaries are largely removed from ozone precursor emissions; thus, they tend to experience lower ozone concentrations.
- Assemble the data file. The data file consists of the Julian date as *YYDDD*, the starting hour (either 10 or 22 LST, for daytime and nighttime respectively), and four ozone concentrations (in ppb) for the northern, southern, eastern, and western boundaries.

3. An experienced air quality specialist determines data validity at the EPA, and some aspects of validity are discussed below.

Note that the above methodology should be considered preliminary. As more accurate methods are developed for estimating boundary conditions, we recommend that they should be used. Possible improvements include the use of larger-scale model to define boundary conditions for ROM, or the use of additional monitoring data. For ROM 2.1, we investigated methods for discriminating the spatial and temporal patterns in greater detail, but we found too much variability in the monitoring data to justify going beyond a day/night, south/west discrimination. We encourage you to seek out improved procedures for boundary conditions, and to incorporate them into the ROM.

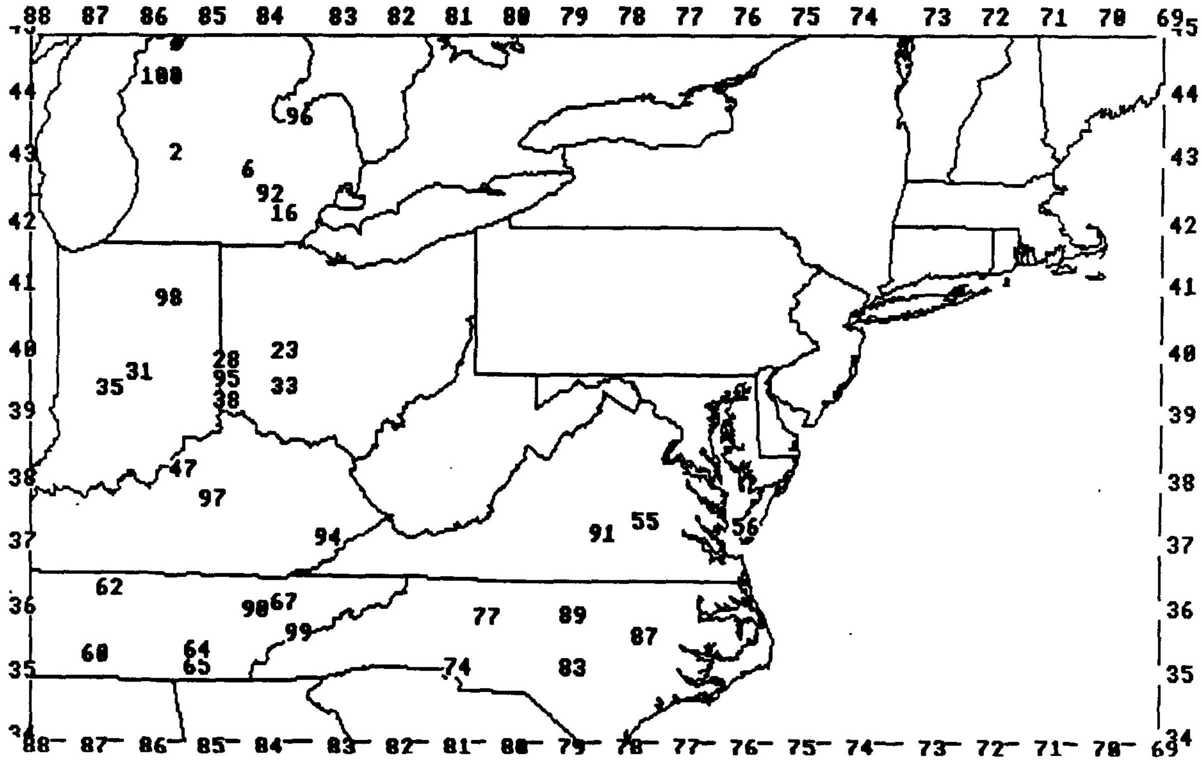

Figure 7.1. Ozone monitoring stations (ID numbers) for the determination of ROMNET ozone boundary conditions.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

Data are obtained as ASCII files from the EPA's Aerometric Information Retreival System (AIRS). The files can then be read into a LOTUS 1-2-3 spreadsheet. The exact format depends on the specific application; note that there is a great deal of flexibility as to how the data from AIRS can be formatted, and that we have not yet settled on a prescribed format.

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

This procedure is used to create OZONE, an ASCII file, that contains ozone boundary conditions at each of the boundaries of the modeling domain. OZONE is used by P24G, and its suggested parameters are shown in Table 7.1.

TABLE 7.1. OZONE PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	DESC1		Char*100	Header text record
2	DESC2		Char*100	Header text record
3	JDATE		Integer*4	Julian date as *YYDDD*
4	TIME	LST	Integer*4	Time as *HH*
5	N	ppm	Real*4	Ozone concentration, north boundary
6	S	ppm	Real*4	Ozone concentration, south boundary
7	E	ppm	Real*4	Ozone concentration, east boundary
8	W	ppm	Real*4	Ozone concentration, west boundary

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

N/A

B. Execution Time Requirements:

N/A

C. Space Requirements: Log and Print Files

N/A

D. Space Requirements: Input and Output Files

Table 7.2 shows the input file and output file space requirements.

TABLE 7.2. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	AIRS data	Nonstd	±50	1 month
Output	OZONE	Nonstd	±11	20 days

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

N/A

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

N/A

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

N/A

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

N/A

IX. INCLUDE FILES

N/A

7.2 PREPROCESSING ROM CHEMISTRY

The ROM reads initial conditions (IC's) and boundary conditions (BC's) that are generated by processors of the ROM processor network. Three chemistry processors (P21G, P23G, and P24G) create the IC's and BC's for the chemically-equilibrated species, using numerical schemes similar to the ones in the ROM itself. The three processors require rate constants for the chemical reactions of the Carbon-Bond 4.2 mechanism (CB4.2; Gery *et al.*, 1988). These rate constants, except for those associated with photolysis reactions, depend only on air temperature, air density, and specific humidity (water vapor concentration).

Rate constants can be set for each hour of the simulation period by using hourly weather observations, but our examination of the files produced by the three chemistry processors shows that this level of detail is unnecessary. Except for ozone, all species concentrations for IC's, BC's, and C_∞ (the top of layer 3) are considered to be at tropospheric background levels. Additionally, in all cases the effect of the IC's proved to be negligible after the first day of simulation. Thus, when you plan start dates of scenarios, you should set back the start date and time of an episode's simulation period to counter any errors that might be introduced by the IC's. BC's affect the simulation results within the model's domain only over a few cells deep, so their overall effect is minimal. Finally, the C_∞ condition affects layer 3 only when there is tropospheric air entrained from above into the layer. For these reasons, regional estimates of air temperature, air density, and specific humidity are sufficient for estimating the rate constants for the chemistry processors.

Since we run the model only during summer conditions, we selected an ideal day for ozone formation to estimate values of these parameters. For 1980, we chose July 20, which occured during an anticyclonic stagnation episode in the Northeast U.S.

Two sets of rate constants require calculation, consisting of daytime values and nighttime values. For each time period, there are three sets of rate constants that represent the three model layers. The rate constants are computed from only daytime and nighttime temperatures, air density, and specific humidity, values of which are needed for all three layers. The values that we used for these meteorological parameters were those of 12:00 LST (representing daytime values) and 20:00 LST (representing nighttime values), and were derived from a simulation that assumed 12 hours of light (at a constant 12:00 LST radiation intensity) and 12 hours of dark (at a constant radiation intensity equal to 0.001 of the 12:00 value).

To further simplify the process, we averaged the daytime and nighttime values for the air densities and specific humidities. Thus, the only day/night variation is in the air temperature field.

The regional estimates of air temperature, air density, and specific humidity that we use in the ROM are shown in Table 7.3.

TABLE 7.3. REGIONAL ESTIMATES OF METEOROLOGICAL PARAMETERS, SUMMER CONDITIONS

Layer	Air temperature Day (K)	Night (K)	Air density mol·ppm·m^{-3}	Specific humidity ppm
1	305	304	3.8×10^{-5}	2.3×10^{4}
2	298	299	3.6×10^{-5}	1.9×10^{4}
3	287	294	3.4×10^{-5}	1.4×10^{4}

All grid cells within a layer are assigned the values of the parameters for that layer.

