

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

954668

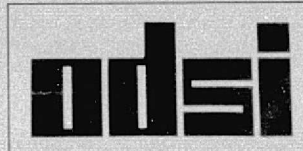
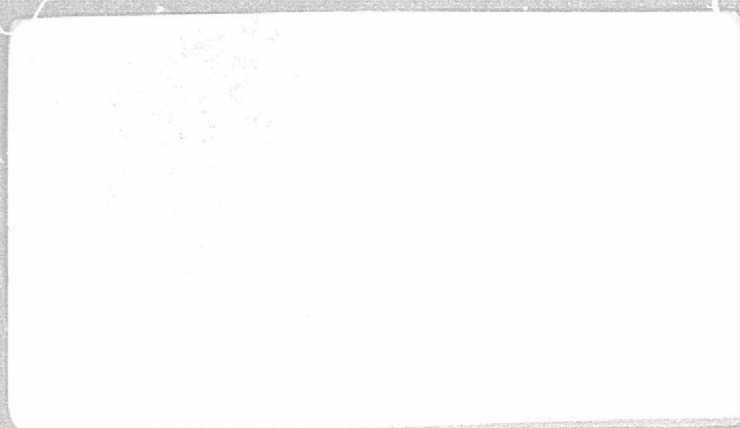
OCEAN DATA SYSTEMS, INC.

(NASA-CR-157329) ATMOSPHERIC MODEL
DEVELOPMENT IN SUPPORT OF SEASAT. VOLUME 3:
PREDICTION MODELS Final Technical Report
(Ocean Data Systems, Inc.) 165 p HC A08/MF
01

N78-28581

Unclas
25957

CSCL 22A G3/43





OCEAN DATA SYSTEMS, INC.

2400 GARDEN ROAD, MONTEREY, CALIFORNIA 93940 • 408/649-1133

Submitted to
JET PROPULSION LABORATORY
Pasadena, California

ATMOSPHERIC MODEL DEVELOPMENT

IN SUPPORT OF SEASAT

VOLUME III - PREDICTION MODELS

Final Technical Report

Prepared under
JPL Contract Number 954668
(Subcontract of NASA Contract Number NAS7-100)

September 30, 1977

Prepared by
Robert E. Wellick
OCEAN DATA SYSTEMS, INC.
Monterey, California

ABSTRACT

As part of the SEASAT program of NASA, a set of four hemispheric, atmospheric prediction models were developed for the Jet Propulsion Laboratory under Contract 954668 (Subcontract under NASA Contract No. NAS7-100). The descriptors applied to these four models, which use a polar stereographic grid in the horizontal and a sigma coordinate in the vertical, are:

- PECHCV - five sigma layers and a 63 x 63 horizontal grid.
- PECHFV - ten sigma layers and a 63 x 63 horizontal grid.
- PEFHCV - five sigma layers and a 187 x 187 horizontal grid.
- PEFHFV - ten sigma layers and a 187 x 187 horizontal grid.

Conservation forms of the difference equations based on the Arakawa technique are integrated using either a fifteen- or four-minute time step on a 381 km. or 127 km. grid (at 60°N) for the 63 x 63 or 187 x 187 models, respectively. Pressure-gradient-force terms are replaced by a single geopotential gradient on local pressure surfaces to reduce inconsistent truncation error (Kurihara modification). Stress is applied at the lowest level. A nonlinear pressure smoother is used along with momentum and temperature diffusion to help control computational noise. The horizontal boundary conditions are insulated slippery walls. Centered time

differencing with time averaging of the pressure-gradient-force term in the momentum equations is used. Robert time filtering of the temperature and moisture solutions is used for computational stability.

The moisture and heat source/sink terms are modeled in a similar manner to those in the early Mintz and Arakawa general circulation model. Terms representing evaporation and large-scale condensation, sensible heat exchange, parameterized cumulus convection and precipitation, and solar and terrestrial radiation are included. Dry convective adjustment precludes hydrostatic instability.

Initialization of the models is based on a pattern conservation technique to obtain objective analyses of the state parameter structure from the surface to 50 mb.

TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
ABSTRACT	ii
LIST OF FIGURES	vii
LIST OF TABLES	ix
I. INTRODUCTION	I-1
II. PHYSICAL-MATHEMATICAL DESCRIPTION OF THE FORECAST MODELS	II-1
A. Basic Model Equations	II-2
B. Grid Structure and Boundary Conditions	II-4
C. Heating and Moisture Source Terms	II-9
1. Solar and Terrestrial Radiation and Sensible Heating	II-10
a. Solar Radiation	II-10
b. Terrestrial Radiation	II-13
c. Sensible Heating and Evapora- tion in the Arakawa-Type Planetary Boundary Layer	II-16
2. Large-Scale Condensation	II-19
3. Convective Adjustment	II-23
a. Parameterized Cumulus Convection	II-24
b. Dry Convective Adjustment	II-30
D. Numerical Techniques	II-32
1. Spacial Finite Differencing	II-32
2. Temporal Finite Differencing	II-37
E. Initialization	II-42
1. Constant Fields	II-42
2. Terrain Pressure	II-44
3. Temperatures	II-45
4. Geopotentials	II-45
5. Winds	II-45
6. Moisture	II-46
III. COMPUTER PROGRAM DESCRIPTION OF THE FORECAST MODELS	III-1
A. Data Structure	III-3
1. Data Storage - Philosophy	III-3
2. Data Storage - Realization	III-8
a. Data Storage - PECHCV and PECHFV	III-11
b. Data Storage - PEFHCV and PEFHFV	III-21

TABLE OF CONTENTS (Continued)

<u>Section</u>	<u>Page</u>
B. Code Structure - PECHCV and PECHFV	III-33
1. Main Overlay - Programs PECHCV and PECHFV	III-34
a. Subroutine BUFERR	III-34
2. Initialization Overlay - Program INITPE	III-39
a. Subroutine SETPHIS.....	III-48
b. Subroutine MOIST	III-48
c. Subroutine WIND	III-49
d. Subroutine TREAD	III-49
e. Subroutine DATAIO	III-49
f. Subroutine RSTART	III-50
g. Subroutine SORTIT	III-50
3. Integration Overlay - Program INTGPE	III-53
a. Subroutine DEFINE	III-53
b. Subroutine LSTEPP	III-59
c. Subroutine PITEND	III-59
d. Subroutine PISM	III-61
e. Subroutine TQTEND	III-61
f. Subroutine LARCON	III-61
g. Subroutine HEATING	III-62
h. Function ALON	III-62
i. Subroutine CONVADJ	III-62
j. Subroutine SETPHI	III-63
k. Subroutine LSTEPV	III-63
l. Subroutine PGFTRM	III-65
m. Subroutine UVTEND	III-65
n. Subroutine DRAG	III-65
o. Subroutine BNDRY	III-66
p. Subroutine DIAGN	III-66
q. Subroutine DIAGNL	III-68
r. Function POW	III-68
s. Subroutine SINCOS	III-68
4. Output Overlay - Program OUTPE	III-70
a. Subroutine SORTOUT	III-70
b. Subroutine PLTWRT	III-77
c. Subroutine FILTR	III-77
d. Subroutine SMOOTH	III-77
e. Subroutine PRT	III-78
f. Subroutine RSTRTD	III-78
C. Code Structure - PEFHCV and PEFHFV	III-79
1. Analysis Data - Program PEPREP	III-80
a. Subroutine TREAD	III-83
b. Subroutine DATAIO	III-83
c. Subroutine REPACK	III-83
d. Subroutine MOIST	III-84
e. Subroutine WIND	III-84
2. Main Overlay - Programs PEFHCV and PEFHFV	III-85
a. Subroutine BUFERR	III-85

TABLE OF CONTENTS (Continued)

<u>Section</u>	<u>Page</u>
3. Initialization Overlay - Program	
INITEX	III-90
a. Program SORT	III-90
b. Program RSTART	III-92
4. Integration Overlay - Program	
INTGPE	III-93
5. Output Overlay - Program OUTPE	III-97
a. Subroutine SORTOUT	III-97
b. Subroutine PLTWRT	III-100
c. Subroutine FILTR	III-100
d. Subroutine SMOOTH	III-101
e. Subroutine PRT	III-101
f. Subroutine RSTRTD	III-101
g. Subroutine AVE	III-102
REFERENCES	R-1

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
II-1	Diagram of Sigma Surfaces and Variables ...	II-5
II-2	Northern Hemisphere Polar Stereographic Grid (63 x 63)	II-7
II-3	Schematic Representation of Three Types of Parameterized Cumulus Convection, Where η is the Entrainment Coefficient and C is the Convective Mass Flux	II-25
III-1	Row or Vertical Slice Data Storage Scheme..	III-6
III-2	Central Memory Storage Buffers Required for Row Storage Scheme Used in Models PECHCV and PECHFV	III-12
III-3	Data Flow for One Sweep of the Data File - Models PECHCV and PECHFV	III-15
III-4	Storage Buffers Required for Row Storage Scheme in Model PEFHCV	III-22
III-5	Storage Buffers Required for Row Storage Scheme in Model PEFHFV	III-24
III-6	Data Flow for One Sweep of the Data File - Model PEFHCV	III-26
III-7	Data Flow for One Sweep of the Data File - Model PEFHFV	III-29
III-8	Overlay Structure of Forecast Models PECHCV and PECHFV	III-35
III-9	Initialization Overlay, Programs PECHCV and PECHFV	III-40
III-10	Integration Overlay, Program INTGPE	III-54
III-11	Computation for One Time Step in LSTEPP ...	III-60
III-12	Computation for One Time Step in LSTEPV ...	III-64
III-13	Output Overlay, Program OUTPE - Models PECHCV and PECHFV	III-71

LIST OF FIGURES (Continued)

<u>Figure</u>		<u>Page</u>
III-14	Structure of Analysis Data Program - PEPREP	III-81
III-15	Overlay Structure of Forecast Models PEFHCV and PEFHFV	III-86
III-16	Initialization Overlay Structure - Models PEFHCV and PEFHFV	III-91
III-17	Output Overlay, Program OUTPE - Models PEFHCV and PEFHFV	III-98

LIST OF TABLES

<u>Table</u>		<u>Page</u>
III-1	Offsets of Variables in Arrays AO and AN for PECHCV	III-17
III-2	Offsets of Variables in Arrays AO and AN for PECHFV	III-19
III-3	Offsets of Variables in Arrays AJM, AJ and AJP for PEFHCV and PEFHFV	III-31
III-4	File Structure of Forecast Models PECHCV and PECHFV	III-36
III-5	Definition of Variables in Common Block /CNTRL/ for PECHCV and PECHFV	III-37
III-6	Definition of Variables in Common Block /INDEX/ for PECHCV and PECHFV	III-38
III-7	Definition of Variables in Common Block /INCON/ for PECHCV and PECHFV	III-41
III-8	Contents of the Input Data File - TAPE1 Created by the Analysis Programs	III-42
III-9	Contents of the Random Access File - TAPE2 for PECHCV and PECHFV, 10 Sigma Level Version of the Forecast Model	III-44
III-10	RESTART Data Tape (TAPE3) Format	III-51
III-11	Definition of Variables in Common Blocks Local to the Programs PECHCV and PECHFV ...	III-55
III-12	Diagnostic Quantities Computed by Subroutine DIAGN	III-67
III-13	Diagnostic Quantities Computed by Subroutine DIAGNL	III-69
III-14	Contents of the Output Data File - TAPE4 per Output Cycle	III-72
III-15	Common Blocks Local to the Output Overlay - Models PECHCV and PECHFV	III-74
III-16	File Structure of Analysis Data Program - PEPREP	III-82

LIST OF TABLES (Continued)

<u>Table</u>		<u>Page</u>
III-17	File Structure of Forecast Models PEFHCV and PEFHFV	III-87
III-18	Definition of Variables in Common Block /CNTRL/ - Models PEFHCV and PEFHFV	III-88
III-19	Definition of Variables in Common Block /INDEX/ - Models PEFHCV and PEFHFV	III-89
III-20	Definition of Variables in Common Blocks Local to the Programs PEFHCV and PEFHFV ...	III-94
III-21	Common Blocks Local to the Output Overlay - Models PEFHCV and PEFHFV	III-99

I. INTRODUCTION

This document describes a set of four primitive-equation atmospheric forecast models for the Northern Hemisphere that were designed and developed under Contract 954668 for the Jet Propulsion Laboratory, (Subcontract under NASA Contract NAS7-100) as part of the SEASAT program of NASA. The models are hemispheric, using a polar stereographic grid in the horizontal and a sigma coordinate in the vertical. The descriptors applied to the four models are:

PECHCV - five sigma layers and a 63 x 63
horizontal grid

PECHFV - ten sigma layers and a 63 x 63
horizontal grid

PEFHCV - five sigma layers and a 187 x 187
horizontal grid

PEFHFV - ten sigma layers and a 187 x 187
horizontal grid

The 63 x 63 and 187 x 187 grids correspond to mesh lengths of 381 km and 127 km at 60°N, respectively.

There are two main sections describing these forecast models. First, Section II, the physical-mathematical description, gives the governing equations and describes the various physical/dynamical processes and their parameterizations that are included. Second, Section III describes the program structure of the various models. In both sections, and particularly in Section II, the

approach has been to describe the five-layer, 63 x 63 model as a baseline and then to describe the necessary modifications to either increase the vertical resolution to ten layers or to increase the horizontal resolution to 187 x 187, or both.

II. PHYSICAL-MATHEMATICAL DESCRIPTION OF THE FORECAST MODELS

This section gives the meteorological and mathematical description of the set of four forecast models; in other words, the governing equations, numerical techniques, algorithms and initialization procedure which constitute the theoretical basis of the models.

The basic model equations, the primitive equations, are given in Section II-A in a form applicable to a polar stereographic map projection with an x,y coordinate system.

Section II-B then describes the grid structure (horizontal and vertical) and boundary conditions for the hemispheric grid. Section II-C describes the heating and moisture source terms which include the solar and terrestrial radiation and sensible heating, large-scale condensation and convective adjustment, and the parameterized cumulus convection and dry convective adjustment. Section II-D describes the numerical procedures used to approximate the solution of the primitive equations.

Finally, Section II-E describes the methods used to create an initial state for a forecast from the data provided by the analysis models.

A. Basic Model Equations

The governing partial differential equations for the forecast models, written in flux form, are similar to sets used by Smagorinsky et al (1965) and Arakawa et al (1969). These equations are listed below for an x-y map projection with map factor m and vertical coordinate, σ .