PREPROCESSOR CVCHEM

I. PREPROCESSOR FUNCTION

This preprocessor computes the rate constants for each of the Carbon-Bond 4.2 reactions for given air temperature, air density, and specific humidity regimes in each model layer (refer to Section 7.2); note that the preprocessor assumes a constant solar radiation intensity over both daytime and nighttime. These rate constants are used only as initial and boundary conditions within the preprocessor network, i.e., in P21G, P23G, and P24G.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. METCOND.DAT

METCOND.DAT contains the air temperature, air density, and specific humidity meteorological parameters that are needed to compute the rate constants. The values of each of these parameters are averages over the grid by layer, and, for air temperatures, determined for both day- and nighttime (see Section 7.2). The READ and FORMAT statements for METCOND.DAT are listed below, and its parameters are shown in Table 7.4.

```
      DO 103 IDORN=1,2
        READ  (JMET,1007) NORD
1007    FORMAT (A8)
        DO 103 ILAY=1,NLAY
          READ  (JMET,1007) INLAY
          READ (JMET,1009) TEMP, DENSTY, H2OCON
1009      FORMAT (F9.0,1X,F9.0,1X,F9.0)
103   CONTINUE
```

TABLE 7.4. METCOND.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	NORD		Char*8	Night or day indicator
2	INLAY		Char*8	Layer name
3	$TEMP_{i,L}$	K	Real*4	Air temperature
4	$DENSTY_{i,L}$	mol·ppm·m^{-3}	Real*4	Air density
5	$H20CON_{i,L}$	ppm	Real*4	Specific humidity

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

b. PHOTORATES.DAT

This file contains the rate constants for the five photolysis reactions of the CB4.2 mechanism that form the basis of all 11 photolytic reactions of the mechanism. The photolytic rate constants as a function of solar zenith angle of the five species were calculated elsewhere (see Gery *et al.*, 1988), and are based on the following formulation:

$$A \int_{L_b}^{L_t} \int_{\lambda_1}^{\lambda_2} ([SI]_{\lambda,\theta,L} \quad [MA]_\lambda \quad [DP]_\lambda) d\lambda dL$$

where SI = solar intensity, MA = molecular absorption, DP = decomposition probability, L = layer height, λ = wavelength, θ = zenith angle, and A = the cloud attenuation factor, a function of cloud type and cloud cover extent. The READ and FORMAT statements for PHOTORATES.DAT are listed below, and Table 7.5 shows the file's parameters.

```
      DO 103 IDORN=1,2
        READ  (JRATE,1007) NORD
1007    FORMAT (A8)
        DO 103 ILAY=1,NLAY
          READ  (JRATE,1007) INLAY
          READ (JRATE,1019) RK(1), RK(9), RK(38), RK(39),
     &                      RK(45)
1019      FORMAT (5E15.0)
103   CONTINUE
```

TABLE 7.5. PHOTORATES.DAT PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	NORD		Char*8	Night or day indicator
2	INLAY		Char*8	Layer name
				Reaction rate constants for:
3	$RK(1)_{i,L}$	min^{-1}	Real*4	NO_2 photolysis
4	$RK(9)_{i,L}$	min^{-1}	Real*4	O_3 photolysis → O^1D
5	$RK(38)_{i,L}$	min^{-1}	Real*4	Formaldehyde photolysis → CO + HO_2
6	$RK(39)_{i,L}$	min^{-1}	Real*4	Formaldehyde photolysis → CO only
7	$RK(45)_{i,L}$	min^{-1}	Real*4	Higher aldehydes photolysis

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

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

RATECB4-2.DAT contains the rate constants for the reactions of the Carbon-bond 4.2 mechanism. The rates are adjusted to the input meteorological parameters. The WRITE and FORMAT statements for RATECB4-2.DAT are listed below, and its parameters are shown in Table 7.6.

```
      WRITE (FRATE,1004) CDATE, CTIME, DSKFIL
1004  FORMAT ('FILE CREATED ON DATE: ',A8,' AT TIME: ',A8,' BY ',
     &        A50)
      WRITE (FRATE,1006)
1006  FORMAT ('THIS FILE OF RATE CONSTANTS IS VALID FOR',
     &        'SUMMERTIME SCENARIOS')

      DO 103 IDORN=1,2
        WRITE (FRATE,1007) NORD
1007    FORMAT (A8)
        DO 103 ILAY=1,NLAY
          WRITE (FRATE,1007) INLAY
          WRITE (FRATE,1021) (RK(I), I=1,NCB42)
1021      FORMAT (5E15.4)
103   CONTINUE
```

TABLE 7.6. RATECB4-2.DAT 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	DSKFIL		Char*50	Version name of CVCHEM
4	NORD		Char*8	Night or day indicator
5	INLAY		Char*8	Layer name
6	$RK(1)_{i,L}$	min^{-1}	Real*4	Rate constant, species 1
.	.		.	.
.	.	or	.	.
.	.		.	.
88	$RK(83)_{i,L}$	$ppm^{-1} \cdot min^{-1}$	Real*4	Rate constant, species 83

Note: i = 1 (day) or 2 (night) indicator; L = layer. Units depend on the number of reacting gases.

III. CONTROL CARDS

None

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	27	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	2	files	15	blocks
Executable file:	1	file	18	blocks
	5	files	60	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	00:00:03
Buffered I/O count:	156
Direct I/O count:	139
Peak working-set size:	559
Peak virtual size:	2650

C. Space Requirements: Log and Print Files

CVCHEM.LOG	6 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 7.7 shows the input file and output file space requirements.

TABLE 7.7. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PHOTORATES.DAT	Nonstd	3	N/A
	METCOND.DAT	Nonstd	3	N/A
			6	
Output	RATECB4-2.DAT	Nonstd	18	N/A

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

CVCHEM.LNK follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[SJR.CHEMISTRY]CVCHEM.LNK
$ ON WARNING THEN EXIT
$ ASSIGN MLAB: SYS$PRINT
$ MAPFILE = "MET1:[SO1S]CVCHEM.MAP"
$ EXEFILE = "USER2$DISK:[SJR.CHEMISTRY]CVCHEM.EXE"
$ ASSIGN/USER_MODE USER2$DISK:[SJR.CHEMISTRY]CVCHEM.OBJ            A1
$ ASSIGN/USER_MODE MET24:[ROMOBJ.UTILIO]ADATE.OBJ                  A2
$ ASSIGN/USER_MODE USER2$DISK:[SJR.CHEMISTRY]RKUPDTCB4-2.OBJ       A3
$ LINK -
  /EXECUTABLE = USER2$DISK:[SJR.CHEMISTRY]CVCHEM.EXE -
  /MAP = MET1:[SO1S]CVCHEM.MAP -
  A1, -
  A2, -
  A3
$ PRINT -
  /NOHEADER -
  /NOFEED -
  /NOTIFY -
  MET1:[SO1S]CVCHEM.MAP
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

CVCHEM.COM follows:

```
$ ! ACCOUNTING CODE
$ ! Runstream is in file USER2$DISK:[SJR.CHEMISTRY]CVCHEM.COM
$ ASSIGN MLAB SYS$PRINT
$ ! Assign files
$ ASSIGN USER2$DISK:[SJR.CHEMISTRY]PHOTORATES.DAT   FOR007
$ ASSIGN USER2$DISK:[SJR.CHEMISTRY]METCOND.DAT      FOR009
$ ASSIGN USER2$DISK:[SJR.CHEMISTRY]RATECB4-2.DAT    FOR008
```

```
$ !
$ RUN USER2$DISK:[SJR.CHEMISTRY]CVCHEM.EXE
$ SHOW PROCESS/ACCOUNTING
```

(end of run stream)

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

A. Main Program

CVCHEM

B. Subroutines

RKUPDTCB4-2

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

B. Functions

None

IX. INCLUDE FILES

None

7.3 PREPROCESSING ROADWAY DIGITAL LINE DATA

7.3.1 PREPROCESSOR LINE_C (EMISSIONS)

I. PREPROCESSOR FUNCTION

This preprocessor copies to disk the line-source road data from Digital Line Graph tapes. These data are needed for the ROM's plume volume fraction calculation that parameterizes the subgrid-scale effects of line sources (and major point sources). The data on these tapes is small-scale (1:2 000 000), and termed *US GeoData* by the United States Geological Survey (USGS). The data tapes are available from the User Services Section, The National Cartographic Information Center, USGS, 507 National Center, Reston, VA 22092. A companion technical bulletin exists (Domaratz *et al.*, 1983).

The US GeoData are organized by region. To assist you in ascertaining your requirements, we list below the states in each region; the USGS region nomenclature is shown in parentheses.

- Region 1: New England States and NY (Northeastern States).
- Region 2: NJ, DE, MD, PA, OH, WV, VA (Middle Atlantic States).
- Region 3: NC, SC, GA (Southeastern States).
- Region 4: FL (Florida).
- Region 5: TN, AL, MS, LA, AR (Southern Mississippi Valley States).
- Region 6: KY, IN, IL, IA, MO (Central Mississippi Valley States).
- Region 7: MI, WI, MN (Northern Great Lakes States).
- Region 8: Southern TX (Southern Texas).
- Region 9: Northern TX, OK (Southern Plains States).
- Region 10: KS, CO, NB (Central Plains States).
- Region 11: SD, ND, Eastern and Central MT, WY (Northern Plains States).
- Region 12: NM, AZ (Arizona and New Mexico).
- Region 13: Southern CA (Southern California).
- Region 14: Central and Northern CA, NV, UT (Central Pacific States).
- Region 15: OR, WA, ID, Western MT (Northwestern States).
- Regions 16, 17, 18, 19, and 20: AK (Southeastern, Central, Northern, Southwestern Alaska, Aleutian Islands).
- Region 21: HA (Hawaiian Islands).

You will note that more than one region's data is necessary to cover the ROMNET domain. When you order US GeoData, they are accompanied by a list of the attributes of the output data set to assist you in determining the contents of each file on the tape. Note that roads and trails data are contained in a separate file from railroads data.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

Each record of a selected file is read from the tape using the VATAPE program, which is specific to the VAX-VMS environment. Table 7.8 shows the tape parameter.

TABLE 7.8. US GEODATA TAPE PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	LINES		Char*20	Line-by-line copy of selected tape files

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

LINE_C generates a sequential disk file (e.g., RAWLINE3.DAT in the run stream below) that is the product of the concatenation of the input tape files; the following WRITE and FORMAT statements are used:

```
    WRITE (10,300) LINES(IL)
300 FORMAT (1X,A20)
```

The characteristics and description of LINES in the output files are identical to those of the nonstandard input files.

III. CONTROL CARDS

One control card is used to input the tape files to be copied to disk, in the format shown below. Table 7.9 defines the control card variables.

NFILES,NFILE(1),NFILE(2),...,NFILE(NFILES)

TABLE 7.9. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Data Type	Description
1	NFILES		Integer*4	Number of tape files to extract
2	$NFILE_1$		Integer*4	First sequential file no. to extract
	.		.	.
	.		.	.
	.		.	.
N	$NFILE_{NFILES}$		Integer*4	Final sequential file no. to extract

Example

7 1,3,5,7,9,11,13

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	20	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	2	files	40	blocks
Executable file:	1	file	80	blocks
	5	files	140	blocks

B. Execution Time Requirements

The charged CPU time and I/O counts depend on the number and size of the files extracted from tape.

	VAX 8650
Charged CPU time (hh:mm:ss):	±00:01:00

C. Space Requirements: Log and Print Files

LINE_C.LOG:	±20 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 7.10 shows the input file and output file space requirements.

TABLE 7.10. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	None		0	
Output	RAWLINE3.DAT	Nonstd	±20000	N/A

E. Space Requirements: Tape Files

Table 7.11 shows the input tape file space requirements.

TABLE 7.11. INPUT TAPE FILE SPACE REQUIREMENTS

File Group	Tape No.	File No.	Record Length (in bytes)	Block Size (in bytes)
Input	B20074	All	20	1040

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

LINE_C.LNK follows:

```
$ LINK USER2$DISK:[BRO.LABDATA]LINE_C.OBJ, -
  USER2$DISK:[BRO.TAPES]VATAPE.OBJ
```

(end of link stream)

VI. SAMPLE RUN STREAM COMMAND FILE

LINE_C.RUN follows:

```
! ACCOUNTING CODE
SET VERIFY
! RUN STREAM IS IN FILE [BRO.LABDATA]LINE_C.RUN
ASSIGN TTA7: SYS$PRINT
ON WARNING THEN EXIT
TRY:
ON ERROR THEN CONTINUE
ALLOCATE MFA0:
IF $STATUS THEN GOTO OKAY
WAIT 00:05
GOTO TRY
OKAY:
SHOW TIME
REQUEST/TO=TAPES/REPLY -
PLEASE MOUNT TAPE B20074 ON DEVICE MFA0:"
MOUNT /FOREIGN/NOWRITE/BLOCKSIZE=1040/RECORDSIZE=20/DENSITY=6250 -
MFA0: B20074
SHOW DEVICE /FULL MFA0:
ASSIGN MET1:[RAWDATA2]RAWLINE3.DAT     FOR010
```

```
RUN [BRO.LABDATA]LINE_C.EXE
7 1,3,5,7,9,11,13
SET MAGTAPE/REWIND MFA0:
DISMOUNT MFA0:
DEALLOCATE MFA0:
SET PROT=(S:RE,O:RE,G,W) MET1:[RAWDATA2]RAWLINE3.DAT
```

(end of run stream)

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

A. Main Program

LINE_C

B. Subroutines

VATAPE

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

7.3.2 PREPROCESSOR LINE_CR (EMISSIONS)

I. PREPROCESSOR FUNCTION

This preprocessor windows the line-source road disk files that are produced by LINE_C (Section 7.3.1) to regional boundaries that you define in the control cards. LINE_CR also reformats these data so that they can be read by processor P13G.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

LINE_C generates one nonstandard file, RAWLINES3.DAT, that consists of header records and data records. The header records of this file are read by the following READ and FORMAT statements, and the header parameters are shown in Table 7.12.

```
     READ(10,200,END=4) LID,IRNK,NPTS,ICODE
200  FORMAT(1X,I7,I2,I6,I5)
```

TABLE 7.12. RAWLINES3.DAT HEADER RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	LID		Integer*4	Sequential line number
2	$IRNK_l$		Integer*4	Line attribute minor code (rank)
3	$NPTS_l$		Integer*4	No. of data records that follow
4	$ICODE_l$		Integer*4	Line attribute major code

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).

Data records are read by the following READ and FORMAT statements, and the data record parameters are shown in Table 7.13.

```
     READ(10,201) LTD,LTM,LTS,LND,LNM,LNS
201  FORMAT(1X,I2,I2,I2,1X,I3,I2,I2)
```

TABLE 7.13. RAWLINES3.DAT DATA RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	LTD_{NPTS}	° N	Integer*4	Node latitude, degrees
2	LTM_{NPTS}	′ N	Integer*4	Node latitude, minutes
3	LTS_{NPTS}	″ N	Integer*4	Node latitude, seconds
4	LND_{NPTS}	° W	Integer*4	Node longitude, degrees
5	LNM_{NPTS}	′ W	Integer*4	Node longitude, minutes
6	LNS_{NPTS}	″W	Integer*4	Node longitude, seconds

B. Output Files

1. Standard Output Files

None

2. Nonstandard Output Files

LINE_CR generates one nonstandard file, LINE1.DAT, that consists of header records and data records. The header records are written by the following WRITE and FORMAT statements, and the header parameters are shown in Table 7.14.

```
    WRITE(11,300) LID,NN,ICODE,IRNK
300 FORMAT(1X,I6,I6,I5,I2.2)
```

TABLE 7.14. LINE1.DAT HEADER RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	LID		Integer*4	Sequential line number
2	NN_l		Integer*4	No. of segments in this line
3	$ICODE_l$		Integer*4	Line attribute major code
4	$IRNK_l$		Integer*4	Line attribute minor code (rank)

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).

Data records are written by the following WRITE and FORMAT statements, and the data record parameters are shown in Table 7.15.

```
    WRITE(11,301) XLAT(N),XLON(N)
301 FORMAT(1X,F8.4,F9.4)
```

TABLE 7.15. LINE1.DAT DATA RECORD PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$XLAT_1$	° N	Real*4	Latitude of first node of line segment
2	$XLON_1$	° W	Real*4	Longitude of first node of line segment
.	.			.
.	.			.
.	.			.
NN	$XLAT_{NN}$	° N	Real*4	Latitude of NN^{th} node of line segment
NN	$XLON_{NN}$	° W	Real*4	Longitude of NN^{th} node of line segment

III. CONTROL CARDS

One control card is used to input the tape files to be copied to disk, in the format shown below. Table 7.16 defines the control card variables.

LATMIN,LATMAX,LONGMIN,LONGMAX

TABLE 7.16. CONTROL CARD VARIABLES

Parm No.	Parm Name	Unit	Data Type	Description
1	LATMIN	° N	Real*4	Southern domain boundary
2	LATMAX	° N	Real*4	Northern domain boundary
3	LONGMIN	° W	Real*4	Eastern domain boundary
4	LONGMAX	° W	Real*4	Western domain boundary

Example

36.3333 45. 69. 85.

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	1	file	10	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	1	file	10	blocks
Executable file:	1	file	20	blocks
	3	files	40	blocks

B. Execution Time Requirements:

	VAX 8650
Charged CPU time (hh:mm:ss):	±00:01:00

C. Space Requirements: Log and Print Files

LINE_CR.LOG:	±20 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 7.17 shows the input file and output file space requirements.