Momentum equation in the x direction:

$$\begin{aligned} \frac{\partial (\pi u)}{\partial t} = & -m^2 \left[\frac{\partial}{\partial x} \left(\frac{uu\pi}{m} \right) + \frac{\partial}{\partial y} \left(\frac{uv\pi}{m} \right) \right] + \pi \frac{\partial (wu)}{\partial \sigma} \quad [\text{II.1}] \\ & + \pi v f - m \left(\pi \frac{\partial \Phi}{\partial x} + RT \frac{\partial \pi}{\partial x} \right) + F_x + D_u \end{aligned}$$

Momentum equation in the y direction:

$$\begin{aligned} \frac{\partial (\pi v)}{\partial t} = & -m^2 \left[\frac{\partial}{\partial x} \left(\frac{uv\pi}{m} \right) + \frac{\partial}{\partial y} \left(\frac{vv\pi}{m} \right) \right] + \pi \frac{\partial (wv)}{\partial \sigma} \quad [\text{II.2}] \\ & - \pi u f - m \left(\pi \frac{\partial \Phi}{\partial y} + RT \frac{\partial \pi}{\partial y} \right) + F_y + D_v \end{aligned}$$

Thermodynamic energy equation:

$$\begin{aligned} \frac{\partial (\pi T)}{\partial t} = & -m^2 \left[\frac{\partial}{\partial x} \left(\frac{\pi u T}{m} \right) + \frac{\partial}{\partial y} \left(\frac{\pi v T}{m} \right) \right] + \pi \frac{\partial (w T)}{\partial \sigma} \quad [\text{II.3}] \\ & + H\pi + \frac{RT}{c_p \sigma} \{ -w\pi + \sigma \left[\frac{\partial \pi}{\partial t} + m \left(u \frac{\partial \pi}{\partial x} + v \frac{\partial \pi}{\partial y} \right) \right] \} \\ & + D_T \end{aligned}$$

Moisture conservation equation:

$$\frac{\partial (\pi q)}{\partial t} = -m^2 \left[\frac{\partial}{\partial x} \left(\frac{\pi u q}{m} \right) + \frac{\partial}{\partial y} \left(\frac{\pi v q}{m} \right) \right] + \pi \frac{\partial (w q)}{\partial \sigma} + Q \pi \quad [\text{II.4}]$$

Pressure tendency equation:

$$\frac{\partial \pi}{\partial t} = -m^2 \int_0^1 \left[\frac{\partial}{\partial x} \left(\frac{u \pi}{m} \right) + \frac{\partial}{\partial y} \left(\frac{v \pi}{m} \right) \right] d\sigma \quad [\text{II.5}]$$

Hydrostatic equation:

$$\frac{\partial \phi}{\partial \sigma} = - \frac{RT}{\sigma} \quad [\text{II.6}]$$

In the above equations,

- u = x direction wind component
- v = y direction wind component
- π = terrain pressure
- T = temperature
- q = vapor pressure
- w = $-\frac{d\sigma}{dt}$ = vertical velocity
- m = map factor
- f = Coriolis parameter
- F_x, F_y = stress components
- D_u, D_v = momentum diffusion terms
- D_T = temperature diffusion term
- H = diabatic heating term
- Q = moisture source/sink term
- ϕ = gz = geopotential

B. Grid Structure and Boundary Conditions

Integration of the forecast model equations is performed on Phillips (1957) sigma surfaces in which pressure, p , is normalized with the underlying terrain pressure, π ; that is:

$$\sigma = \frac{p}{\pi} \quad [\text{II.7}]$$

Figure II-1 shows the sigma surfaces used in the forecast models.

The horizontal wind components, u and v , temperature, T , and geopotentials, Φ , are carried at the levels where $\sigma = 0.9, 0.7, 0.5, 0.3$ and 0.1 for the five-layer models, and at $\sigma = 0.95, 0.85, 0.75, 0.65, 0.55, 0.45, 0.35, 0.25, 0.15$ and 0.05 for the ten-layer models. The moisture variable, q , is carried at the lower three and lower six of these sigma surfaces for the five- and ten-layer models, respectively. The vertical velocity, w , is calculated diagnostically from the continuity equation for the layer interfaces ($\sigma = 0.8, 0.6, 0.4$, and 0.2 for the five-layer models, and $\sigma = 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2$ and 0.1 for the ten-layer models), and is assumed to vanish at $\sigma = 1.0$ and $\sigma = 0$. All variables are carried at each point (i,j) of the horizontal grid.

The hemisphere is mapped onto a polar stereographic projection true at 60° north for which the map factor is given by:

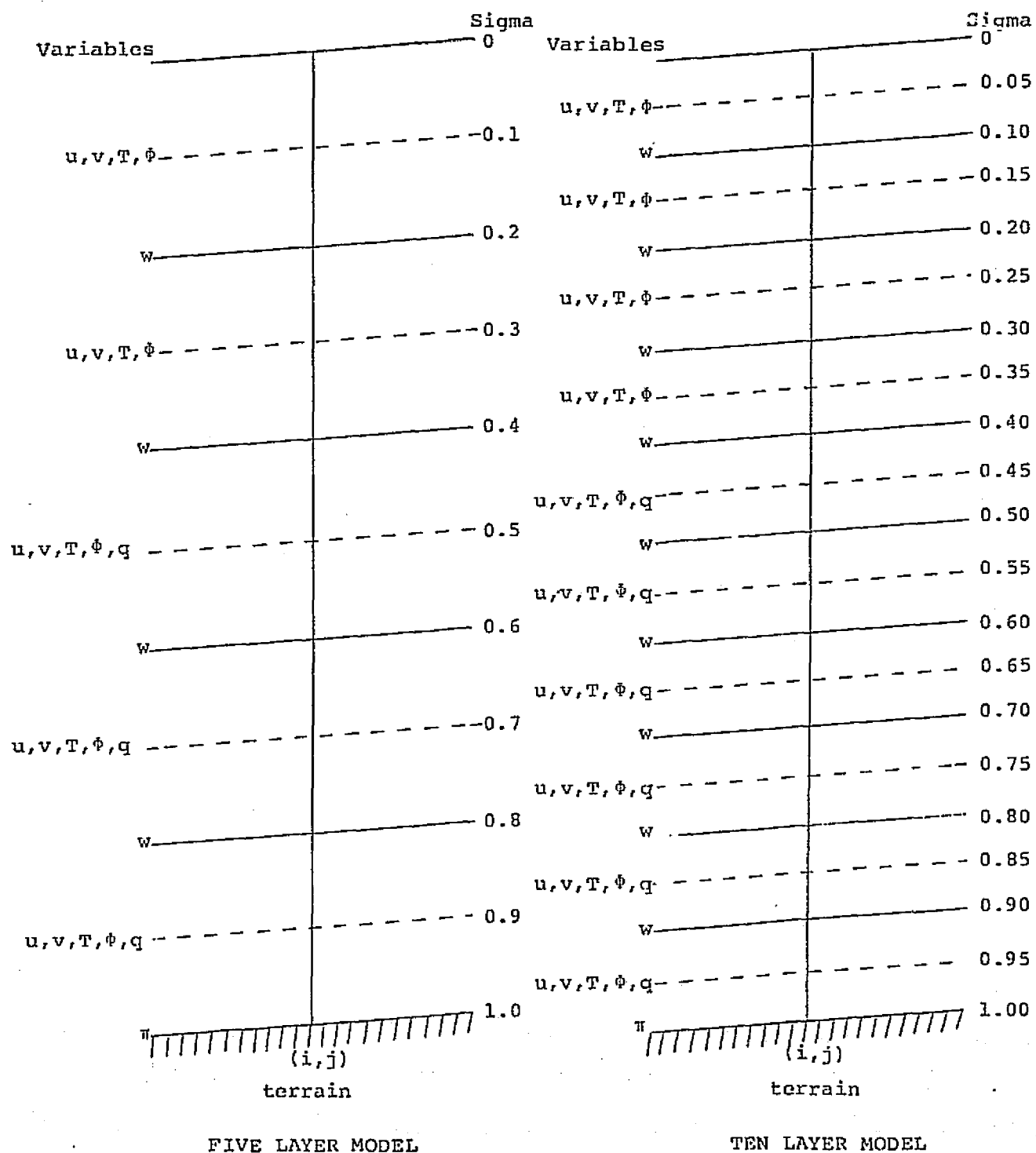


FIGURE II-1: DIAGRAM OF SIGMA SURFACES AND VARIABLES

$$m = \frac{1 + \sin 60^\circ}{1 + \sin \phi} \quad [\text{II.8}]$$

where ϕ is the north latitude. The square grid is centered about the North Pole (and extends approximately to the equator, as shown in Figure II-2). A 63 by 63 or 187 by 187 horizontal grid system is used which gives a grid distance of $\Delta x = \Delta y = 381$ km and 127 km at 60°N , respectively.

The horizontal boundary conditions are similar to those used in the six-layer NMC forecast model described by Shuman et al (1968). The model is bounded by rigid, impermeable vertical walls placed on the grid on the next to outermost row of grid points, as shown in Figure II-2. The boundary is both insulated and slippery; that is, no heat or momentum exchange is permitted across the wall, but a parallel flow is permitted along the wall. This means that along the lower boundary row where y is fixed, the boundary conditions can be written as:

$$\pi_1 = \pi_3; \quad (\pi u)_1 = (\pi u)_3; \quad (\pi v)_1 = -(\pi v)_3 \quad [\text{II.9}]$$

$$T_1 = T_3; \quad (\pi q)_1 = (\pi q)_3; \quad \phi_1 = \phi_3$$

where the subscripts refer to the first and third rows (row 2 being the boundary).

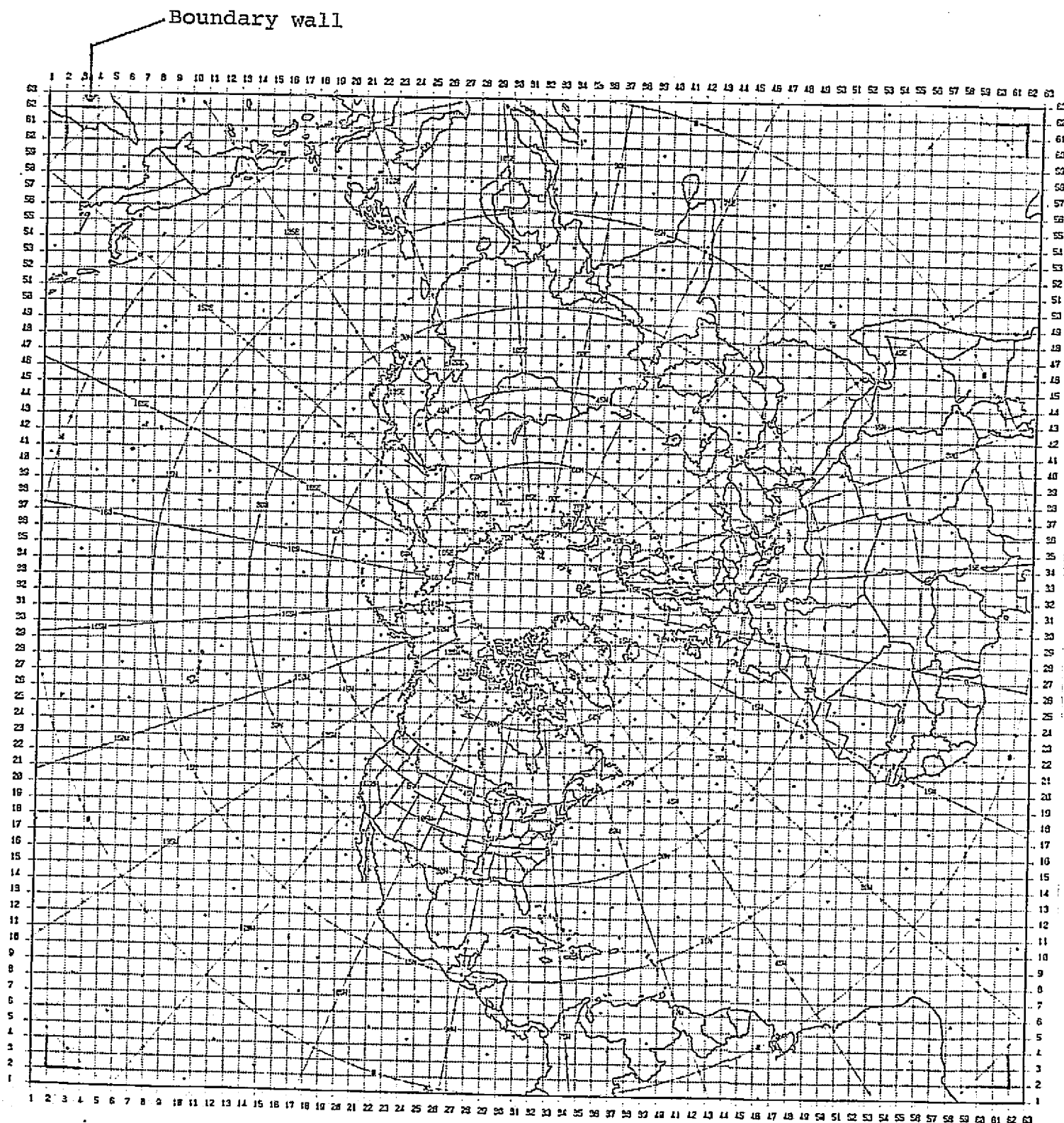


FIGURE II-2: NORTHERN HEMISPHERE POLAR STEREOGRAPHIC GRID (63 x 63).

ORIGINAL PAGE IS
OF POOR QUALITY

Since the polar stereographic map projection becomes quite distorted in the corner areas of the grid and since these regions are not of meteorological significance for the forecast, the computed tendencies are truncated near the edge of the grid. From 20°N to the Pole, full values of the computed tendencies are used. Below 5°N, the tendencies are set to zero. In the region between 5°N and 20°N the amount of the computed tendencies used varies according to the relation $[(\sin\phi - \sin 5^\circ)/(\sin 20^\circ - \sin 5^\circ)]^2$. This procedure results in a persistence forecast in the outermost areas of the grid and increases the stability of the computation near the edge of the grid.

C. Heating and Moisture Source Terms

The diabatic heating rates in the forecast models are calculated in a manner similar to that of Mintz and Arakawa, as outlined by Langlois and Kwok (1969). The equation for the total heating can be written as

$$H = H_{SW} + H_{LW} + H_T + H_C \quad [II.10]$$

where H_{SW} is the heating due to absorption of shortwave (solar) radiation; H_{LW} is the heating due to longwave (terrestrial) radiation; H_T is the contribution of sensible heat from the surface layer ($\sigma = .9$ to $\sigma = 1.0$ for the five-layer models and $\sigma = 0.95$ to $\sigma = 1.0$ for the ten-layer models, a constant flux boundary layer); and H_C is the release of latent heat resulting from phase changes of water substance, usually the latent heat of condensation.

Section 1 describes the computation of the heating rates due to shortwave radiation, H_{SW} , longwave radiation, H_{LW} and sensible heat, H_T . Section 2 describes the large-scale condensation process which contributes to H_C . Section 3 describes the moist and dry convective adjustment processes which also contribute to H_C .

1. Solar and Terrestrial Radiation and Sensible Heating

Diagnostic calculations of the heating; i.e., the temperature tendency, $\Delta T/\Delta t$, represented by H_{SW} , H_{LW} and H_T , are computed every hour and saved for use over the next one-hour forecast period. The temperature tendencies are of the form

$$\frac{\Delta T}{\Delta t} = \frac{g}{C_p} \frac{\Delta F}{\Delta p} \quad [\text{II.11}]$$

where g is the acceleration of gravity; C_p is the specific heat of air at constant pressure and $\frac{\Delta F}{\Delta p}$ is the flux divergence. Flux computations depend on the vertical distribution of temperature, moisture and pressure above each surface grid point; i.e., only vertical fluxes are computed.

The computations of H_{SW} , H_{LW} and H_T are described in Subsections 2, 3 and 4, respectively.

a. Solar Radiation

The incoming solar radiation at the top of the model atmosphere is given by

$$S = S_0 \cos \mu \quad [\text{II.12}]$$

where S_0 is the solar constant (taken to be $1.92 \text{ cal/cm}^2 \text{ min.}$) and μ is the zenith angle which is a function of time of day,

latitude and season.

The zenith angle is given by

$$\cos \mu = \sin \phi \sin \delta + \cos \phi \cos \delta \cosh \quad [\text{II.13}]$$

where ϕ = latitude (0° at Equator, 90° at North Pole)

h = hour angle

δ = declination angle of sun

and

$$\begin{aligned} \sin \delta = & 0.39785 \sin [4.88578 + 0.0172 * \text{DAY}] \\ & + 0.03342 \sin (0.0172 * \text{DAY}) - 0.001388 \cos (0.0172 * \text{DAY}) \\ & + 0.000348 \sin (0.0344 * \text{DAY}) \\ & - 0.000028 \cos (0.0344 * \text{DAY}) \end{aligned}$$

where DAY is the number of days measured from the first of the year.

In order to compute the depletion of the solar beam, S is initially partitioned into two segments. The radiation with wavelength shorter than .9 microns (65.1% of the energy) is subject to scattering by the atmosphere, but not to absorption. Wavelengths greater than .9 microns are subject only to absorption. The calculations that are outlined below follow the scheme constructed by Joseph (1966).