TABLE 7.17. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	RAWLINES3.DAT	Nonstd	Varies	N/A
Output	LINES1.DAT	Nonstd	±12000	N/A

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

LINE_CR.LNK follows:

```
LINK [BRO.ROMDATA]LINE_CR.OBJ
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

LINE_CR.RUN follows:

```
! ACCOUNTING CODE
SET VERIFY
! RUN STREAM IS IN FILE [BRO.ROMDATA]LINE_CR.RUN
ASSIGN TTA7: SYS$PRINT
ON WARNING THEN EXIT
ASSIGN MET1:[RAWDATA]RAWLINE3.DAT     INFILE
ASSIGN MET1:[RAWDATA2]LINE1.DAT       OUTFILE
RUN [BRO.ROMDATA]LINE_CR.EXE2 36.3333 45. 69. 85.
SET PROT=(S:RWED,O:RE,G:RE,W) OUTFILE
```

(end of run stream)

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

A. Main Program

LINE_CR

B. Subroutines

None

C. Functions

None

D. Block Data Files

None

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

A. Subroutines

None

B. Functions

None

IX. INCLUDE FILES

None

7.4 PREPROCESSOR MASK (METEOROLOGY AND EMISSIONS)

I. PREPROCESSOR FUNCTION

MASK creates PF195, which contains the weightings that are used by processor P03G in its nocturnal inversion-decision scheme.[4] MASK is run once per emissions data set, i.e., once per scenario. The weights are assigned in the following manner: the (background) weight for grid cells over large bodies of water is 0.0; cells on land, or bordering land, have weight 1.0, unless they are among, or are adjacent to,[5] cells that contain one of the top *N* area-source and point-source emitters;[6] these cells are assigned a weight of 2.0. MASK determines which are the top *N* emitters during a scenario by summing and ranking the hourly major point- and area-source emissions.

The rationale for the inversion scheme in P03G is based on the fact that the ROM is most concerned with pollution transport, and that this transport is dependent on the presence or absence of a temperature inversion. If a nocturnal inversion exists (and persists into the morning hours), the implication for all but the tallest point sources is that the pollutants will be trapped below the inversion, and no advection takes place. The weightings afforded by MASK balance the inversion/no inversion decision in P03G more uniformly between the larger rural land areas with no significant point- or area-source emitters, and the higher-emitting but smaller land area of, e.g., the Northeast Corridor in the ROMNET domain.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

MASK uses two standard input files:

PF118 from P25G
PF145 from P14G

4. Further information on PF and MF *standard files* can be found in Part 2, Section 1; information on setting up the standard files database is contained in Part 2, Section 2.

5. We use an eight-directional bordering, one cell deep.

6. *N* is an integer (usually equal to 100) that separates the top *N* emitters from all other area and point sources; it is user-defined from SYS$INPUT, i.e., in the run stream.

2. Nonstandard Input Files

MASK uses one nonstandard binary file, PTE3.DAT, that is generated by P14G, and which contains the major point-source emissions data. PTE3 is read by the function RDPTE3, and its parameters are shown in Table 7.18.

TABLE 7.18. 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	Emissions rate for aldehydes
11	$SEK(2)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for CO
12	$SEK(3)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for ethene
13	$SEK(4)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for formaldehyde
14	$SEK(5)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for isoprenes
15	$SEK(6)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for NO
16	$SEK(7)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for NO_2
17	$SEK(8)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for olefins
18	$SEK(9)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for paraffins
19	$SEK(10)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for toluene
20	$SEK(11)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for xylene
21	$SEK(12)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for nonr.
22	$SEK(13)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for methane
23	$SEK(14)_{p,h}$	$mol \cdot h^{-1}$	Real*4	Emissions rate for methanol
24	$SEK(15)_{p,h}$	$kg \cdot h^{-1}$	Real*4	Emissions 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.

B. Output Files

1. Standard Output Files

PF195

MASK produces one standard output file that contains the gridded weights surrounding the top N emitters in the modeling domain. It is used by P03G. The PF195 parameter is shown in Table 7.19.

TABLE 7.19. PF195 PARAMETER

Parm No.	Parm Name	Unit	Data Type	Description
1	$INMASK_{i,j}$		Real*4	Gridded weighting: 0 = large water body 1 = land not among/adjacent to top *N* emitters 2 = land among/adjacent to top *N* emitters

2. Nonstandard Output Files

None

III. CONTROL CARDS

One control card is used to input control data in the format shown below. Table 7.20 defines the control card variable.

NN

TABLE 7.20. CONTROL CARD VARIABLE

Variable Name	Unit	Description
NN		Size of the subset of largest emitters belonging to the set of area- and point-source emitters

Example:

100

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	2	files	42	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	2	files	15	blocks
Executable file:	1	file	66	blocks
	5	files	123	blocks

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

	VAX 8650
Charged CPU time (hh:mm:ss):	00:01:17
Buffered I/O count:	188
Direct I/O count:	3467
Peak working-set size:	1410
Peak virtual size:	3560

C. Space Requirements: Log and Print Files

MASK.LOG:	27 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 7.21 shows the input file and output file space requirements.

TABLE 7.21. I/O FILE SPACE REQUIREMENTS

File Group	File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	PF118	4	286	
	PF145	5	26574	72 h
	PTE3	Nonstd	14571	72 h
			41431	
Output	PF195	4	52	
			52	

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

MASK.LNK follows:

```
$ ! ACCOUNTING CODE
$ !
$ SET VERIFY
$ !
$ ! Run stream is in file USER2$DISK:[PPROCS.ROMNET]MASK.LNK
$ !
$ ON WARNING THEN EXIT
$ !
$ !  MASK-specific routines:
$ !
$ ASSIGN/USER_MODE USER2$DISK:[PPROCS.ROMNET]MASK.OBJ              A1
$ ASSIGN/USER_MODE USER2$DISK:[PPROCS.ROMNET]RDPTE3.OBJ            A2
$ !
$ !  Utility and I/O routines:
$ !
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]ADATE.OBJ             A3
```

```
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]CELL0.OBJ                A4
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]CELL1.OBJ                A5
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]DATHR6.OBJ               A6
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]FLSMRY.OBJ               A7
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]FSTAT1.OBJ               A8
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]IBLKR.OBJ                A9
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]IOCL.OBJ                 A10
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JFILE1.OBJ               A11
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JHOUR1.OBJ               A12
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JFILE2.OBJ               A13
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]JUNIT.OBJ                A14
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]NEXTHR.OBJ               A15
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]OPDIR2.OBJ               A16
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]OPEN2.OBJ                A17
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]PROGID.OBJ               A18
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]RDDIR1.OBJ               A19
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]RDFHD2.OBJ               A20
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]RDREC.OBJ                A21
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]RDFT4.OBJ                A22
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]RDFT5.OBJ                A23
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]WRREC.OBJ                A24
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]WRFT4.OBJ                A25
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]RDFHD1.OBJ               A26
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]LEN1.OBJ                 A27
$ ASSIGN/USER_MODE USER2$DISK:[ROMNET.UTILIO]DATEHR.OBJ               A28
$ !
$ LINK -
  /EXECUTABLE = USER2$DISK:[PPROCS.ROMNET]MASK.EXE -
  /MAP = MET1:[S01S]MASK.MAP -
  A1,  A2,  A3,  A4,  A5,  A6,  A7,  A8,  A9, A10, -
 A11, A12, A13, A14, A15, A16, A17, A18, A19, A20, -
 A21, A22, A23, A24, A25, A26, A27, A28
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

MASK.COM follows:

```
$ ! ACCOUNTING CODE
$ SET VERIFY
$ ! RUN STREAM IS IN FILE USER2$DISK:[FOREMAN.FV1050A]MASK.COM
$ ! Runs the inversion-mask creation [PPROCS.ROMNET]MASK.EXE
$ ASSIGN MLAB: SYS$PRINT
$ ! ASSIGN INPUT FILES
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                 PFMFDIR
$ ASSIGN MET1:[ROMNET.D000000]PF118A.DAT          PF118
$ ASSIGN MET57:[ROMNET1.D80194]PF145A.DAT         PF145
$ ASSIGN MET57:[ROMNET1.D80194]PTE3A.DAT          PTE3
$ ! ASSIGN OUTPUT FILES
$ ASSIGN MET1:[ROMNET.D000000]PF195A.DAT          PF195
$ SHOW PROCESS / ACCOUNTING
$ !
$ RUN USER2$DISK:[PPROCS.ROMNET]MASK.EXE
$ ! ASSIGN CONTROL CARD VALUE
100
$ !
$ SHOW PROCESS / ACCOUNTING
```

(end of run stream)

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

A. Main Program

MASK

B. Subroutines

None

C. Functions

RDPTE3

D. Block Data Files

None

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

A. Subroutines

DISK$VAXSET:[ROMLIB.UTILIO]

ADATE	IBLKR	RDFHD1
CELL0	JFILE1	RDFHD2
CELL1	NEXTHR	RDFT4
DATEHR	OPDIR2	RDFT5
DATHR6	OPEN2	RDREC
FLSMRY	PROGID	WRFT4
FSTAT1	RDDIR1	WRREC

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

IOCL
JFILE2
JHOUR1
JUNIT
LEN1

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

PARMS.EXT

7.5 PREPROCESSOR TEMPROC (METEOROLOGY)

I. PREPROCESSOR FUNCTION

This preprocessor computes the gridded daily surface maximum, minimum, and mean temperatures that are based on the hourly surface meteorology data, and determines the gridded daily temperature ranges.

II. I/O COMPONENTS

A. Input Files

1. Standard Input Files

None

2. Nonstandard Input Files

a. SURMET1

SURMET1 contains hourly surface meteorology observations from the National Climatic Data Center and the Canadian Climate Centre raw data tapes. SURMET1, generated by preprocessor SFC_VRFY, is read in two stages. The first stage reads header information. READ and FORMAT statements for SURMET1 are called by TEMPROC from RDSRF2 in the UTILIO library. Table 7.22 shows the SURMET1 Stage 1 parameters.

TABLE 7.22. 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 are described in Table 7.23:

TABLE 7.23. 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	SLP_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	$STNPR_n$	mb	Real*4	Station pressure

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

b. BUOYDAT

BUOYDAT contains hourly buoy data, and is generated by prepreprocessor 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 TEMPROC from RDSRF2 in the UTILIO library. Table 7.24 shows the BUOYDAT Stage 1 parameters.

TABLE 7.24. 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	NSTAT		Integer*4	No. of reporting stations at *HH*

The Stage 2 parameters are described in Table 7.25:

TABLE 7.25. 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
12	$SLP_{n,h}$	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. Output Files

1. Standard Output Files

TEMPROC generates one standard file, PF192, that contains the gridded daily surface mean temperatures and the daily surface temperature range (T_{max} - T_{min}). It is used by P26G, and Table 7.26 shows its parameters.

TABLE 7.26. PF192 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1	$AVGT_{i,j,DDD}$	°F	Real*4	Daily average surface air temperature
2	$RANT_{i,j,DDD}$	°F	Real*4	Daily surface air temperature range

Note: i = column number, j = row number, DDD = Julian day

2. Nonstandard Output Files

None

III. CONTROL CARDS

Three control cards are used to input control data in the format shown below. Table 7.27 defines the control card variables.

NF
INTFLAG
JDATHR,EDATHR

TABLE 7.27. CONTROL CARD VARIABLES

Variable Name	Unit	Description
NF		No. of monthly surface meteorology files to be used
INTFLAG		Use of WBAN-format station identifiers: .TRUE. - WBAN format used .FALSE. - WBAN format not used
JDATHR	EST	Julian scenario start date and hour as *YYDDDHH*
EDATHR	EST	Julian scenario end date and hour as *YYDDDHH*

Example:

2
.TRUE.
8818200, 8821200

IV. RESOURCE SUMMARY FOR A ROMNET REGION APPLICATION

A. Memory Requirements (1 block = 512 bytes)

FORTRAN source files:	3	files	60	blocks
FORTRAN INCLUDE files:	0	files	0	blocks
Object files:	3	files	30	blocks
Executable file:	1	file	81	blocks
	7	files	171	blocks

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

	VAX 8650
Charged CPU time (hh:mm:ss):	00:03:10
Buffered I/O count:	193
Direct I/O count:	3771
Peak working-set size:	792
Peak virtual size:	1124

C. Space Requirements: Log and Print Files

TEMPROC.LOG:	15 blocks
Print Files:	None

D. Space Requirements: Input and Output Files

Table 7.28 shows the input file and output file space requirements. Note that the storages will vary according to the length of each episode, and whether more than one month's data are required as input.

TABLE 7.28. I/O FILE SPACE REQUIREMENTS

File Group	Logical File Name	File Type	Storage (in blocks)	Scenario Data Span
Input	SURMET1	Nonstd	88617	1 month
	SURMET2	Nonstd	31287	1 month
			119904	
Output	PF192	5	3806	1 episode

E. Space Requirements: Tape Files

None

V. LINK STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

TEMPROC.LNK follows:

```
$ ! ACCOUNTING CODE
$ !
$ SET VERIFY
$ !
$ ! Run stream is in file TEMPROC.LNK
$ !
$ ON WARNING THEN EXIT
$ !
$ !
$ !  TEMPROC-specific routines:
$ !
$ ASSIGN/USER_MODE DISK$VAXSET:[PPROCS.ROMNET]TEMPROC.OBJ     A1
$ ASSIGN/USER_MODE DISK$VAXSET:[PPROCS.ROMNET]RDSRF2.OBJ      A2
$ ASSIGN/USER_MODE DISK$VAXSET:[PPROCS.ROMNET]STARTSRF.OBJ    A3
$ !
$ !  utility and I/O routines:
$ !
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]ADATE.OBJ      A4
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]BGRID1.OBJ     A5
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]CELLO.OBJ      A6
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]FLSMRY.OBJ     A7
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]FSTAT1.OBJ     A8
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]GETYN.OBJ      A9
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]GMTLCL.OBJ     A10
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]IBLKR.OBJ      A11
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]IOERR.OBJ      A12
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]IOCL.OBJ       A13
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]JFILE1.OBJ     A14
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]JFILE2.OBJ     A15
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]JHOUR1.OBJ     A16
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]JULIAN.OBJ     A17
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]JUNIT.OBJ      A18
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]OPDIR2.OBJ     A19
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]OPEN2.OBJ      A20
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]PROGID.OBJ     A21
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]RDDIR1.OBJ     A22
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]RDREC.OBJ      A23
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]WRREC.OBJ      A24
$ ASSIGN/USER_MODE DISK$VAXSET:[ROMNET.UTILIO]WRFT5.OBJ      A25
$ !
$ LINK -
  /EXECUTABLE = DISK$VAXSET:[PPROCS.ROMNET]TEMPROC.EXE -
  /MAP = MET1:[SO1S]TEMPROC.MAP -
  A1, -
```

```
A2, -
A3, -
  .
  .
  .
A23, -
A24, -
A25
```

(end of link stream)

VI. RUN STREAM COMMAND FILE FOR A ROMNET REGION APPLICATION

TEMPROC.COM follows:

```
$ ! ACCOUNTING CODE
$ !
$ !  Run stream is in file TEMPROC.COM
$ !
$ !
$ !  assign input files
$ !
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                          PFMFDIR
$ !
$ ASSIGN MET23:[FOREMAN.RAWDATA]SRFC_JUL88.DAT             SURMET1
$ ASSIGN MET25:[FOREMAN.RAWDATA]BUOY_JULAUG88.DAT          SURMET2
$ !
$ !  assign output file
$ !
$ ASSIGN MET1:[ROMNET1.D85182]PF192A.DAT                   PF192
$ !
$ RUN DISK$VAXSET:[PPROCS.ROMNET]TEMPROC.EXE
2
.TRUE.
8818200,8821200
```

(end of run stream)

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

A. Main Program

TEMPROC

B. Subroutines

RDSRF2
STARTSRF

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
BGRID1	IBLKR	PROGID
CELL0	IOERR	RDDIR1
FLSMRY	JFILE1	WRFT5
FSTAT1	OPDIR2	WRREC

B. Functions

DISK$VAXSET:[ROMLIB.UTILIO]

GETYN	JULIAN
IOCL	JUNIT
JFILE2	

IX. INCLUDE FILES

DISK$VAXSET:[ROMLIB.ROMNET]

REGPRM.EXT

DISK$VAXSET:[ROMLIB.PROCES]

PARMS.EXT

7.6 THE PFMFCOPY PROGRAM

PFMFCOPY copies a set of records from one PF/MF file to another PF/MF file of the same type.[7,8] The program is able to change the date and time of the data it copies. The inputs required are the logical names of the files, the source data's starting date and time, the number of records to be copied, and the target date and time.

To operate the program, you must assign the logical name PFMFDIR to the appropriate PF/MF database directory, and logical names PFxxx or MFxxx to the source and target files. You will be automatically prompted for these inputs, and thus the program can easily be run interactively, if you so desire. The successive prompts are as follows:

> Enter logical name for source file >>
> Enter source file starting date & hour >>
> Enter number of records to be copied [1] >>
> Enter logical name for target file >>
> Enter target copy starting date & hour >>

The sample run stream below will operate PFMFCOPY in batch mode.

```
$ ! ACCOUNTING CODE
$ !
$ !  Runstream to use PFMFCOPY2 to copy one record from source file PF113
$ !  at date and hour 88238:11 to target file PF170, for date and hour
$ !  88223:16
$ !
$ !
$ !
$ ! assign input and output files:
$ !
$ ASSIGN MET1:[ROMNET]DIRECT2.DAT                    PFMFDIR
$ !
$ ASSIGN MET74:[ROMNET1.D88235]PF113A.DAT            PF113
$ ASSIGN MET56:[ROMNET1.D88223]PF170A.DAT            PF170
$ !
$ RUN  USER2$DISK:[XCC.WORK]PFMFCOPY2
PF113              !  source file
88238 11           !  source starting date and hour
1                  !  number of records to copy
PF170              !  target file
88223 16           !  date and hour on target file
$ !
$ !
$ !  end of runstream
$ !
```

7. Further information on PF and MF *standard files* can be found in Part 2, Section 1; information on setting up the standard files database is contained in Part 4.