In order to compute the radiative fluxes, a simple form of middle cloud is assumed to exist between $\sigma = 0.4$ and $\sigma = 0.8$. An empirical formula for middle clouds:

$$CL = 1.3 \left(\frac{e}{e_s} \right)_C - \left[0.46 + 0.6 \left(\frac{P_C - C_1}{C_2} \right) \right] \left[1 - \left(\frac{e}{e_s} \right)_C \right] \quad [\text{II.14}]$$

gives the amount of cloudiness, CL, in terms of the relative humidity and pressure at $\sigma = C$. C is taken to be 0.7 for the five-layer models and 0.65 for the ten-layer models. $C_1 = 400$ and $C_2 = 300$ for the five-layer models, and $C_1 = 300$ and $C_2 = 350$ for the ten-layer models. Limits of 1.0 and 0.0 are set on the range of values given by equation [II.14].

The absorption of incoming solar radiation for the upper layer ($\sigma = 0$ to $\sigma = 0.6$) is given by:

$$A_2 = 0.349S[(1-CL)F_1 + CL(1-\alpha_c)F_2] \quad [II.15]$$

and for the lower layer ($\sigma = 0.6$ to $\sigma = 1.0$) by:

$$A_1 = 0.349S[(1-CL)F_3 + CL(1-\alpha_c)F_4] \quad [II.16]$$

where

$$F_1 = 0.271 \left[\frac{pw(U)}{\cos \mu} \right]^{0.303} \quad [II.17]$$

$$F_2 = 0.271 [2.91 pw(U)]^{0.303} \quad [II.18]$$

$$F_3 = 0.271 \left[\frac{pw(T)}{\cos \mu} \right]^{0.303} - F_1 \quad [II.19]$$

$$F_4 = 0.271 [2.91 pw(T)]^{0.303} - F_2 \quad [II.20]$$

The symbol pw is the precipitable water given by:

$$pw = \frac{1}{g} \int_{p_1}^{p_2} 0.622 \frac{e}{1000} dp \quad [II.21]$$

and U refers to the amount of precipitable water above $\sigma = 0.6$

and T refers to the total amount of precipitable water. These absorption functions are from Manabe and Moller (1961). The clouds' albedo, α_c , is set at 0.5. Finally, the amount of solar radiation that reaches the surface, S_g , is needed for the calculation of sensible heating over the land areas. Following Mintz and Arakawa:

$$S_g = (1-\alpha_g) \left((1-CL) \left[0.349S - A_2 - A_1 + \frac{0.651S(1-\alpha_s)}{1-\alpha_s\alpha_g} \right] \right. \\ \left. + CL \left\{ \frac{0.349S[(1-\alpha_s) - (1-\alpha_c)(F_1 + F_2)]}{1-\alpha_c\alpha_g} + \frac{1-\alpha_{sc}}{1-\alpha_{sc}\alpha_g} 0.651S \right\} \right) \quad [II.22]$$

where α_g , the ground surface albedo, is given as a function of $\cos\mu$ over the sea and as a function of snow cover over the land; α_{sc} is the albedo of the cloudy sky and α_s is the albedo of the clear sky and is given by:

$$\alpha_s = 0.085 - 0.10727 \ln\left[\sin\mu \left(\frac{P_s}{1000}\right)\right] \quad [II.23]$$

b. Terrestrial Radiation

The infrared fluxes at $\sigma = 1.0$, 0.6 and 0.2 are computed in the manner suggested by Mintz and Arakawa and are given by:

$$F_{1.0} = F_{1.0}^* + St(T_g^4 - T_s^4) \quad [II.24]$$

$$F_{0.6} = F_{0.6}^* + \frac{0.8(1-CL)(F_{1.0} - F_{1.0}^*)}{1 + 1.75[pw(L)]^{0.416}} \quad [II.25]$$

and

$$F_{0.2} = F_{0.2}^* + \frac{0.8(1-CL)(F_{1.0} - F_{1.0}^*)}{1 + 1.75[pw(L) + 1.2 pw(U)]^{0.416}} \quad [II.26]$$

where St is the Stefan-Boltzmann constant and U and L refer to the amount of precipitable water above and below $\sigma = 0.6$, respectively.

$F_{1.0}^*$, $F_{0.6}^*$ and $F_{0.2}^*$ are the fluxes computed by assuming the surface temperature is given by a downward extrapolated temperature $T_s = \frac{1}{2} (3T_{0.9} - T_{0.7})$ for the five-layer models and $T_s = \frac{1}{2} (3T_{0.95} - T_{0.85})$ for the ten-layer models. The factor 0.8 is an empirical correction to account for the effect of CO_2 .

In equation [II.24], T_g is given over the oceans by the sea surface analysis, and over the land areas it is given by the solution of a heat balance equation explained in Section II-C-1-c on sensible heat transfer and evaporation.

To compute the approximate fluxes, F^* , the method developed by Danard (1969) is used with an emissivity function for water vapor from Kuhn (1963). Thus,

$$F_{1.0}^* = St \left((1-CL) \left\{ -T_s^4 + T_{0.8}^4 \phi(X1) + T_{0.4}^4 [\phi(X2) - \phi(X1)] \right\} + CL[T_{0.8}^4 - T_1^4 - T_{0.8}^4] \phi(X4) \right) \quad [II.27]$$

$$F_{0.6}^* = St(1-CL) [-T_s^4 + (T_s^4 - T_{0.8}^4) \phi(X1) + T_{0.4}^4 \phi(X3)] \quad [II.28]$$

and

$$F_{0.2}^* = St \left((1-CL) \left\{ -T_S^4 + (T_S^4 - T_{0.8}^4) [\Phi(X2) - \Phi(X3)] \right. \right. \\ \left. \left. + (T_S^4 - T_{0.4}^4) \Phi(X3) \right\} - CL T_{0.4}^4 \right) \quad [II.29]$$

where

$$\Phi(X) = 0.0105X^2 + 0.1675X + 0.542 \quad [II.30]$$

$$X1 = \log_{10} [pw(L)] \quad [II.31]$$

$$X2 = \log_{10} [pw(L) + 1.2pw(U)] \quad [II.32]$$

$$X3 = \log_{10} [1.2pw(U)] \quad [II.33]$$

and

$$X4 = \log_{10} [pw(1)] \quad [II.34]$$

where 1 denotes the lowest level.

For these calculations, pw as given by equation [II.21] is modified to:

$$pw = \frac{1}{g} \int_{P_1}^{P_2} 0.622 \frac{e}{P} \left(\frac{P}{1000} \right)^{0.85} \left(\frac{273}{T} \right)^{0.5} dP \quad [II.35]$$

which is the pressure- and temperature-corrected mass of water vapor in the atmospheric column.

Finally, the fluxes are converted to heating rates by

$$H_{LW2} = - (F_{1.0} - F_{0.6}) \left(\frac{3600}{.4 \cdot 1000} \right) \frac{g}{P_s C_p} (^\circ K/hr) \quad [II.36]$$

which is used below the $\sigma = 0.6$ level.

$$H_{LW1} = - (F_{0.6} - F_{0.2}) \left(\frac{3600}{.4 \cdot 1000} \right) \frac{g}{P_s C_p} (^\circ K/hr) \quad [II.37]$$

which is used above the $\sigma = 0.6$ level.

c. Sensible Heating and Evaporation in the Arakawa-Type Planetary Boundary Layer

Sensible heating is computed as a function of the difference of the surface temperature, T_g , and the temperature of the air near the surface, T_x . Similarly, evaporation is computed as a function of the saturation specific humidity at the surface, Q_s , and the specific humidity near the surface, Q_x , except over land, where a Bowen ratio is computed to determine the flux of moisture. Difficulty arises in the forecast models since near-surface temperature, wind and moisture are not predicted. To surmount this difficulty, the technique of Mintz and Arakawa is used.

The air near the surface; i.e., $\sigma = 1.0$ to $\sigma = 0.9$, in the five-layer models and $\sigma = 1.0$ to $\sigma = 0.95$ in the ten-layer models, is assumed to be a very thin boundary layer which does not absorb or store a significant amount of heat or moisture. The fluxes of sensible heat and moisture are, therefore, the same through the top and the bottom of this layer. Generally, for heat transfer, an equation of the form:

$$\Gamma = -K_H \frac{\Delta\theta}{\Delta z} \quad [\text{II.38}]$$

is used where θ is the potential temperature. Equation [II.38] is modified by Arakawa to

$$\Gamma = \rho_{1.0} C_p \frac{K^*}{\left[1 + \frac{a^*(\theta_{BL} - \theta_s)}{\Delta z}\right]} \left[\gamma_c - \frac{(\theta_{BL} - \theta_x)}{\Delta z}\right] \quad [\text{II.39}]$$

where $\Delta Z = Z_{BL} - Z_{1.0}$. In equation [II.39], the expression for K_H is allowed to vary with the stability of the boundary layer. K^* , a^* and γ_c are empirical constants. The constant γ_c allows for a somewhat more realistic variation of Γ between day and night. The subscript s refers to the value of a parameter extrapolated to the surface from the two lowest sigma levels.

The sensible heat flux through the bottom of the boundary layer is given by

$$H_T = \rho_{1.0} C_p C_D V_s (T_g - T_x) \quad [\text{II.40}]$$

where, for this computation,

$$V_s = 0.8 [U_{BL}^2 + V_{BL}^2]^{1/2} + Gu \quad [\text{II.41}]$$

where Gu , the gustiness factor, is 2.2 m/sec. A drag coefficient $C_D = 0.002$ is used.

Over the oceans, T_x can easily be obtained from equations [II.39] and [II.40], since the sea surface temperature T_g , is specified. Knowing T_x , a heating rate due to sensible heating for the lowest sigma layer can be determined.

Over the land areas, T_g must be determined from a surface energy balance equation:

$$(1+r)H_T - S_g + F_{1.0} = 0 \quad [\text{II.42}]$$

in which only the heat storage has been neglected. Data from Budyko (given in Sellers 1965) has been used to determine r , a Bowen ratio, as:

$$r = 9.6 (\sin\phi)^2 - 7.93 (\sin\phi) + 2.0 \quad [\text{II.43}]$$

in equation [II.42], the amount of solar radiation reaching the ground, S_g , is given by equation [II.22], and the infrared flux from the surface, $F_{1.0}$, is given by equation [II.24]. Equation [II.42] is modified over ice-covered ocean to:

$$(1+r)H_T - S_g + F_{1.0} = B(T_w - T_g) \quad [\text{II.44}]$$

where $B = 697.83 \text{ erg/cm}^2 \text{ sec } ^\circ\text{K}$ is the ice conduction coefficient for three-meter-thick ice and $T_w = 271.2^\circ\text{K}$.

Evaporation over the ice-free oceans is determined in a similar manner to the heat flux. The equation resulting is:

$$E = \frac{\rho_{1.0} C_p V_s}{1 + \frac{\bar{K} C_p V_s}{\Delta z}} (q_s - q_x) \quad [\text{II.45}]$$

where

$$\bar{K} = \frac{K^*}{1 + \frac{a^* (\theta_{BL} - \theta_s)}{\Delta z}} \quad [\text{II.46}]$$

$$q_s = \frac{0.622 e_s}{p_s - e_s} \quad [\text{II.47}]$$

and

$$e_s = \exp(Ae - \frac{Be}{T_s}) \quad [\text{II.48}]$$

$$Ae = 21.656 \text{ and } Be = 5418.$$

The details of the derivation of equation [II.45] are given by Langlois and Kwok (1969).

Using the Bowen ratio from equation [II.43] allows the determination of the evaporation for land areas and ice-covered ocean as:

$$E = H/r \quad [\text{II.49}]$$

The expression for r given by equation [II.43] has the effect of making the land surface at low latitudes a greater source of moisture than those at higher latitudes.

2. Large-Scale Condensation

The large-scale condensation mechanism operates under the premise that synoptic scale motions do not contain supersaturated states. If a state of supersaturation is discovered (which may be due to excessive evaporation from the underlying surface or due to the effects of horizontal or vertical advection of moisture), then sufficient moisture is removed (as precipitation with the latent heat of condensation) to achieve a saturated state at a warmer temperature. The technique suggested by Mintz and Arakawa as outlined by Langlois and Kwok (1969) is followed in the forecast models.

A solution to the equation

$$F(T) = C_p(T-T_o) - L[q_o - q_s(T)] = 0 \quad [\text{II.50}]$$

is sought; where q_o is the supersaturation mixing ratio associated with the temperature T_o and q_s is the saturation mixing ratio associated with the new temperature T . L is the latent heat of condensation which is taken to be 600 cal/gm.

Equation [II.50] is a non-linear equation. A second-order Newton-Raphson approximation to the solution is used, which is given by:

$$T = T_o - \frac{F(T_o)}{F'(T_o)} \left[1 + \frac{F''(T_o)}{2} \frac{F(T_o)}{F'(T_o)^2} \right] \quad [\text{II.51}]$$

Expressions for the derivatives of $F(T_o)$ must now be generated. $F'(T)$ may be written as:

$$F'(T) = 1 + \frac{L}{C_p} \frac{d}{dT} q_s(T) \quad [\text{II.52}]$$

where

$$q_s(T) = \frac{e e_s}{p - e_s}$$

and e_s is given by equation [II.48]. Differentiating $q_s(T)$ and ignoring dp/dT gives:

$$F'(T_o) = 1 + \frac{L}{C_p} \frac{e p}{(p - e_s)^2} \frac{B e}{T_o^2} e_s(T_o) \quad [\text{II.53}]$$

which may be further simplified to give

$$F'(T_o) = 1 + \frac{L}{C_p} Y \quad [\text{II.54}]$$

where:

$$Y = \frac{\text{Beq}_s(T_o)}{T_o^2} \left[1 + \frac{1}{\epsilon} q_s(T_o) \right] \quad [\text{II.55}]$$

The second derivative $F''(T_o)$ is given by:

$$F''(T_o) = \frac{L}{C_p} \frac{dY}{dT} \quad [\text{II.56}]$$

which, after considerable algebraic manipulation, can be written as:

$$F''(T_o) = \frac{2LY}{C_p T_o} \left[\frac{\text{Be}}{T_o} \left(\frac{1}{2} + \frac{1}{\epsilon} q_s \right) - 1 \right] \quad [\text{II.57}]$$

Using equation [II.51], the change in temperature may be obtained and from this, the change in mixing ratio can be computed by:

$$\Delta q = \frac{C_p}{L} \Delta T \quad [\text{II.58}]$$

The air is warmed by the amount ΔT and moisture is removed by the amount Δq from the supersaturated layer.

Since moisture is forecast at the three lower levels in the five-layer models and at the lower six levels in the ten-layer models, a precipitation algorithm is required.

This algorithm proceeds from the lowest level to the highest level, and is outlined below.

- Examine the layer for supersaturation. If we are at the lowest sigma level and the layer is supersaturated, then precipitation occurs; otherwise, proceed to the next step.

- Examine the next lower layer for saturation. If the layer is saturated, then we allow the precipitation to pass through the layer to the surface where it is added to the total at that particular grid point; otherwise, proceed to the next step.

- If the next lower layer was initially unsaturated and the addition of the precipitation from the layer above results in supersaturation, we repeat the calculations summarized in equations [II.51] and [II.58]. The moisture removed is treated as precipitation; otherwise, proceed to the next step.

- If the additional precipitation does not result in supersaturation at the next lower level, we add the moisture to the layer and decrease the temperature accordingly.

Finally, precipitation is given by the equation:

$$Pr = \frac{\epsilon 1000}{g} \ln(\sigma r) \Delta q \text{ (cm)} \quad [\text{II.59}]$$

where $r = 1.0/0.8$ or $0.8/0.6$ for the levels in the five layer models, and $r = 1.0/0.9$, $0.9/0.8$, $0.8/0.7$, $0.7/0.6$, or $0.6/0.5$ for the levels in the ten layer models.