8. For example, type 4 file records can be copied only to another type 4 file.

7.7 SHOWMET (METEOROLOGY QUALITY CONTROL)

SHOWMET creates plots from the surface meteorology files that are output from the meteorology preprocessor. These files consist of hourly land surface observations (SURMET1 logical files), hourly buoy ocean surface observations (BUOYDAT logical files), and monthly averaged buoy observations (SURMETM logical files). The parameters of each of these file types are shown in Table 7.29, Table 7.30, and Table 7.31 respectively. Note that the data from each file are plotted in a different color.

TABLE 7.29. SURMET1 PARAMETERS

Parm No.	Parm Name	Unit	Data Type	Description
1a	$STAID_{n,h}$		Char*4	Station ID non-WBAN format
1b	$STAID_{n,h}$		Integer*4	Station ID WBAN (integer) format
2	$SLAT_{n,h}$	° N	Real*4	Latitude
3	$SLON_{n,h}$	° W	Real*4	Longitude
4	$OPQCOV_{n,h}$	%	Real*4	Opaque sky cover
5	$TOTCOV_{n,h}$		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	$COV1_{n,h}$	%	Real*4	Coverage, lowest cloud layer
7	$BASE1_{n,h}$	m	Real*4	Lowest cloud base
8	$COV2_{n,h}$	%	Real*4	Coverage, second-lowest cloud layer
9	$BASE2_{n,h}$	m	Real*4	Second-lowest cloud base
10	$COV3_{n,h}$	%	Real*4	Coverage, third-lowest cloud layer
11	$BASE3_{n,h}$	m	Real*4	Third-lowest cloud base
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	Temperature
16	$TD_{n,h}$	°C	Real*4	Dew point
17	$P_{n,h}$	mb	Real*4	Station pressure

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

TABLE 7.30. BUOYDAT 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.

TABLE 7.31. SURMETM 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	$TAVG_{n,h}$	°C	Real*4	Monthly average air temperature (AT)
16	$SSTAVG_{n,h}$	°C	Real*4	Monthly average sea surface temperature (SST)
17	$DELAVG_{n,h}$	°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.

Note the following:

- SHOWMET is written in Digital Equipment Corporation's (DEC) VAX™ FORTRAN version 5.0, and requires a VMS™ version 5.0 (or higher) operating system. You can refer to the VAX FORTRAN User Manual (AA-D035E-TE), and the VAX FORTRAN Language Reference Manual (AA-D034E-TE), for information on the special features offered by VAX FORTRAN. These manuals are available from Corporate User Publications ZK1-3/J35, Digital Equipment Corporation, 110 Spit Brook Road, Nashua, NH 03062.
- SHOWMET is written using DEC's implementation of release 4.0 of the Graphics Kernel System (GKS™). You can refer to the DEC GKS Reference Manual, Volume 1 (AA-HW43C-TE) and Volume 2 (AA-HW44C-TE), and the GKS User Manual (AA-HW45C-TE) for more information on the GKS. These manuals are available from SSG Publications ZK1-3/J35, Digital Equipment Corporation, 110 Spit Brook Road, Nashua, NH 03062.

  DEC GKS is a device-independent graphics subroutine package based on the International Standards Organization (ISO) and American National Standards Institute (ANSI) Graphical Kernel System. DEC GKS is a development tool that you can use to create graphics applications without regard to device specifics. It can be used for a wide variety of graphics applications including those in business, science, and engineering. DEC GKS implements level 2c of the GKS standard. Level 2c includes support for all GKS output primitives, Workstation Dependent, Workstation Independent Segment Storage, six different logical input devices, and support for synchronous and asynchronous input. DEC GKS RTO is the run-time only version of DEC GKS. It is functionally equivalent to DEC GKS, except that GKS applications cannot be linked against it; it can be used only for applications which have already been linked against DEC GKS.
- The GKS programs can be run only from a node licensed for DEC GKS. To see whether your node is licensed or not, type the command "SHOW LOGICAL GKS*" at the DCL prompt. The VAX system will respond with (LNM$SYSTEM_TABLE), where LNM is the abbreviation for a logical name table. If the node is not licensed, there will be an empty list (a single blank line) beneath the (LNM$SYSTEM_TABLE) response; if it is licensed, a substantial list follows the response.
- SHOWMET is written in the GKS graphics standard and generates output on devices supported by GKS device drivers. DEC GKS includes support for a number of different physical devices that you can choose by specifying the "workstation type" in the open workstation (GKS$OPEN_WS) subroutine. (A workstation is a logical graphics device that permits an application to be written so that it is device-independent.) You must specify the physical connection to the device through the "connection identifier", i.e., the output file.[9] The supported workstations for DEC GKS are shown in Table 7.32. For more information on the supported devices, we suggest that you consult the DEC GKS Device Specific Reference Manual (AA-MJ32A-TE), available from the address listed above.

9. The output file is defined as SYS$OUTPUT for output to a terminal screen, or <file name> for output to other devices.

TABLE 7.32. GKS-SUPPORTED DEVICES

No.	Constant	Description
0	GKS$K_WSTYPE_DEFAULT	Default workstation type
2	GKS$K_GKSM_OUTPUT	GKS output metafile
3	GKS$K_GKSM_INPUT	GKS input metafile
5	GKS$K_WSTYPE_WISS	Workstation Independent Segment Storage
7	GKS$K_CGM_OUTPUT	CGM output metafile (character and ASCII only)
10	GKS$K_VT_OUTPUT	DIGITAL™ VT125 black and white output only
11	GKS$K_VT125	DIGITAL VT125 with color option
12	GKS$K_VT125BW	DIGITAL VT125 (black and white)
13	GKS$K_VT240	DIGITAL VT240 with color option
14	GKS$K_VT240BW	DIGITAL VT240 (black and white)
15	GKS$K_LCP01	DIGITAL LCP01 printer
15	GKS$K_LCG01	DIGITAL LCG01 color printer
16	GKS$K_VT330	DIGITAL VT330 terminal
17	GKS$K_VT340	DIGITAL VT340 terminal
31	GKS$K_LA34	DIGITAL LA34 with graphics option
31	GKS$K_LA100	DIGITAL LA100
32	GKS$K_LA50	DIGITAL LA50 with 2:1 aspect ratio
34	GKS$K_LA210	DIGITAL LA210
38	GKS$K_LN03PLUS	DIGITAL LN03 PLUS laser printer
41	GKS$K_VSII	DIGITAL VAXstation II (monochrome) (VWS)
41	GKS$K_VSII_GPX	DIGITAL VAXstation II/GPX (color) (VWS)
41	GKS$K_VS2000	DIGITAL VAXstation 2000 (VWS)
41	GKS$K_VS3200	DIGITAL VAXstation 3200 (VWS)
41	GKS$K_VS3500	DIGITAL VAXstation 3500 (VWS)
42	GKS$K_VSI	DIGITAL VAXstation I
51	GKS$K_LVP16A	DIGITAL LVP16 color graphics plotter ($8^1/_2 \times 11$ paper size)
51	GKS$K_HP7475	HP7475® color graphics plotter
52	GKS$K_LVP16B	DIGITAL LVP16 color graphics plotter (11 x 17 paper size)
53	GKS$K_HP7550	HP7550® pen plotter
54	GKS$K_HP7580	HP7580® pen plotter
55	GKS$K_LG_MPS2000	MPS2000™ film recorder
56	GKS$K_HP7585	HP7585® pen plotter
61	GKS$K_POSTSCRIPT	DIGITAL LPS40 and PostScript® graphics handler
70	GKS$K_TEK4014_OUTPUT	Tektronix®--4014 output only
72	GKS$K_TEK4014	Tektronix--4014
80	GKS$K_TEK4107_OUTPUT	Tektronix 4107 output only
82	GKS$K_TEK4107	Tektronix 4107
83	GKS$K_TEK4207_OUTPUT	Tektronix 4207 output only
84	GKS$K_TEK4207	Tektronix 4207
85	GKS$K_TEK4128_OUTPUT	Tektronix 4128 output only
86	GKS$K_TEK4128	Tektronix 4128
87	GKS$K_TEK4129_OUTPUT	Tektronix 4129 output only
87	GKS$K_VS500_OUTPUT	DEC VS500 output only
88	GKS$K_TEK4129	Tektronix 4129
89	GKS$K_VS500	DEC VS500
91	GKS$K_LJ250	DEC LJ250 ink jet printer (90 dpi mode)
92	GKS$K_LJ250_180DPI	DEC LJ250 ink jet printer (180 dpi mode)
210	GKS$K_DECWINDOWS_OUTPUT	DECwindows--output only
211	GKS$K_DECWINDOWS	DECwindows--input/output
212	GKS$K_DECWINDOWS_DRAWABLE	DECwindows--an application window, output only
213	GKS$K_DECWINDOWS_WIDGET	DECwindows--input/output within an application widget

DIGITAL and DEC are trademarks of the Digital Equipment Corporation. HP7475, HP7550, HP7580, and HP7585 are registered trademarks of Hewlett Packard, Inc. MPS2000 is a trademark of Lasergraphics. PostScript is a registered trademark of Adobe Systems, Inc. TEKTRONIX is a registered trademark of TEKTRONIX, Inc.