3. Convective Adjustment

In the atmosphere, the direct consequence of vertical static instability is the initiation of small-scale vertical convection which is very active in transferring heat and moisture from the surface through the lowest layers of the atmosphere. This process is very instrumental in maintaining the overall potential energy of the atmosphere against its destruction by radiative absorption and emission.

In the forecast models, the primary purpose of the convective adjustment mechanism is the removal of vertical instabilities, whether they be generated by a large-scale dynamical process or by heat and moisture transfer from the surface to the lowest computational level of the model. If the vertical instabilities are allowed to grow, they will promote vertical convection whose scale is dictated only by the grid length in the model. This will lead to computational instability and destroy the solution. In order to control this type of instability and also to simulate vertical convection in the atmosphere, a convective adjustment process consisting of two parts, a parameterized cumulus convection and a dry convective adjustment, is performed at each time step.

a. Parameterized Cumulus Convection

Following a method devised by Arakawa et al (1969), three types of parameterized cumulus convections are considered. The first type is that occurring between levels 2 and 3; the second type is that occurring between levels 1, 2 and 3; and the third type is that occurring between levels 1 and 2. Precipitation may result from types 1 and 2, but not from type 3. Figure II-3 schematically illustrates these three types of parameterized convection.

Both the five-layer forecast models and the ten-layer forecast models use the same parameterized cumulus convection levels as shown in Figure II-3. In the case of the five-layer models, the convection levels coincide with sigma surfaces, while in the case of the ten-layer models, two adjacent sigma levels are combined to form a composite convection level with an averaged temperature and mixing ratio. Should cumulus convection then take place, the changes in moisture and temperature are apportioned between the two model layers on the basis of their mixing ratios.

Three parameters used to indicate which type of convection can occur are the energy integrals

$$S = C_p T + gz \quad [\text{II.60}]$$

$$h = C_p T + gz + Lq = S + Lq \quad [\text{II.61}]$$

$$E = C_p T + gz + Lq_s = S + Lq_s \quad [\text{II.62}]$$

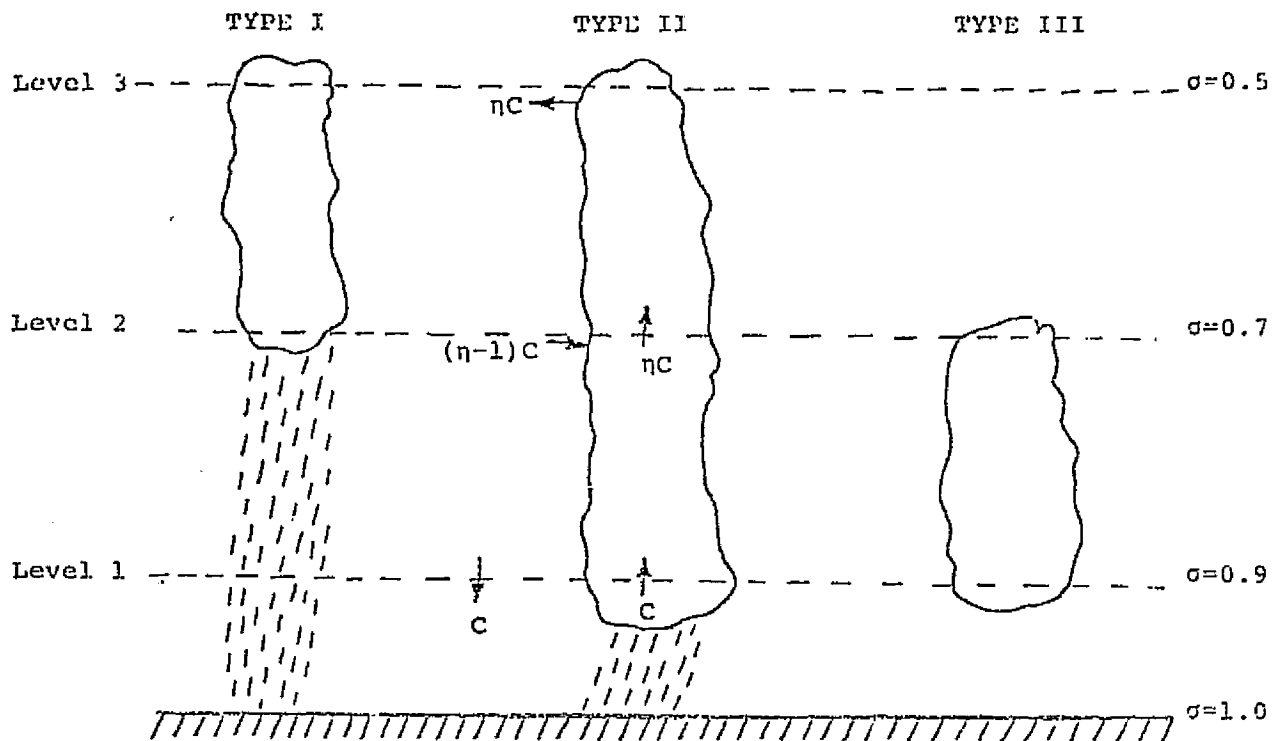


FIGURE II-3: SCHEMATIC REPRESENTATION OF THREE TYPES OF PARAMETERIZED CUMULUS CONVECTION, WHERE η IS THE ENTRAINMENT COEFFICIENT AND C IS THE CONVECTIVE MASS FLUX.

E is constant along a moist pseudo-adiabat and S is constant along a dry adiabat. Thus, comparison of h and E for various levels allows for a sufficient test of conditional instability.

Type 1 convection will occur if

$$h_2 > E_3 \quad [\text{II.63}]$$

If $E_2 = E_3$, a moist adiabatic lapse rate would exist between levels 2 and 3. Since $h_2 < E_2$, inequality [II.63] implies that a conditionally unstable lapse rate exists. Furthermore, inequality [II.63] can be rewritten as:

$$q_2 > \frac{1}{L} (E_3 - S_2) \quad [\text{II.64}]$$

Type 2 convection will occur if

$$E_3 > h_2 \quad \text{and} \quad h_1 > E_3 \quad [\text{II.65}]$$

Conditional instability does not exist here between levels 2 and 3, but does exist between levels 1 and 3.

Finally, type 3 convection will occur if:

$$E_3 > h_1 > E_2 \quad [\text{II.66}]$$

Levels 1 and 2 are conditionally unstable; levels 2 and 3 are not.

The procedure now is to write down equations describing the budget for the moisture, q, and the dry static energy, S, for each type of convection. Before this can be done, the

energy budget for the lower part of the cloud is considered, neglecting any accumulated storage of energy within the cloud.

We use type 2 convection as an example. When the entrainment coefficient, η , is greater than 1, entrainment exists and the parameter E in the clouds, E_c , is given by a mixing of h_1 and h_2 in the lower part of the clouds, as follows:

$$E_c = h_2 + \frac{1}{\eta} (h_1 - h_2) \quad [\text{II.67}]$$

because h is approximately conserved with respect to an air parcel. At level 3,

$$E_c = C_p T_{c3} + gZ_3 + Lq_s(T_{c3}) \quad [\text{II.68}]$$

whereas, for the environmental air:

$$E_3 = C_p T_3 = gZ_3 + Lq_s(T_3) \quad [\text{II.69}]$$

Therefore,

$$T_{c3} - T_3 = \frac{1}{1+\gamma_3} \frac{(E_c - E_3)}{C_p} \quad [\text{II.70}]$$

where

$$\gamma_3 = \frac{L}{C_p} \left. \frac{\partial q_s}{\partial T} \right|_3 \quad [\text{II.71}]$$

For type 1 and 3 convection, E_c is given by h_2 and h_1 , respectively. From equation [II.70], one can obtain approximately:

$$q_{c3} - q_{s3} = \frac{\gamma_3}{1+\gamma_3} \frac{(E_c - E_3)}{L} \quad [\text{II.72}]$$

For types 1 and 2, the convective terms in the moisture and dry static energy budget can be written in flux form as:

$$\frac{\Delta p_3}{g} \frac{\partial q_3}{\partial t} = \eta C \left[q_{s3} + \frac{\gamma_3}{1+\gamma_3} \frac{(E_c - E_3)}{L} - \frac{q_2 + q_3}{2} \right] \quad [\text{II.73}]$$

$$\frac{\Delta p_2}{g} \frac{\partial q_2}{\partial t} = \eta C \left[\frac{q_3 - q_2}{2} + \frac{1}{\eta} \frac{(q_2 - q_1)}{2} \right] \quad [\text{II.74}]$$

$$\frac{\Delta p_1}{g} \frac{\partial q_1}{\partial t} = C \left(\frac{q_2 - q_1}{2} \right) \quad (\text{type 2 only}) \quad [\text{II.75}]$$

$$\frac{\Delta p_3}{g} \frac{\partial s_3}{\partial t} = \eta C \left[\frac{E_c - E_3}{1+\gamma_3} + \frac{s_3 - s_2}{2} \right] \quad [\text{II.76}]$$

$$\frac{\Delta p_2}{g} \frac{\partial s_2}{\partial t} = \eta C \left[\frac{s_3 - s_2}{2} + \frac{1}{\eta} \left(\frac{s_2 - s_1}{2} \right) \right] \quad [\text{II.77}]$$

and

$$\frac{\Delta p_1}{g} \frac{\partial s_1}{\partial t} = C \left(\frac{s_2 - s_1}{2} \right) \quad (\text{type 2 only}) \quad [\text{II.78}]$$

For type 3 convection:

$$\frac{\Delta p_2}{g} \frac{\partial q_2}{\partial t} = - \frac{\Delta p_1}{g} \frac{\partial q_1}{\partial t} = C \left(\frac{q_1 - q_2}{2} \right) \quad [\text{II.79}]$$

and

$$\frac{\Delta p_2}{g} \frac{\partial S_2}{\partial t} = - \frac{\Delta p_1}{g} \frac{\partial S_1}{\partial t} = C \left(\frac{S_1 - S_2}{2} \right) \quad [\text{II.80}]$$

where η is the entrainment parameter and C is the convective mass flux.

Following Arakawa (1969) we now state that the solution of the above differential equations has a characteristic time scale, τ , such that:

$$\frac{\partial}{\partial t} (h-E) = - \frac{(h-E)}{\tau} \quad [\text{II.81}]$$

For type 1 convection, we can then obtain

$$\eta C = \frac{\Delta p}{g \tau_1} \left(\frac{1+\gamma_3}{2+\gamma_3} \right) \frac{(h_2 - E_3)}{[(h_2 - E_3) + \left(\frac{1+\gamma_3}{2} \right) (S_3 - S_2)]} \quad [\text{II.82}]$$

where τ_1 is given by the empirical relations:

$$\tau_1 = 3600 [1 + 0.667(h_2 - E_3)] \quad h_2 - E_3 < 0.5 \quad [\text{II.83}]$$

and

$$\tau_1 = 1200 [22(h_2 - E_3) - 7] \quad h_2 - E_3 \geq 0.5 \quad [\text{II.84}]$$

For type 2 convection, a value of 1°K was used for $T_c - T_3$;

therefore, we obtain:

$$\eta = \frac{h_1 - h_2}{E_3 + C_p (1+\gamma_3) - h_2} \quad [\text{II.85}]$$

and

$$C = - \frac{\Delta p}{g \tau_2} \frac{(h_1 - E_e)}{[\eta (1+\gamma_3) (C_p + \frac{S_3 - S_2}{2}) + (\frac{h_1 - h_2}{2})]} \quad [\text{II.86}]$$

where τ_2 is given by the empirical relations

$$\tau_2 = 3600 [1 + 2.25(h_1 - E_3)] \quad h_1 - E_3 < 0.5 \quad [\text{II.87}]$$

and

$$\tau_2 = 3600 [15.75(h_1 - E_3) - 5.75] \quad h_1 - E_3 \geq 0.5 \quad [\text{II.88}]$$

For type 3 convection, one can obtain:

$$C = - \frac{1}{\tau_3} \frac{(h_1 - E_2)}{\left[\left(\frac{2 + \gamma_2}{2} \right) (S_1 - S_2) - L \left(\frac{q_1 - q_2}{2} \right) \right]} \quad [\text{II.89}]$$

where τ_3 is given by the empirical relations:

$$\tau_3 = 3600 [1 + 0.667 (h_1 - E_2)] \quad h_1 - E_2 < 0.5 \quad [\text{II.90}]$$

and

$$\tau_3 = 1200 [22(h_1 - E_2) - 7] \quad h_1 - E_2 \geq 0.5 \quad [\text{II.91}]$$

Finally, precipitation from types 1 and 2 convection is given by

$$Pr = \left(\frac{1000 \cdot 600 \cdot 0.2}{980.6} \right) P_s (\Delta q_1 + \Delta q_2 + \Delta q_3) \quad [\text{II.92}]$$

b. Dry Convective Adjustment

The dry convective adjustment process consists of checking the model atmosphere for static stability and modifying the temperatures when necessary to achieve static

stability. This is accomplished by first computing the vertical profile of potential temperature, θ , by:

$$\theta_k = T_k \left(\frac{1000}{\pi \sigma_k} \right)^\kappa \quad [\text{II.93}]$$

where π is the terrain pressure, σ_k is the k^{th} sigma level and $\kappa = R/c_p = 0.2858$. Starting at the bottom of the profile, we then test to see if $\theta_k > \theta_{k-1} + \theta_s$, where $\theta_s = 3^\circ\text{K}$. If this is the case, the layer is stable and no temperature modification is necessary. If it is not the case, then the potential temperatures from layer 1 through k are modified to achieve a stable profile, using the average potential temperature and a fraction of the temperature difference θ_s per layer. (The fraction depends on the number of layers whose temperatures are being modified.) This process is continued until the entire profile has been checked for unconditional stability and modified, if necessary. The revised temperature profile is then recovered using equation [II.93].

D. Numerical Techniques

The numerical techniques used to approximate the solution of the forecast model equations are given in this section in two parts. First, the spatial finite differencing techniques are described. Then a description of the time differencing technique is given which includes an explanation of the pressure-gradient-force averaging technique, Robert time filtering and the non-linear pressure smoother.

1. Spatial Finite Differencing

Energy conserving (in space) difference equations based on the Arakawa technique (1966) are used to approximate the spacial derivatives. Thus, the finite difference equation for the momentum equation in the x direction is given by:

$$\left. \frac{\partial (\pi u)}{\partial t} \right|_*^n \sim [-L(u)_* + f(\pi_* v_*) - (\frac{\overline{\Delta \Phi}}{\Delta x})_*]^n - F_x^{n-1} + D_u^{n-1} \quad [\text{II.94}]$$

and in the y direction by:

$$\left. \frac{\partial (\pi v)}{\partial t} \right|_*^n \sim [-L(v)_* - f(\pi_* u_*) - (\frac{\overline{\Delta \Phi}}{\Delta y})_*]^n - F_y^{n-1} + D_v^{n-1} \quad [\text{II.95}]$$

where the subscript * denotes the point (i,j,k); the superscripts denote the time level; L is the Arakawa advective

operator; f is the Coriolis parameter; the Δ notation refers to the Kurihara pressure gradient; and the bar denotes time averaging as explained in the next section. F_x and F_y are the frictional terms; D_u and D_v are the lateral diffusion terms.

The Arakawa advective operator is given by:

$$\begin{aligned}
 L(\Psi)_* = & \frac{m_{i,j}^2}{4d} \left([(\alpha_{i+1} + \alpha_i)(\Psi_{i+1} + \Psi_i) - (\alpha_{i-1} + \alpha_i)(\Psi_{i-1} + \Psi_i)]_{j,k} \right. \\
 & + [(\beta_{j+1} + \beta_j)(\Psi_{j+1} + \Psi_j) - (\beta_{j-1} + \beta_j)(\Psi_{j-1} + \Psi_j)]_{i,k} \quad [\text{II.96}] \\
 & \left. + \frac{\pi_{i,j}}{2\Delta\sigma} [w_k(\Psi_{k+1} + \Psi_k) - w_{k-1}(\Psi_k + \Psi_{k-1})]_{i,j} \right)
 \end{aligned}$$

where $\alpha = \frac{u\pi}{m}$

$\beta = \frac{v\pi}{m}$

$m = \text{map factor}$

$d = \text{grid mesh size}$

The pressure gradient terms $\frac{\Delta\Phi}{\Delta x}$ and $\frac{\Delta\Phi}{\Delta y}$ are computed using a type of modification suggested by Kurihara (1968). Instead of evaluating both the gradient of the geopotential distribution on a sigma surface and the terrain pressure gradient (invariant with height) as would be required by equations [II.1] and [II.2], the value of the geopotential is interpolated at neighboring grid point locations to the same pressure surface as at the computation point, allowing computation of the geopotential gradients on pressure surfaces.

The interpolation formula used is a form of:

$$\frac{\partial \phi}{\partial \ln \sigma} = -RT \quad [\text{II.97}]$$

This procedure of differencing ϕ on local pressure surfaces reduces the inconsistent truncation error over high, irregular terrain.

Frictional dissipation is accounted for in the gross boundary layer ($\sigma = 1.0 - 0.8$ for the five-layer models and $\sigma = 1.0 - 0.9$ for the ten-layer models) only. The stress is assumed to vanish at the top of this layer. The stress terms are the same as those given by Shuman and Hovermale (1968), except that the surface wind speed is obtained by extrapolation from the boundary layer sigma surface. Thus, the friction in the x direction is given by:

$$F_x^{n-1} = \frac{g C_D}{\Delta \sigma R} V_s^{n-1} \left(\frac{\pi u}{T} \right)_{BL}^{n-1} \quad [\text{II.98}]$$

and in the y direction by

$$F_y^{n-1} = \frac{g C_D}{\Delta \sigma R} V_s^{n-1} \left(\frac{\pi v}{T} \right)_{BL}^{n-1} \quad [\text{II.99}]$$

$$\text{where } V_s = 0.3 [u_{BL}^2 + v_{BL}^2]^{1/2} \quad [\text{II.100}]$$

and $C_D = 0.0015$ for sea level and 0.0025 for non-sea level terrain.

The lateral diffusion terms are given by

$$D_u = D \frac{m_{i,j}^2}{d^2} (\pi \nabla^2 u)_*^{n-1} \quad [\text{II.101}]$$

$$D_v = D \frac{m_{i,j}^2}{d^2} (\pi \nabla^2 v)_*^{n-1} \quad [\text{II.102}]$$

where the Laplacian is

$$\nabla^2(\psi)_* = \psi_{i+1,j} + \psi_{i-1,j} + \psi_{i,j+1} + \psi_{i,j-1} - 4\psi_{i,j} \quad [\text{II.103}]$$

and the diffusion coefficient, D , varies with latitude according to the relation $K (1 - \sin^2 \phi)$ with 0.9 taken as a maximum for $\sin \phi$. K is taken to be 10^6 for the 63×63 models and 4×10^5 for the 187×187 models.

The change in the terrain pressure is computed from an approximate form of the continuity equation as:

$$\left. \frac{\partial \pi}{\partial t} \right|_{i,j}^n \sim -\frac{1}{K} \sum_{k=1}^K \frac{m_{i,j}^2}{2d} (\alpha_{i+1,j,k} - \alpha_{i-1,j,k} + \beta_{i,j+1,k} - \beta_{i,j-1,k})^n \quad [\text{II.104}]$$

The thermodynamic energy equation is approximated by:

$$\left. \frac{\partial(\pi T)}{\partial t} \right|_*^n \sim -L(T)_*^n \quad [\text{II.105}]$$

$$\begin{aligned} & + \frac{RT_*}{cp} \left(-\frac{\pi_{i,j}}{2\sigma_k} (w_k + w_{k-1})_{i,j} + \left(\frac{\partial \pi}{\partial t} \right)_{i,j} \right. \\ & + \frac{m_{i,j}}{2d} [u_{i,j,k} (\pi_{i+1,j,k} - \pi_{i-1,j,k}) \\ & + v_{i,j,k} (\pi_{i,j+1,k} - \pi_{i,j-1,k})] \left. \right)^n \\ & + H_* \pi_{i,j}^{n-1} + D_T^{n-1} \end{aligned}$$

where D_T is given by an expression similar to those for D_u and D_v (Equations II-101 and II-102).

Vertical motions for non-material surfaces are obtained from a form of the continuity equation:

$$\begin{aligned} w_{i,j,k+1} = w_* - \frac{\Delta \sigma}{\pi_{i,j}} \left(\left(\frac{\partial \pi}{\partial t} \right)_{i,j} \right. \\ \left. + \frac{m_{i,j}^2}{2d} [(\alpha_{i+1} - \alpha_{i-1})_{j,k} + (\beta_{j+1} - \beta_{j-1})_{i,k}] \right) \end{aligned} \quad [\text{II.106}]$$

The hydrostatic equation is given at the lower level by:

$$\phi_* = \phi_T - RT_* \ln(\sigma_1) \quad [\text{II.107}]$$

where T_* is extrapolated from the lowest two levels by $1.25T_1 - 0.25T_2$ and ϕ_T is the terrain geopotential. For subsequent levels

$$\phi_{k+1} = \phi_k - \frac{R}{2} \ln \left(\frac{\sigma_{k+1}}{\sigma_k} \right) (T_k + T_{k+1}) \quad [\text{II.108}]$$

Finally, the moisture equation is approximated by

$$\left. \frac{\partial (\pi q)}{\partial t} \right|_*^n \sim -L(q)_*^n + Q_* \pi_{i,j}^{n-1} \quad [\text{II.109}]$$

2. Temporal Finite Differencing

A centered, or leap-frog, difference scheme is used to approximate the time derivative in the forecast model equations; that is

$$\left. \frac{\partial A}{\partial t} \right|_*^n \sim \frac{1}{2\Delta t} (A_*^{n+1} - A_*^{n-1}) \quad [\text{II.110}]$$

In order to lengthen the permissible time step used in the computation, a numerical technique known as pressure-gradient-force time averaging is used in conjunction with the centered time differencing. This procedure was suggested by Shuman (1971) and has been successfully used at the National Meteorological Center. Pressure-gradient-force time averaging requires arranging the integration of the primitive equations into two parts. First, the pressure tendency, thermodynamic energy and moisture equations, including the diabatic terms, are integrated forward to the

(n+1) time level. Then the momentum equations are integrated forward in time to the (n+1) time level, but the pressure gradient force terms (refer to Equations [II.94] and [II.95]) are time-averaged according to the relation:

$$\overline{\text{PGF}}^n = (1-\alpha)\text{PGF}^n + \frac{\alpha}{2}(\text{PGF}^{n-1} + \text{PGF}^{n+1}) \quad [\text{II.111}]$$

For stability, $0 < \alpha < 0.5$, and it can be shown that when $\alpha = 0.5$, a time step twice as long as when $\alpha = 0$ can be used. However, a weak computational instability is present and a value of α slightly less than one-half should be used. The value $\alpha = 0.49$ is used in the forecast models.

A centered difference scheme is neutrally stable in that there is no damping, only phase shift, of the various frequency components present in the solution of the difference equations. In order to achieve damping of the shorter wave length components to avoid non-linear instability and, in particular, to damp the computational mode, the simple leap-frog scheme is modified slightly to time-filter the solutions of the thermodynamic and moisture equations. This procedure is known as time filtering and was originally used by Robert (1966).

Assume that the partial differential equation that we wish to approximate is

$$\frac{\partial A}{\partial t} = F(A) \quad [\text{II.112}]$$

Then, using the leap-frog difference scheme, we have

$$A^{n+1} = A^{n-1} + 2\Delta t F(A) \quad [\text{II.113}]$$

This step is modified by Robert to become

$$(A')^{n+1} = A^{n-1} + 2\Delta t F(A, A') \quad [\text{II.114}]$$

$$A^n = (1-\alpha) (A')^n + \frac{\alpha}{2} [A^{n-1} + (A')^{n+1}] \quad 0 < \alpha < 1 \quad [\text{II.115}]$$

where we time-filter the solution. It can be shown that this procedure adds damping to the solution and heavily damps the computational mode that is present in the solution of the finite difference equations.

If $\omega\Delta t < 1$ is the linear stability criteria for the simple leap-frog difference scheme, then the addition of Robert time filtering will change the stability criteria to:

$$\omega\Delta t < \left[\frac{2-\alpha}{2+\alpha} \right]^{1/2} \quad [\text{II.116}]$$

A value of $\alpha=0.1$ is used to time-filter the solution of the thermodynamic and moisture equations. No time filtering of the solutions of the momentum equations is done since lateral diffusion terms are included in these equations.

The terrain pressure tendency equation [II.104] is essentially an ordinary differential equation; thus, rather than using Robert time filtering of the solution to control small-scale noise and solution separation, a non-linear

smoothing operator is applied every time step. This smoothing operator has been described by Oliger and Welck (1970) and is given by:

$$\begin{aligned} \Delta P_s = \frac{\epsilon}{|\delta P_s|_{\max}} [& |P_s^n_{i+1,j} - P_s^{n+1}_*| (P_s^n_{i+1,j} - P_s^{n+1}_*) \\ & - |P_s^{n+1}_* - P_s^n_{i-1,j}| (P_s^{n+1}_* - P_s^n_{i-1,j}) \\ & + |P_s^n_{i,j+1} - P_s^{n+1}_*| (P_s^n_{i,j+1} - P_s^{n+1}_*) \\ & - |P_s^{n+1}_* - P_s^n_{i,j-1}| (P_s^{n+1}_* - P_s^n_{i,j-1})] \end{aligned} \quad [\text{II.117}]$$

where $|\delta P_s|_{\max}$ is the maximum absolute difference between P_s at the point (i,j) and P_s at any of the four immediately neighboring points, and ϵ is the smoothing coefficient which has been determined to be of the order of 0.01 for control of computational noise.

The smoothed surface pressure is then given by:

$$P_s^{n+1} (\text{smoothed}) = P_s^{n+1} + \Delta P_s \quad [\text{II.118}]$$

The terrain pressure is reduced to sea level pressure and vice versa through the relations

$$P_s^n = \pi^n + \Delta \pi^n \quad [\text{II.119}]$$

and

$$\Delta \pi^n = \Delta \pi_0 \left[1 + \left(\frac{\pi^n}{\pi_0} - 1 \right) \left(\frac{\pi^n - 700}{300} \right) \right] \quad [\text{II.120}]$$

where π_0 is the initial terrain pressure and $\Delta\pi_0$ is the initial difference between sea level and terrain pressure. These relations are also used to create surface pressure fields for output.

Due to the interaction of these various computational devices described above (pressure-gradient-force time averaging, Robert time filtering, lateral diffusion and non-linear pressure smoothing) a doubling of the permissible time step is not possible. However, a fifteen-minute time step for the 63 by 63 models as opposed to a ten-minute time step, and a four-minute time step for the 187 by 187 models is possible which results in a considerable saving of computer time over simple centered time differencing.

Finally, the integrations are restarted after each output cycle (normally every twelve model-hours) using an Euler-backward or Matsuno time step in the 63 x 63 models. Provision is made to perform an Euler-backward time step (which is highly dissipative) at specified intervals in the 187 x 187 models. This integration step is given by:

$$(A')^{n+1} = A^n + \Delta t F[A^n] \quad [\text{II.121}]$$

and

$$A^{n+1} = A^n + \Delta t F[(A')^{n+1}] \quad [\text{II.122}]$$

E. Initialization

In order to be able to forecast the meteorological conditions at some time in the future using a primitive equation model, the conditions at the present time must be known; that is, an initial state must be specified for the variables forecast by the PEM. This initialization process is accomplished by using analyzed data produced by a set of analysis programs and synthesizing this data into a complete specification of the initial state for the PEM.

The following sections describe the method of interpolating the analyzed data fields to the forecast model sigma surfaces. The specification of various other required data fields is also described.

1. Constant Fields

Constant data fields are of two types -- those that are truly constants (sine of latitude, map factor, tendency truncation coefficient and terrain height) and those that, while not constants, are held constant during the course of a forecast (sea surface temperature, land-sea-ice indicator and albedo).

For a polar stereographic grid, true at 60°N and centered about the North Pole, the sine of latitude is given by:

$$\sin \phi = \frac{STP - ST}{STP + ST} \quad [II.123]$$

where

$STP = (31.205)^2$ for the 63 by 63 grid

$STP = (93.615)^2$ for the 187 by 187 grid

and

$ST = (I-POL)^2 + (J-POL)^2$

$POL = 32$ for the 63 by 63 grid

$POL = 94$ for the 187 by 187 grid

The map factor is given by

$$m = \frac{1 + \sin 60^\circ}{1 + \sin \phi} \quad [II.124]$$

The tendency truncation coefficient used to reduce the tendencies of the variables in the outer regions of the grid is given by:

$$\begin{aligned} re &= 1.0 \text{ for } \phi \geq 20^\circ \\ &= 0.0 \text{ for } \phi \leq 5^\circ \end{aligned} \quad [II.125]$$

and varies as $[(\sin \phi - \sin 5^\circ) / (\sin 20^\circ - \sin 5^\circ)]^2$ for ϕ between 5° and 20° .

The terrain height is derived from Scripps Institute of Oceanography data and has been area-averaged and gradient-limited. That is, the terrain gradient at any grid point is constrained to be less than 2000 m per 381 km, the basic 63 by 63 grid size. In addition, the terrain is multiplied by the tendency truncator thereby tapering it towards sea level around the edges of the grid.

The land-sea-ice indicator is also derived from the geography and varies seasonally as ice masses expand and contract.

The albedo, ALB, of the earth's surface is obtained from the seasonal albedo charts of the northern hemisphere developed by Posey and Clapp. The open sea albedo is modified in the heating computation as a function of the zenith angle of the sun according to the relation:

$$ALB' = ALB + 0.54 (0.7 - \cos \mu) \quad [II.126]$$

The sea surface temperature is one of the analyzed data fields produced by the pattern conservation technique analysis programs.

2. Terrain Pressure

The terrain pressure, π , is determined from the analyzed sea level pressure, P_s , the terrain height, Z_T , and the heights of the two analyzed pressure surfaces bounding the terrain height. That is, if the terrain height lies between Z_L and Z_U , which correspond to the heights of the pressure surfaces, P_L and P_U , then the terrain pressure is given by

$$\pi = P_U \left(\frac{P_L}{P_U} \right)^X \quad \text{where } X = \frac{Z_U - Z_T}{Z_U - Z_L} \quad [II.127]$$

The difference between the sea level pressure, P_s , and the terrain pressure, π , is also computed and stored for use in reducing the forecast terrain pressure to give a forecast sea level pressure.

3. Temperature

The sigma surface temperatures are determined by logarithmic interpolation between the analyzed pressure surfaces. The terrain pressure, π , and the temperatures of the analyzed pressure surfaces, are used to determine the temperatures of the sigma surface from the following relation:

$$T_{\sigma} = T_L + \left[\frac{\ln(P_L) - \ln(\sigma\pi)}{\ln(P_L) - \ln(P_U)} \right] (T_U - T_L) \quad [\text{II.128}]$$

That is, the pressures, P_L and P_U , are found which bound the sigma surface, $\sigma\pi$, and along with the corresponding temperatures, T_L and T_U , are used in equation [II.128].

4. Geopotentials

Having computed a set of sigma surface temperatures, and having obtained the terrain geopotential, the hydrostatic equation can be used to compute an initial set of geopotentials. That is, equations [II.107] and [II.108] are used to compute the sigma surface geopotentials.

5. Winds

The initial winds are obtained from an analysis procedure which gives analyzed winds on pressure surfaces from 1000 to 100 mb, and are computed geostrophically on the 50 mb pressure surface. The 63 x 63 models use wind data on eleven pressure surfaces: 950, 850, 700, 500, 400, 300, 250, 200, 150, 100 and 50 mb. The 187 x 187 models use

wind data on all thirteen of the pressure surfaces. The initial winds on pressure surfaces are then vertically interpolated linearly in the logarithm of pressure to the model sigma surfaces using equation [II.128].