- SHOWMET displays the partial list of terminal types that is shown in Figure 7.2. Note that if you specify the default workstation type (selection 0) but the logical name is undefined, DEC GKS will use GKS$K_VT240BW (selection 14 in the table below) as the workstation type. However, the programs that use GKS declare the TEKTRONIX 4207 (selection 83) to be the default device.

```
  0:  GKS default            7:  CGM metafile
 11:  VT125 color           13:  VT240 color
 17:  VT340 color           38:  LN03 PLUS
 41:  VAXstation II         42:  VAXstation I
 51:  HP7475 plotter        52:  DEC LVP16 plotter
 53:  HP7550 plotter        54:  HP7580    plotter
 61:  POSTSCRIPT            70:  TEK 4014
 80:  TEK 4107  (640x480)   83:  TEK 4207  (640x480)
 85:  TEK 4128 (1280x1024)  87:  TEK 4129 (1280x1024)
211:  DECWINDOWS
```

Figure 7.2. Terminal display of graphics devices for SHOWMET

Before running SHOWMET, if you want to overlay the geopolitical map on the plot, you must assign the map file. You must always assign the logical names to the surface meteorology data files. A sample run stream that performs these assignments is shown below.

```
$ !  ACCOUNTING CODE
$ !
$ !  Sample run stream to assign files to program SHOWMET
$ !
$ SAVERIFY = F$VERIFY("YES")
$ !
$ !
$ !  Assigns ROMNET region map files to SHOWMET
$ !
$ ASSIGN USER2$DISK:[GRAFIX.DATA]ROMNET18IN.DAT        ROMNET18IN
$ !
$ ! ASSIGN SURFACE METEOROLOGY FILES FOR SHOWMET
$ !
$ ASSIGN MET25:[FOREMAN.RAWDATA]BUOY_MAYAUG88.DAT      BUOYDAT
$ ASSIGN MET25:[FOREMAN.RAWDATA]SRFC_MAY88.DAT         SURMET1A
$ ASSIGN MET25:[FOREMAN.RAWDATA]SRFC_JUN88.DAT         SURMET1B
$ ASSIGN MET25:[FOREMAN.RAWDATA]SRFC_JUL88.DAT         SURMET1C
$ ASSIGN MET25:[FOREMAN.RAWDATA]SRFC_AUG88.DAT         SURMET1D
$ ASSIGN MET1:[ROMNET.RAWDATA]BUOY_AVGALL.DAT          SURMETM
$ !
$ RUN SHOWMET
$ !
$ !  end of run stream
$ !
```

SHOWMET will ask you a sequence of questions associated with the setup for its plots. We show the setup prompts for SHOWMET in Table 7.33; the program's defaults are contained within square brackets, and the informational text that it writes is shown italicized. We indicate user responses in shaded boxes; [<RET>] or no shaded box response implies that you must press RETURN for the default response. We give additional information concerning a prompt immediately below the prompt, with the first line indented. Your responses to prompts are checked for appropriateness by the program. If you enter an inappropriate response, the prompt will be repeated. A station key to help you interpret the program's output is shown in Figure 7.3. Note that since the data are contained in sequential ASCII files, it may take several minutes for a plot to be completed. If you plot to a file, you must make sure that you include a path to a directory in which you have write privileges.

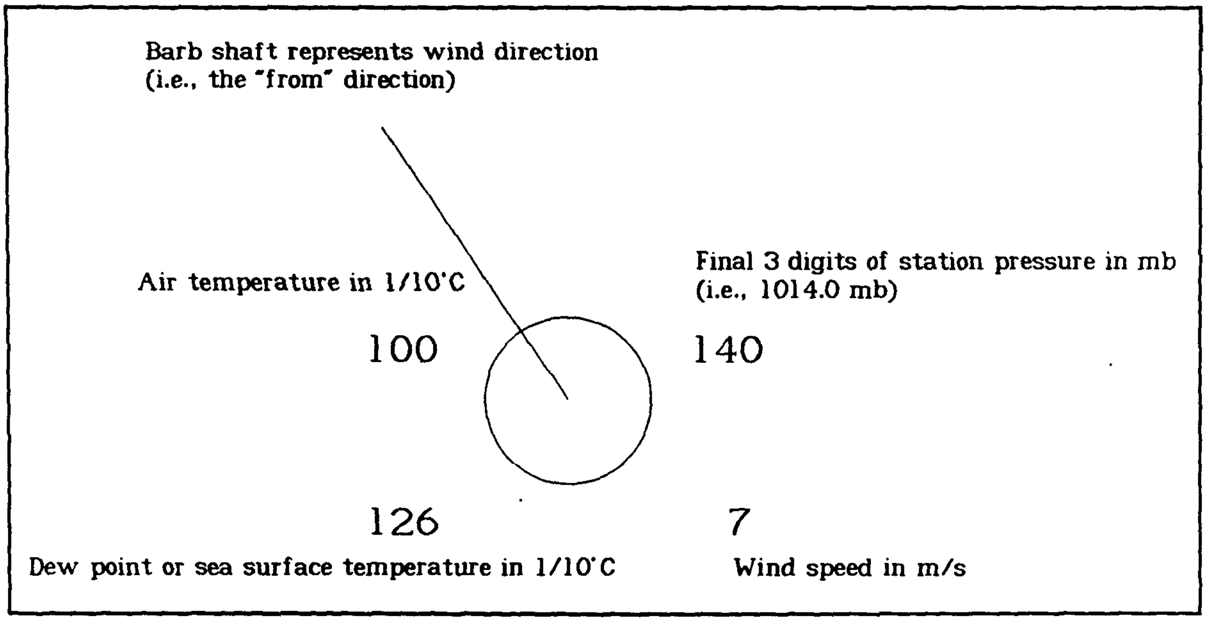

Figure 7.3. SHOWMET station key.

TABLE 7.33. SHOWMET PROMPTS

Prompts and Messages when All Defaults Are Accepted	Prompts and Messages Arising from Non-Default Responses
Enter NUMBER for terminal-type [83] >>>	
Enter REGION name [ROMNET1] >>>	
Enter TIME ZONE for this REGION[5] >>	
Please enter input file name (RET to quit entries): [BUOYDAT] Please enter input file name (RET to quit entries): [SURMET1A] Please enter input file name (RET to quit entries): [SURMET1B] Please enter input file name (RET to quit entries): [SURMET1C] Please enter input file name (RET to quit entries): [SURMET1D] Please enter input file name (RET to quit entries): [SURMETM] Please enter input file name (RET to quit entries): [RETURN] You can assign up to 50 files (including MAP files).	
Do these files use integer IDs ? [Y] >> Integer ID's are the modern WBAN numbers that have been assigned to weather stations.	
SEQUENTIAL FILE "BUOYDAT" OPENED ON UNIT 1 *STATUS: OLD READONLY: YES SHARED: NO FORMATTED: YES* *EQNAME = 2DUA25:[FOREMAN.RAWDATA]BUOY_MAYAUG88.DAT;1* *STARTING DATE & HOUR: 88121:19*	
SEQUENTIAL FILE "SURMET1A" OPENED ON UNIT 2 *STATUS: OLD READONLY: YES SHARED: NO FORMATTED: YES* *EQNAME =* *2DUA25:[FOREMAN.RAWDATA]SRFC_MAY88.DAT;1* *STARTING DATE & HOUR: 88121:19*	
SEQUENTIAL FILE "SURMET1B" OPENED ON UNIT 3 *STATUS: OLD READONLY: YES SHARED: NO FORMATTED: YES* *EQNAME =* *2DUA25:[FOREMAN.RAWDATA]SRFC_JUN88.DAT;1* *STARTING DATE & HOUR: 88152:19*	