6. Moisture

An important part of the initialization process is the specification of the initial moisture distribution. The forecast models carry moisture at the lower three levels in the five-layer models and at the lower six levels in the ten-layer models and would, thus, require a large amount of analyzed moisture data for initialization. Since moisture data are virtually non-existent over the oceans, a parameterized approach to the specification of the initial moisture distribution has been taken.

Studies by Kesel and Lewit (1974) showed that existing analysis/procedures for moisture initialization do not necessarily result in an optimum moisture distribution. It was further shown that improved forecasts could result from moisture initialization based upon the relative vorticity distribution at each model level.

ODSI's studies have shown that the initialization of moisture has a significant effect on deepening lows, but has little effect on filling lows or highs. It was shown that by basing the initial relative humidity on the intensity

of the geostrophic relative vorticity, specific regions of properly posed relative humidity values will more likely coincide with actively developing systems.

Therefore, a procedure based on the intensity of the geostrophic relative vorticity is used to define the initial moisture distributions on the 1,000, 900, 700, 500 and 300 mb pressure surfaces. These values are then interpolated linearly in the logarithm of pressure using equation [II.128] to the sigma surfaces. This procedure for each of the five pressure surfaces can be outlined as follows:

- a. Compute the geostrophic relative vorticity for each point in the field using the relation

$$RV = \frac{g}{f} \nabla^2 Z \quad [\text{II-129}]$$

where Z is the height in meters of the pressure surface and f is the Coriolis parameter.

- b. Compute the mean and standard deviation of the relative vorticity for the field using the relations

$$M = \frac{1}{N} \sum^N RV_{i,j} \quad [\text{II.130}]$$

$$SD = \left[\frac{1}{N} \sum^N RV_{i,j}^2 - M^2 \right]^{1/2} \quad [\text{II.131}]$$

- c. Normalize the relative vorticity field according to the relation

$$RVN_{i,j} = \frac{RV_{i,j} - M}{SD} \quad [\text{II.132}]$$

d. Compute the relative humidity for each point based on the normalized relative vorticity values using the empirical relation

$$RH_{i,j,k} = REL_k (1 + RVN_{i,j}) \quad [II.133]$$

with the upper limit of relative humidity set at 100% and the lower limit set at 20%. The parameter REL depends on the pressure surface and is given by 0.7, 0.6, 0.5, 0.4 and 0.3 for 1000 through 300 mb, respectively.

e. Modify the relative humidities computed using equation [II.133] according to whether they are over ocean or land or are in thermal lows (summers). If the point is over the ocean, then the computed humidity is unchanged, while if the point is over land the computed humidity is multiplied by 0.8. Further, if the point is in a thermal low, then the computed humidity is multiplied by 0.5. A thermal low is defined as existing when the day as measured from January 1 is between 150 and 250 (summer), and either the 1000 mb - 500 mb thickness is greater than 5790 meters or the 500 mb temperature is greater than -5°C .

III. COMPUTER PROGRAM DESCRIPTION OF THE FORECAST MODELS

The designer of any large, time-consuming computer code is faced with numerous problems in trying to design an efficient and readable program. First, there are the problem constraints on how many grid points are desired, how many variables are to be carried at each point, exactly what must be computed, what are the inputs and what are the desired outputs. Second, there are the hardware constraints -- the size of main memory, the type of secondary storage available (rotating or extended memory), the number and characteristics of the I/O channels and the speed of the CPU. Third, there are the software system constraints -- exactly what will the system allow one to do, what sort of high-level language is available and what are its characteristics. Finally, there is the requirement that the resulting code should be efficient, readable and easily modified, both from the standpoint of computational changes and portability between computers.

What follows is a description of the realization of these various design objectives, including the various compromises that were required. There are four forecast models described that range in data base size from approximately 200,000 to 3,000,000 words and span roughly a factor of fifty in computational requirements. The forecast models

use two basic data flow structures and are grouped as follows: the five- and ten-layer, coarse horizontal grid models, PECHCV and PECHFV, and the five- and ten-layer fine horizontal grid models, PEFHCV and PEFHFV.

During the course of developing these forecast models, three different computer systems were utilized: a CDC 6500, a CDC CYBER 175 and a CDC CYBER 76, all using CDC software. Thus, an attempt at portability was required and largely realized, although the forecast models are in no sense totally portable between all computer systems with FORTRAN IV as their high-level language.

It is not the intent of this section to give complete detail down to the definition of the last local variable in each subprogram, but to describe the data structure of each program and to describe the code structure with sufficient detail to be easily understandable without overwhelming the reader with trivia. Also, since there is a large amount of duplication of code between the four forecast models, differentiation between programs in the explanations which follow is only made when necessary.

Section A explains the data structures of the four forecast models. Section B describes the program structures of models PECHCV and PECHFV and Section C describes the program structures of models PEFHCV and PEFHFV. Included in these sections are block diagrams, definitions of the various common blocks and functional descriptions of each of the subprograms.

A. Data Structure

One of the most critical factors involved in attempting to design an efficient computer program is an efficient data storage scheme, since a primitive equation forecast model (PEM) requires the handling of large amounts of data. Section 1 gives a general picture of the data storage philosophy used in the models. Section 2 then discusses the details of the two versions of data storage structure used in the models giving diagrams, tables and illustrative examples of the working of this method of data storage.

1. Data Storage - Philosophy

Most PE models numerically approximate the solution of a complicated set of partial differential equations -- the primitive equations -- through the use of finite differences. While these PEM are three-dimensional in space, the treatment of the third space dimension, the vertical dimension, is quite different from the two horizontal dimensions. The use of the hydrostatic assumption to render the numerical solution feasible and vertical column computations in the heating packages immediately offer themselves as obvious examples of the difference between the horizontal and the vertical dimensions.

If one assumes some sort of grid system, for example, either a polar stereographic projection or a

spherical polar grid, one then has an array of data, say $M \times N$ in the horizontal with K vertical levels. Since there will be at least five variables at every point, the total number of pieces of information can be quite large, on the order of millions of words. Except for the very largest computer memories, this vast amount of data, together with the necessary computer code, cannot be contained in memory. Thus, one is faced with the problem of devising some sort of efficient storage scheme for the data.

Meteorologists are accustomed to looking at weather data on pseudo-horizontal surfaces; for example, the geopotential heights at 500 mb. Very few outputs from a PEM are presented as vertical slices. This, of course, is physically quite reasonable, since the scales of motion are quite different between the horizontal and the vertical, as are the grid distances in a PEM. However, horizontal planes are not the best way to organize the data storage for computation in a PEM.

Since it has been assumed that the numerical method used to approximate the solution of the primitive equations is finite differencing, it is easily seen that an entire horizontal plane is not necessarily needed in memory simultaneously. The second order finite difference approximations for horizontal derivatives used for the forecast models are given by

$$\left. \frac{\partial u}{\partial x} \right|_{i,j,k} \sim \frac{1}{2\Delta x} (u_{i+1,j} - u_{i-1,j})_k \quad [\text{III.1}]$$

and

$$\left. \frac{\partial u}{\partial y} \right|_{i,j,k} \sim \frac{1}{2\Delta y} (u_{i,j+1} - u_{i,j-1})_k \quad [\text{III.2}]$$

If one imagines the j subscript being held fixed and i varying, then there are only three j rows involved in the computation along a single j row, the one being computed, the one below and the one above. In addition, since the heating package computations are all performed on vertical columns, it would seem reasonable to use a data storage scheme such as the one shown in Figure III-1.

The scheme shown in Figure III-1 has the computational data broken down into vertical slices which are $M \times K$. There are N of these slices. It has been seen that only three of these slices are needed at any one time to compute along a given j row. Thus, if one were to construct a file with N records on it, each containing one vertical slice, one could perform a forward integration of the primitive equations by sweeping this file in sequence. Of course, a new file should be constructed as this file is swept, in order to have data for the next step. The details of such a file scheme will be explained in the next section.

A PEM consists of three basic sections, the initialization section, the computational section which

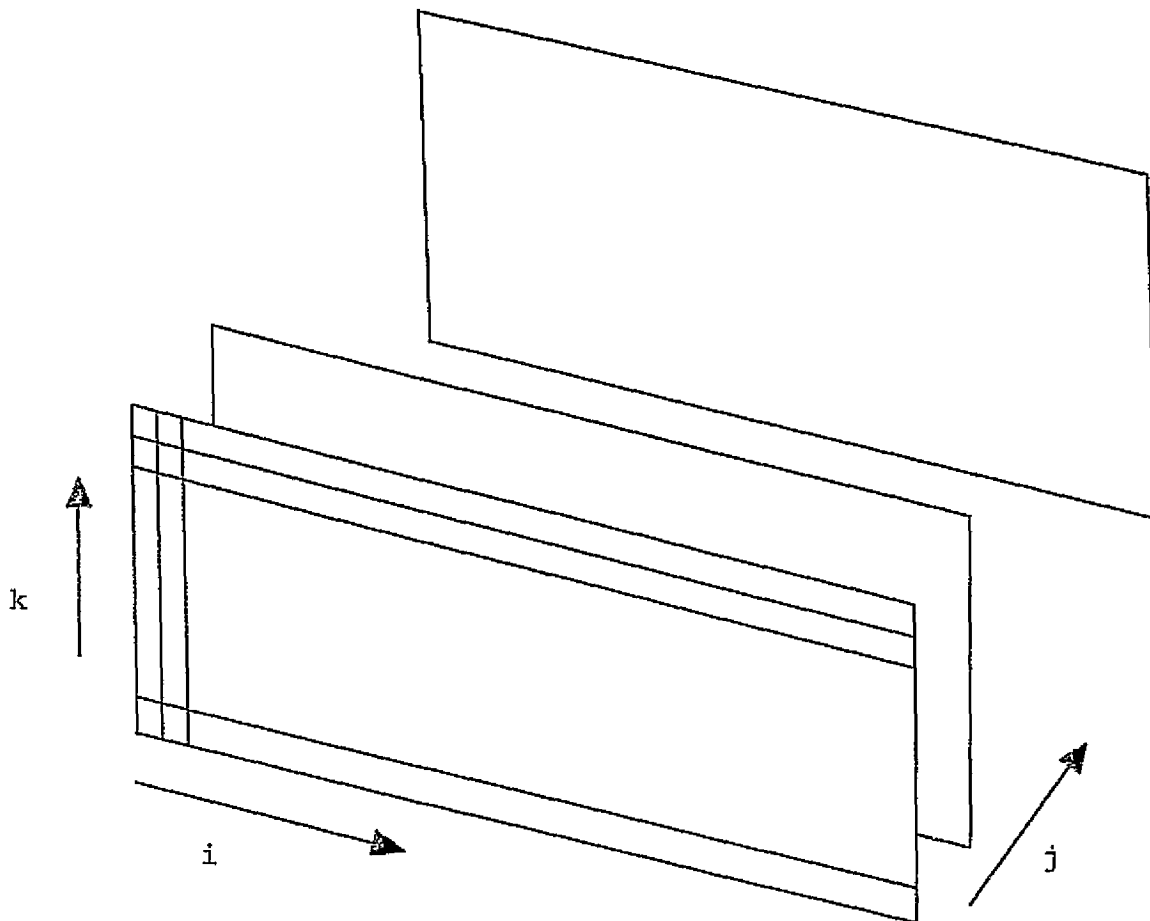


FIGURE III-1: ROW OR VERTICAL SLICE
DATA STORAGE SCHEME

actually integrates the primitive equations, and an output section. By far the most computer time is spent in the computational section; thus, it seems reasonable to arrange the data storage for maximum convenience of this section of the code. This hypothetical PEM would take, as input, data on horizontal planes (pressure surfaces), since this is the way that analyses are performed. It would then interpolate this data to the sigma surfaces and finally sort out this data into vertical slices for use by the computational section. The computational section would then integrate the data forward in time to obtain a forecast and pass it to the output section. The output section will then sort out the data into horizontal planes and interpolate, smooth, etc., for viewing by the users of the forecast.

An obvious question that arises is, does this method of data storage result in an efficient PEM from the standpoint of central memory and computational speed? The answer is that it does and that this method of data storage is in use at numerous meteorological centers. As will be explained in the next section, rotating storage can be used quite effectively and more efficient utilization of central memory can be obtained when using the vertical slice storage technique.

2. Data Storage - Realization

The concept of a data file consisting of N records, each representing a vertical slice of the three-dimensional grid system, was introduced in the previous section. If the grid is $M \times N \times K$, then there will be N records on the file. Each record will hold data for an $M \times K$ vertical slice. It was also pointed out that, in order to perform computations on the j row, only the j-1 and j+1 rows are needed in addition to the j row. In this section, the file structure, memory storage requirements and data flow will be examined in detail.

For the sake of clarity, let us call the existing data file IOLD and the data file created by one sweep of the data base INEW. If it is assumed that centered time differencing is used, i.e.,

$$\left. \frac{\partial A}{\partial t} \right|_n \sim \frac{1}{2\Delta t} (A^{n+1} - A^{n-1}) \quad [\text{III-3}]$$

and that the numerical technique of pressure-gradient-force averaging is used, then file IOLD will contain information at the n-1 and n time levels. After the first sweep of IOLD, INEW will contain partially updated information at the n and n+1 time levels (temperatures, moistures and geopotentials) along with the non-updated information. At the end of the sweep, the file INEW becomes the file IOLD and a second sweep of the data is made to integrate the momentum equations

thereby completing the time step. The file INEW now contains all updated data at time levels n and n+1. This process is repeated until the desired forecast time is reached.

In order to achieve more efficient use of central memory and to reduce the size of the data files, variables with time levels are "time packed" on a 2:1 basis. That is, the n time level occupies the upper thirty bits of the word and the n-1 time level occupies the lower thirty bits of the word. Certain other infrequently referenced variables are also packed on a 2:1 basis. A set of constructs which take the form of FORTRAN statement functions have been designed to be able to manipulate these packed data words. These statement functions are listed below with a brief explanation of their function:

retrieving the n-1 time level

or lower value -

$$\text{SNM}(V) = \text{SHIFT}(V, 30) \quad [\text{III.4}]$$

pack two values into a word -

$$\begin{aligned} \text{PACK}(VNP, VN) = & (VNP.\text{AND}.77\dots70\dots0B).\text{OR}. \\ & (\text{SHIFT}(VN, -30).\text{AND}.0\dots07\dots7B) \quad [\text{III.5}] \end{aligned}$$

store a value into the top half

of a packed word -

$$\begin{aligned} \text{STOR}(VNP, VN) = & (VNP.\text{AND}.77\dots70\dots0B).\text{OR}. \\ & (VN.\text{AND}.0\dots07\dots7B) \quad [\text{III.6}] \end{aligned}$$

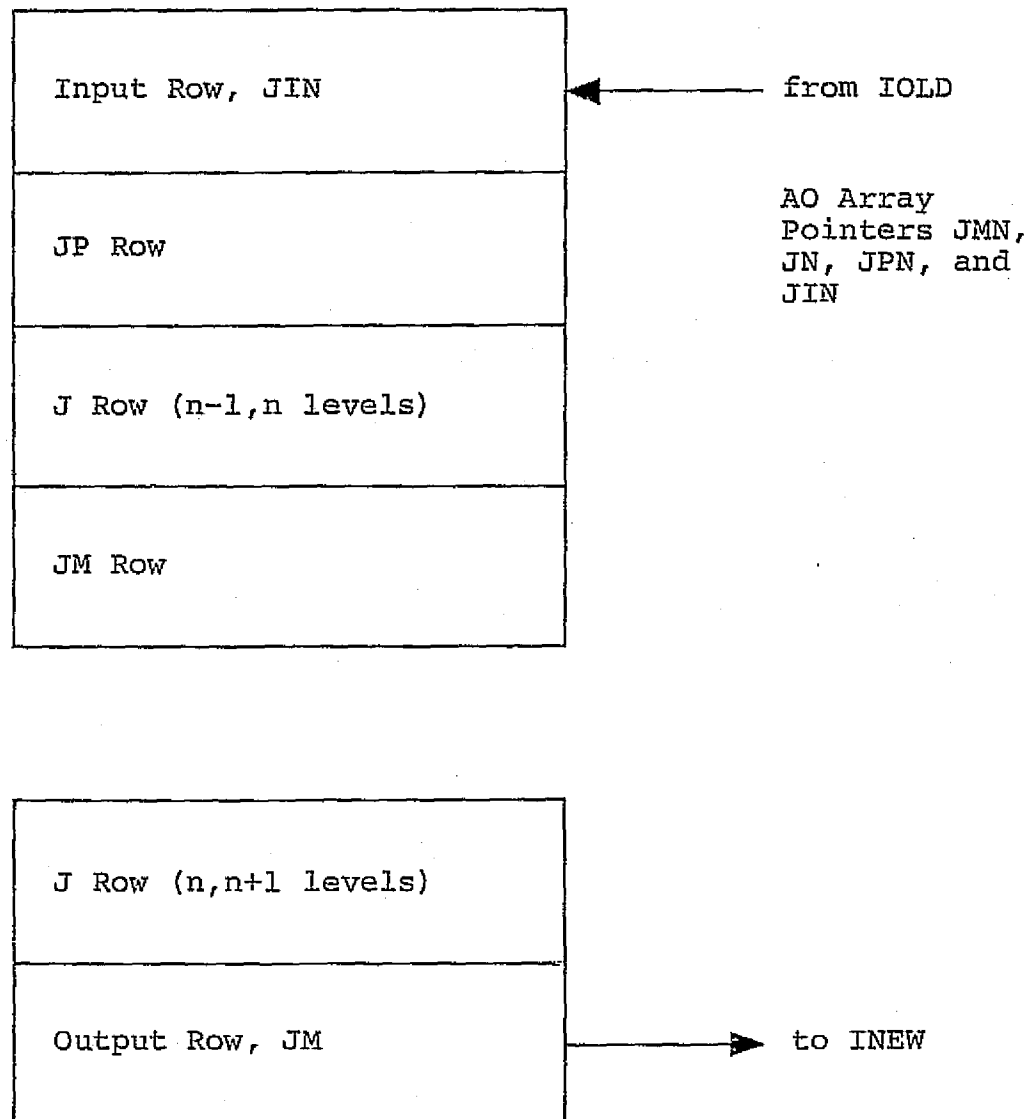
Since this "time packing" method relies on the CDC FTN compiler intrinsic function SHIFT, the forecast models are not portable to other compilers. However, this was thought to be worthwhile since it affords a considerable compression in data file size and has a minimal effect on the computation time (approximately a 3% increase).

a. Data Storage - PECHCV and PECHFV

The concept of an old data file being read and a new data file being created on each sweep of the data base was presented in the previous section. In addition, the idea of "time packing" and the necessary methods of accessing the packed data were also presented. The next item to be considered is an efficient means of flowing the data to and from central memory in order to implement the computation.

A diagram of the necessary central memory storage buffers for this data storage scheme to work is given in Figure III-2. Two buffer areas are required, AO for the three working rows and the row being input, and AN for the row being created and the row being output. It has been assumed that the computer hardware will allow simultaneous I/O and computation-buffered computation; thus, the row JIN in array AO and the row JM in array AN. Further, since it is time consuming to move data about in memory, the buffers AO and AN are circular buffers, with the pointers redefined as the computation proceeds.

One more construct is needed before a given piece of data can be accessed in the arrays AO and AN. This is a method of pointing to the particular variable that is required on the particular row that is desired. For computational convenience, two subscript functions have been



An Array
Pointers JMNP and JNP

FIGURE III-2: CENTRAL MEMORY STORAGE
BUFFERS REQUIRED FOR ROW
STORAGE SCHEME USED IN
MODELS PECHCV AND PECHFV

defined, one for data that exists on only one vertical level, and one for data that exists on numerous vertical, or K, levels. These subscript functions use the row pointers, pointers to the variables in a row which are known as offsets, and the vertical level, k, for the second type of subscript function.

The two subscript functions, which take the form of FORTRAN statement functions, are defined below.

$$\text{ISUBS}(J, \text{IOFF}) = J * \text{NVAR} + \text{IOFF} \quad [\text{III.7}]$$

for data on only one vertical level and

$$\text{ISUBK}(J, \text{IOFF}, K) = J * \text{NVAR} + \text{IOFF} + (K-1) * \text{IMAX} \quad [\text{III.8}]$$

for data on many vertical levels. In these definitions,

J = Pointer to the row being referenced in the buffer array

IOFF = Offset of the particular variable desired

K = Vertical level desired

NVAR = Total number of words per row record

IMAX = Total length of a row (range on I)

It is assumed that the pointers to the AO array (JMN, JN, JPN and JIN) range between 0 and 3 and that the pointers to the AN array (JMNP and JNP) range between 0 and 1. These subscript functions then point to the I = 0 element for ISUBS and the I = 0 element on level K for ISUBK.

A simple example of the use of the subscript functions is called for at this point. Assume that it is desired to compute $\frac{\partial u}{\partial x}$ on line J, level K; then the FORTRAN code for

doing this would be as follows:

```
DO 10 K=1,KMAX
IU=ISUBK(JN,UN,K)
DO 10 I=2,IMAXM
DU(I,K)=(AO(IU+I+1)-AO(IU+I-1))/TDX
10 CONTINUE
```

The mechanics for accessing data in the storage buffers AO and AN have now been constructed. The data access mechanism together with the buffer structure shown in Figure III-2 allows a buffered computation to be performed; that is, simultaneous computations and I/O to the files IOLD and INEW. The next, and final, step is to put these ideas together to generate the data flow for one complete time step. The data flow shown assumes that free slip, insulated wall boundary conditions are used. Figure III-3 shows the data flow for one sweep of the data file for the forecast models PECHCV and PECHFV.

Finally, Tables III-1 and III-2 define the offsets which are used by the two forecast models PECHCV and PECHFV in the pointer scheme to access data in the buffer arrays AO and AN.

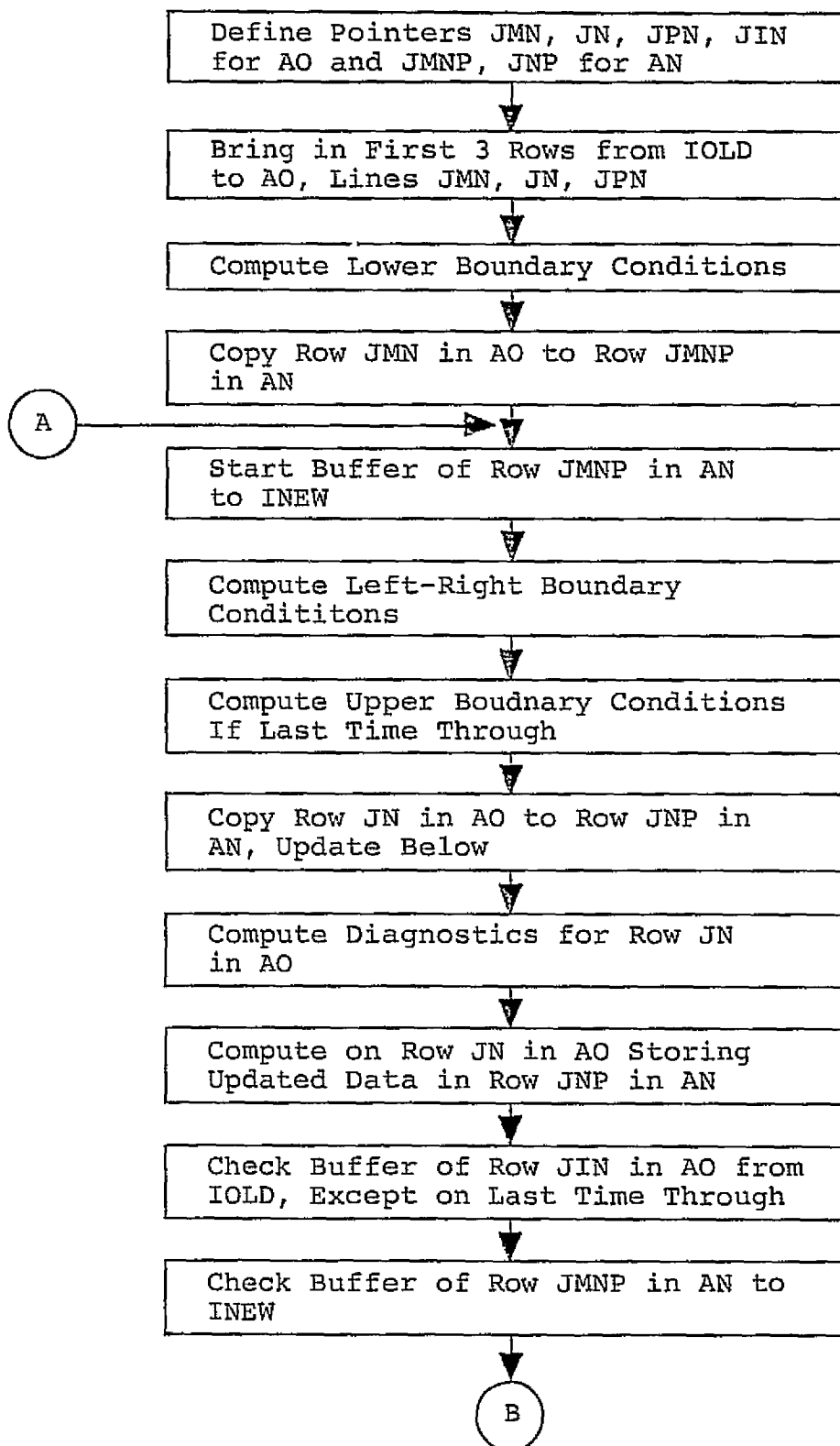


FIGURE III-3: DATA FLOW FOR ONE SWEEP OF THE DATA FILE - MODELS PECHCV AND PECHFV

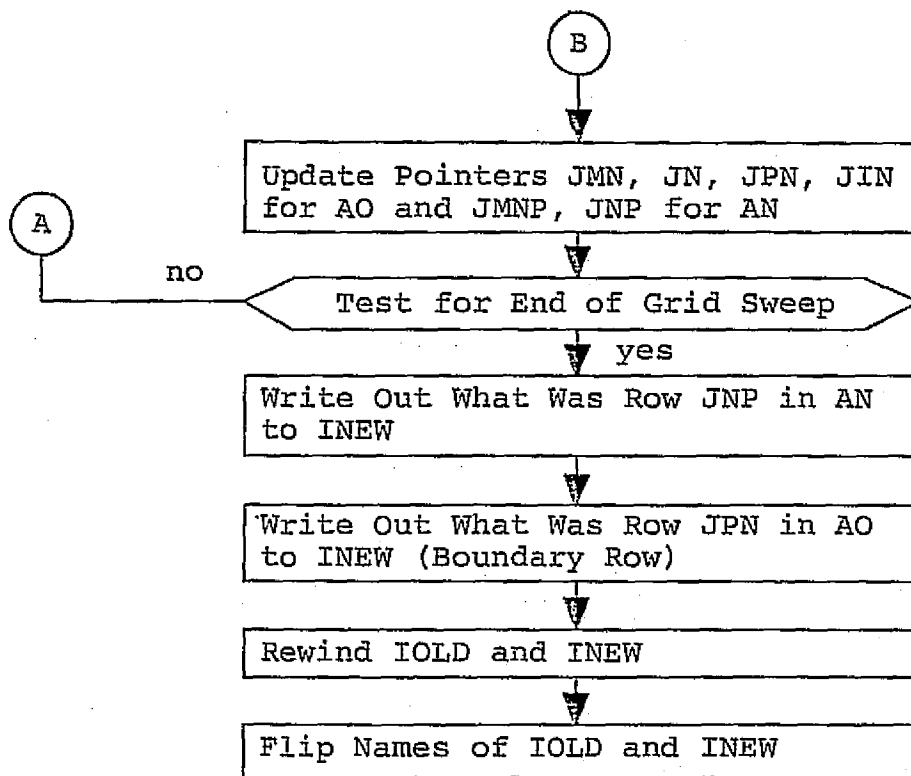


FIGURE III-3: (Continued)

TABLE III-1: OFFSETS OF VARIABLES IN ARRAYS
AO AND AN FOR PECHCV

<u>DESCRIPTION</u>	<u>NAME</u>	<u>LEVELS</u>	<u>NO.</u>
Initial terrain pressure, π_o	PTI	1	0
π reduction to sea level	PIR	1	63
Terrain geopotential	GZ	1	126
Albedo and land-sea-ice indicator (packed)	ALSI	1	189
Sea surface temperature	SST	1	252
Tendency truncator	RC	1	315
Map factor, m	MF	1	378
Sine of latitude, $\sin\phi$	SL	1	441
Geopotential, ϕ	PHI	5	504
Vertical velocity, w	WV	4	819
Precipitation	PRE	1	1071
Sea surface vapor pressure	QSS	1	1134
Saturation vapor pressure	QS	3	1197
Heating rates	HT	5	1386
X direction pressure gradient force, n and n-1 time levels	PXN	5	1701
Y direction pressure gradient force, n and n-1 time levels	PYN	5	2016
U wind, n and n-1 time levels	UN	5	2331
V wind, n and n-1 time levels	VN	5	2646

TABLE III-1: (Continued)

<u>DESCRIPTION</u>	<u>NAME</u>	<u>LEVELS</u>	<u>NO.</u>
Temperature, n and n-1 time levels	TN	5	2961
Moisture, n and n-1 time levels	QN	3	3276
Terrain pressure, π , n+1 and n time levels	PTNP	1	3465
Terrain pressure, π , n and n-1 time levels	PTN	1	3528
	NVAR =		3591

TABLE III-2: OFFSETS OF VARIABLES IN ARRAYS
AO AND AN FOR PECHFV

<u>DESCRIPTION</u>	<u>NAME</u>	<u>LEVELS</u>	<u>NO.</u>
Initial terrain pressure, π_0 , and π reduction to sea level (packed)	PTIR	1	0
Map factor, m	MF	1	63
Geopotential, ϕ	PHI	10	126
U wind, n and n-1 time levels	UN	10	756
V wind, n and n-1 time levels	VN	10	1386
Temperature, n and n-1 time levels	TN	10	2016
Moisture, n and n-1 time levels	QN	6	2646
Terrain pressure, π , n+1 and n time levels	PTNP	1	3024
Terrain pressure, π , n and n-1 time levels	PTN	1	3087
Terrain geopotential	GZ	1	3150
Albedo and land-sea-ice indicator (packed)	ALSI	1	3213
Sea surface temperature	SST	1	3276
Sea surface vapor pressure	QSS	1	3339
Tendency truncator	RC	1	3402
Sine of latitude, $\sin\phi$	SL	1	3465
Precipitation	PRE	1	3528

TABLE III-2: (Continued)

<u>DESCRIPTION</u>	<u>NAME</u>	<u>LEVELS</u>	<u>NO.</u>
Heating rates			3591
Vertical velocities, w	HT	3	3780
X direction pressure gradient force, n and n-1 time levels	WV	9	4347
Y direction pressure gradient force, n and n-1 time levels	PXN	10	4977
	PYN	10	5607
	NVAR =		

NOTE: IMAX = 63

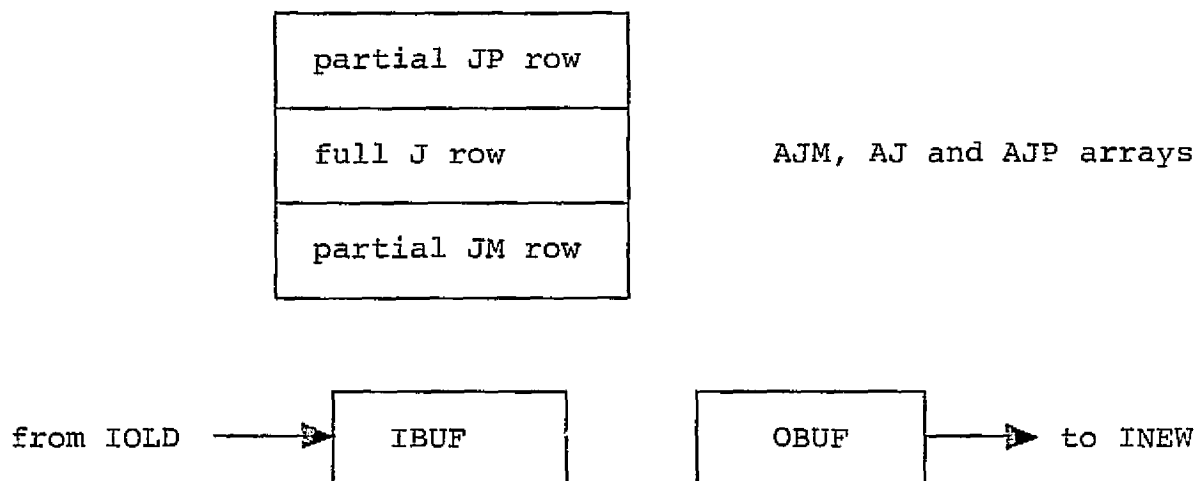
b. Data Storage - PEFHCV and PEFHFV

The magnitude of the data file that must be processed for the fine horizontal resolution forecast models is considerably larger than for the coarse horizontal resolution models. This rather severely restricts the type of data flow scheme that can be constructed, since the problem becomes one of fitting the computation into the machine (a CYBER 76 was used as the target machine).