continued

Prompts and Messages when All Defaults Are Accepted	Prompts and Messages Arising from Non-Default Responses
SEQUENTIAL FILE "SURMET1C"OPENED ON UNIT 4 STATUS: OLD READONLY: YES SHARED: NO FORMATTED: YES EQNAME = 2DUA25:[FOREMAN.RAWDATA]SRFC_JUL88.DAT;1 STARTING DATE & HOUR: 88182:19	
SEQUENTIAL FILE "SURMET1D" OPENED ON UNIT 7 STATUS: OLD READONLY: YES SHARED: NO FORMATTED: YES EQNAME = 2DUA25:[FOREMA.RAWDATA]SRFC_AUG88.DAT;1 STARTING DATE & HOUR: 88213:19	
SEQUENTIAL FILE "SURMETM" OPENED ON UNIT 8 STATUS: OLD READONLY: YES SHARED: NO FORMATTED: YES EQNAME = 2DUA6:[ROMNET.RAWDATA]BUOY_AVGALL.DAT;7 STARTING DATE & HOUR: 80182:19	
WARNING -- starting years don't match	
Enter new file names and try again (Y/N) [Y] >> N Note that SURMETM contains the 1988 data.	
Enter STEPSIZE (integer HOURS)[1] >>	
Do you want the map ? [Y] >>	
Do you want the grid displayed? [Y] . When you do not subsequently select a window, the plot displays a latitude-longitude grid.	
Do you want a subregion window? [N] Note from Figure 7.4 (a standard weather map) that the data are crowded around major population centers, and cannot be discriminated because of overwriting problems. We recommend appropriate windowing.	Enter first COLUMN for window[1] >> 43 Enter last COLUMN for window[64] >> 50 Enter first ROW for window[1] >> 25 Enter last ROW for window[52] >> 30 See Figure 7.5 for this windowed region. Note that when you request a windowed region and a grid depiction, the grid that is shown is the model grid.

continued

TABLE 7.33 (concluded)

```
******************************************************************************
*                                                                            *
*   1:  Station ID's                                                         *
*   2:  Standard weather map                                                 *
*   3:  Parameter  3 (for SURMET1:  opaque sky cover, %)                     *
*   4:  Parameter  4 (for SURMET1:  total sky cover, coded 0-8)              *
*   5:  Parameter  5 (for SURMET1:  lowest cloud cover, %)                   *
*   6:  Parameter  6 (for SURMET1:  lowest cloud height, m)                  *
*   7:  Parameter  7 (for SURMET1:  2nd lowest cloud cover, %)               *
*   8:  Parameter  8 (for SURMET1:  2nd lowest cloud height, m)              *
*   9:  Parameter  9 (for SURMET1:  3rd lowest cloud cover, %)               *
*  10:  Parameter 10 (for SURMET1:  3rd lowest cloud height, m)              *
*  11:  Parameter 11 (for SURMET1:  sea level pressure, mb)                  *
*  12:  Parameter 12 (for SURMET1:  wind direction, deg)                     *
*  13:  Parameter 13 (for SURMET1:  wind speed, m/s)                         *
*  14:  Parameter 14 (for SURMET1:  temperature, deg C)                      *
*  15:  Parameter 15 (for SURMET1:  dew point, deg C)                        *
*  16:  Parameter 16 (for SURMET1:  station pressure, mb)                    *
*                                                                            *
******************************************************************************
```

What do you want to plot?[2] > >

Enter desired DATE for plot (0 to quit)[80186]> > > 88121

Enter desired HOUR for plot[19] > >

Output file name? [<RET>: screen output, RESET: new device] > >

Do you wish to adjust the terminal colors (Y/N)[N] > >

Just plotted: data for parameter 2, date & hour 88121:19

STEP means to go to the date & hour STEPSIZE hours later
VARIABLE means to select a new parameter for this date & hour
EXAMINE means to select a new parameter, date & hour
CHANGE means to change the setup, from STEPSIZE onward
DEVICE means to select a new device, then reprompt "S step ..."
QUIT means to close down the graphics and stop the program

S step, V variable, E examine, C change, D device, Q quit [S] > >

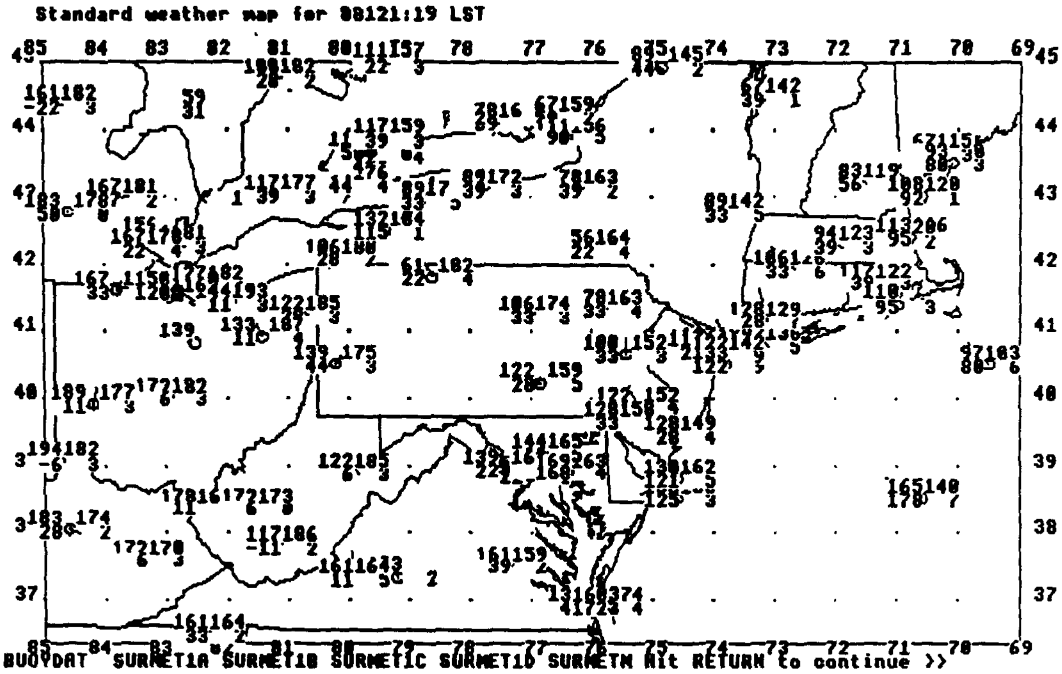

Figure 7.4. Sample ROMNET plot for SHOWMET.

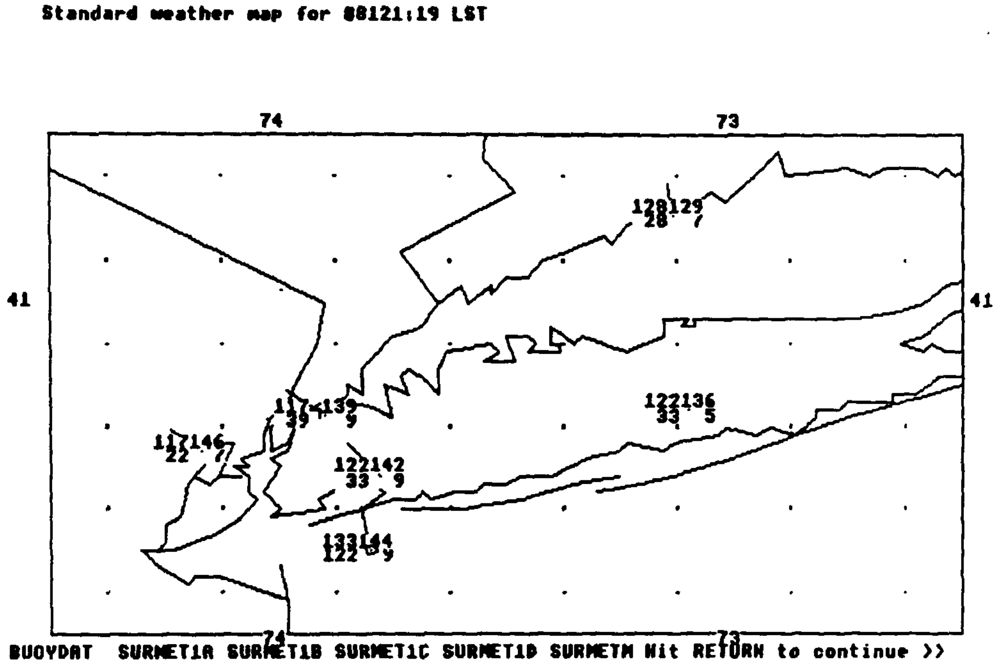

Figure 7.5. Sample ROMNET windowed plot for SHOWMET.

INDEX