In order to fit the computation in the machine, a split row was devised. A split row contains all of the variables for a vertical slice, but they are grouped into two segments. The lower portion of the split row contains the variables that are required on the three rows J-1, J and J+1 during computation on row J. The upper portion of the split row contains variables that are only needed on row J. This allows smaller central memory buffers since there need be only one full row and two partial rows in central memory simultaneously.

A diagram of the necessary central memory and ECS or LCM storage buffers for this scheme to work for the model PEFHCV is given in Figure III-4. There are three rows in central memory, a partial J-1 row, AJM, a full J row, AJ, and a partial J+1 row, AJP. There are also two buffer arrays IBUF and OBUF in central memory used for buffering data from and to files IOLD and INEW, respectively. For ECS or LCM

Central Memory



Extended Memory or LCM

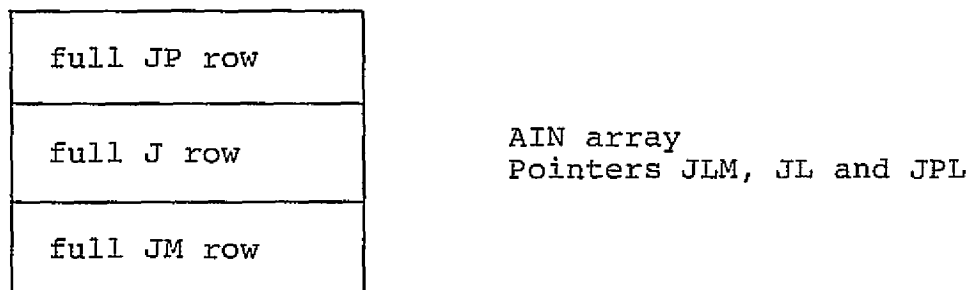


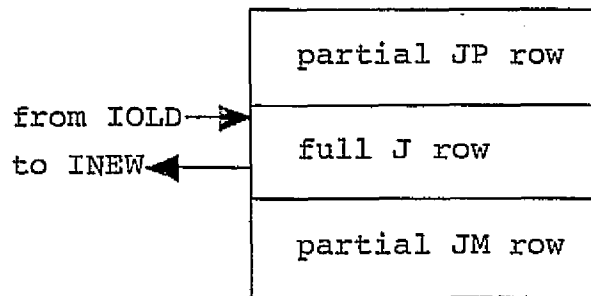
FIGURE III-4: STORAGE BUFFERS REQUIRED
FOR ROW STORAGE SCHEME
IN MODEL PEFHCV

there is a buffer array, AIN, for three full rows. This buffer array is a circular buffer and has pointers JLM, JL and JLP pointing to rows J-1, J and J+1, respectively.

Similarly a diagram of the necessary central memory and ECS or LCM storage buffers for this scheme to work for the model PEFHFV is given in Figure III-5. There are three rows in central memory, a partial J-1 row, AJM, a full J row, AJ, and a partial J+1 row, AJP. IN ECS or LCM there is a buffer array, AIN, for three full rows. This buffer array is a circular buffer and has pointers: JLM, JL and JLP pointing to rows J-1, J and J+1, respectively. Due to the way that the CDC system software works (buffer transfers into LCM are not allowed), access to the files IOLD and INEW is made through the central memory array AJ with a block transfer either to or from ECS or LCM also being required.

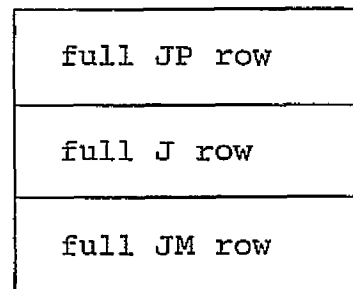
One more construct is needed before a given piece of data can be accessed in the arrays AJM, AJ and AJP. This is a method of pointing to the particular variable that is required. The offset of the variable can be used to point to single vertical level data since there are no row pointers to the central memory arrays. However, for computational convenience, a subscript function has been defined for data that exists on more than one vertical level, or K level. This subscript function, which takes the form

Central Memory



AJM, AJ and AJP arrays

Extended Memory or LCM



AIN array
Pointers JLM, JL and JLP

FIGURE III-5: STORAGE BUFFERS REQUIRED
FOR ROW STORAGE SCHEME
IN MODEL PEFHFV

of a FORTRAN statement function, is defined below:

$$\text{ISUBK}(\text{IOFF}, \text{K}) = \text{IOFF} + (\text{K} - 1) * \text{IMAX} \quad [\text{III.9}]$$

where

IOFF = Offset of the particular variable desired

K = Vertical level desired

IMAX = Total length of a row (range on I).

For single level data the offset of the variable points to the $I = 0$ element, and for multiple vertical level data the subscript function ISUBK points to the $I = 0$ element for level K. An example of the use of the subscript function was given in the previous section.

The mechanics for accessing data in the central memory arrays AJM, AJ and AJP have now been constructed. The data access mechanism together with the buffer structures shown in Figures III-4 and III-5 allow the computation to sweep the data base on file IOLD while constructing an updated data base on file INEW. Figures III-6 and III-7 combine these ideas and present the data flow for one complete sweep of the data file in the forecast models PEFHCV and PEFHFV, respectively. The data flow shown assumes that free slip, insulated wall boundary conditions are used.

Finally, Table III-3 defines the offsets which are used by the two forecast models to access data in the buffer arrays AJM, AJ and AJP.

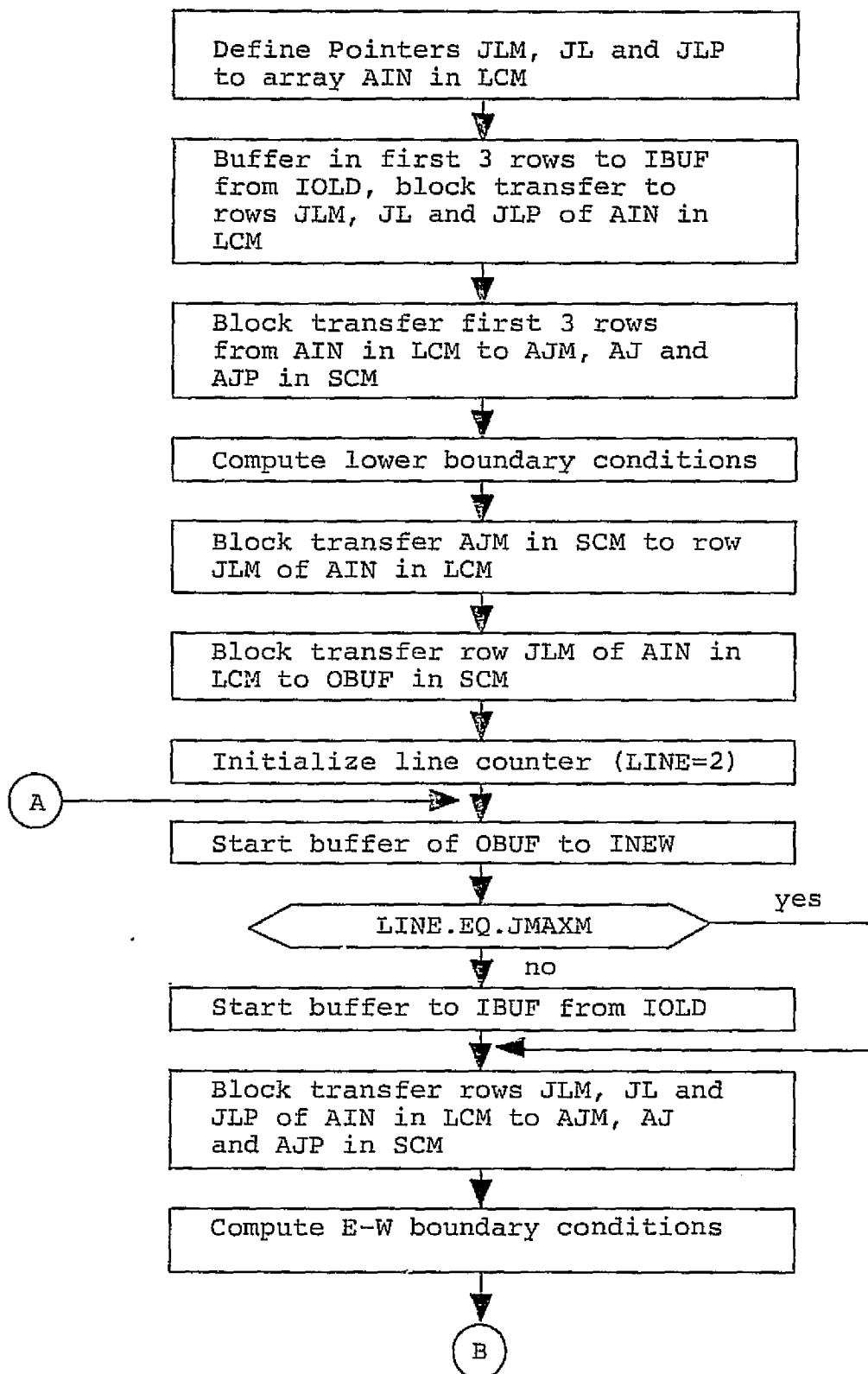


FIGURE III-6: DATA FLOW FOR ONE SWEEP OF THE
DATA FILE - MODEL PEFHCV

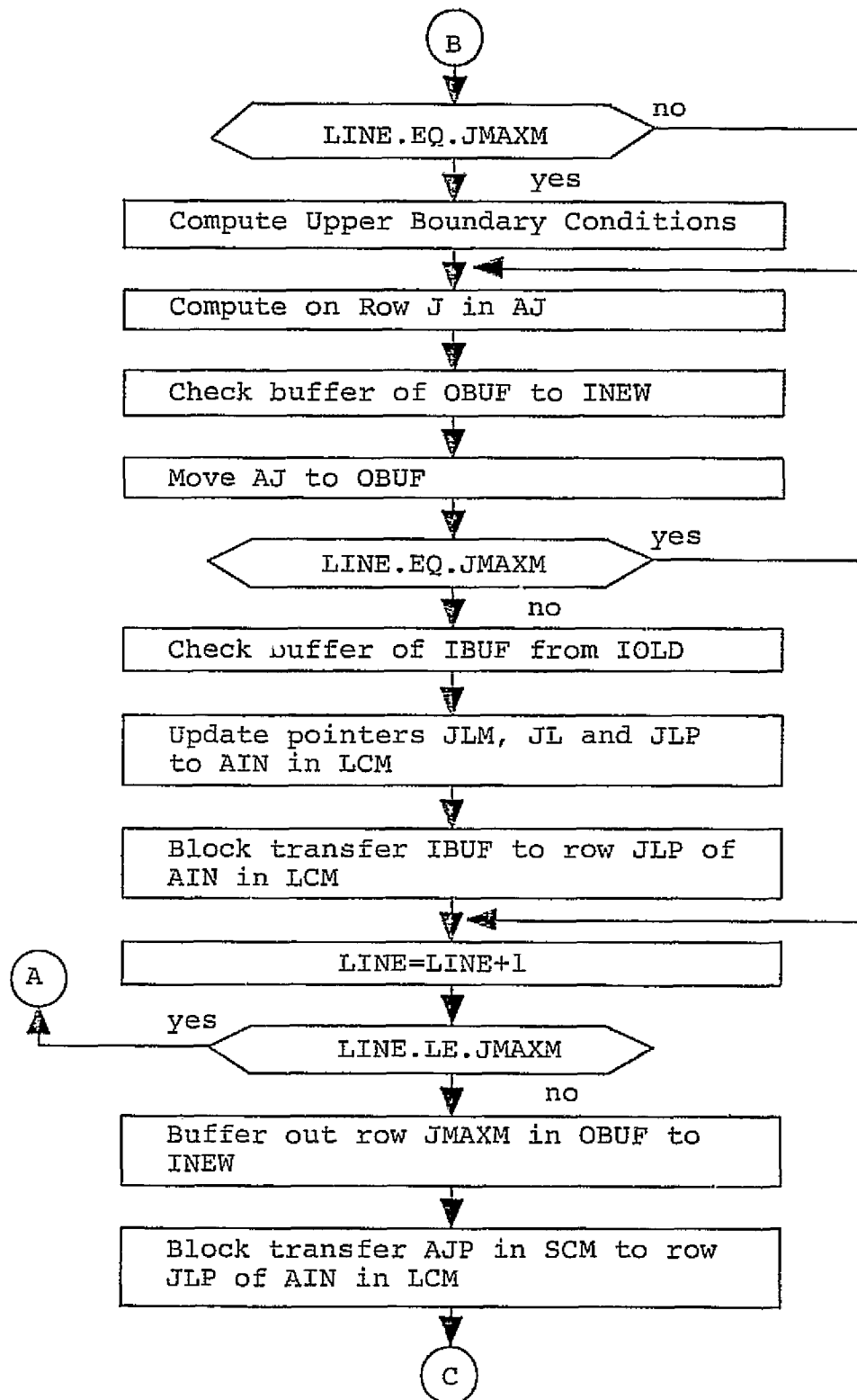


FIGURE III-6: (Continued)

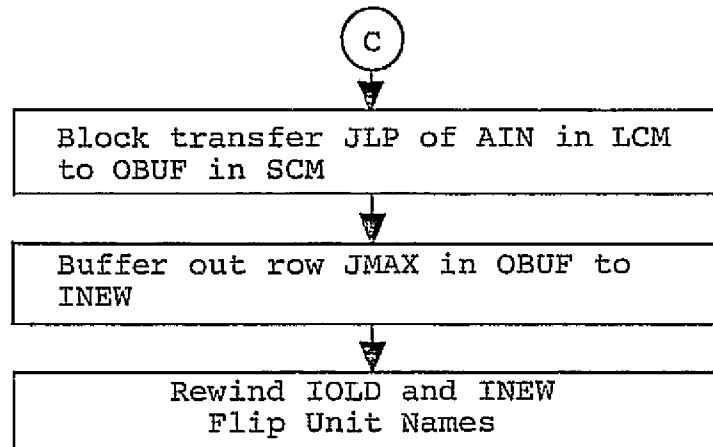


FIGURE III-6: (Continued)

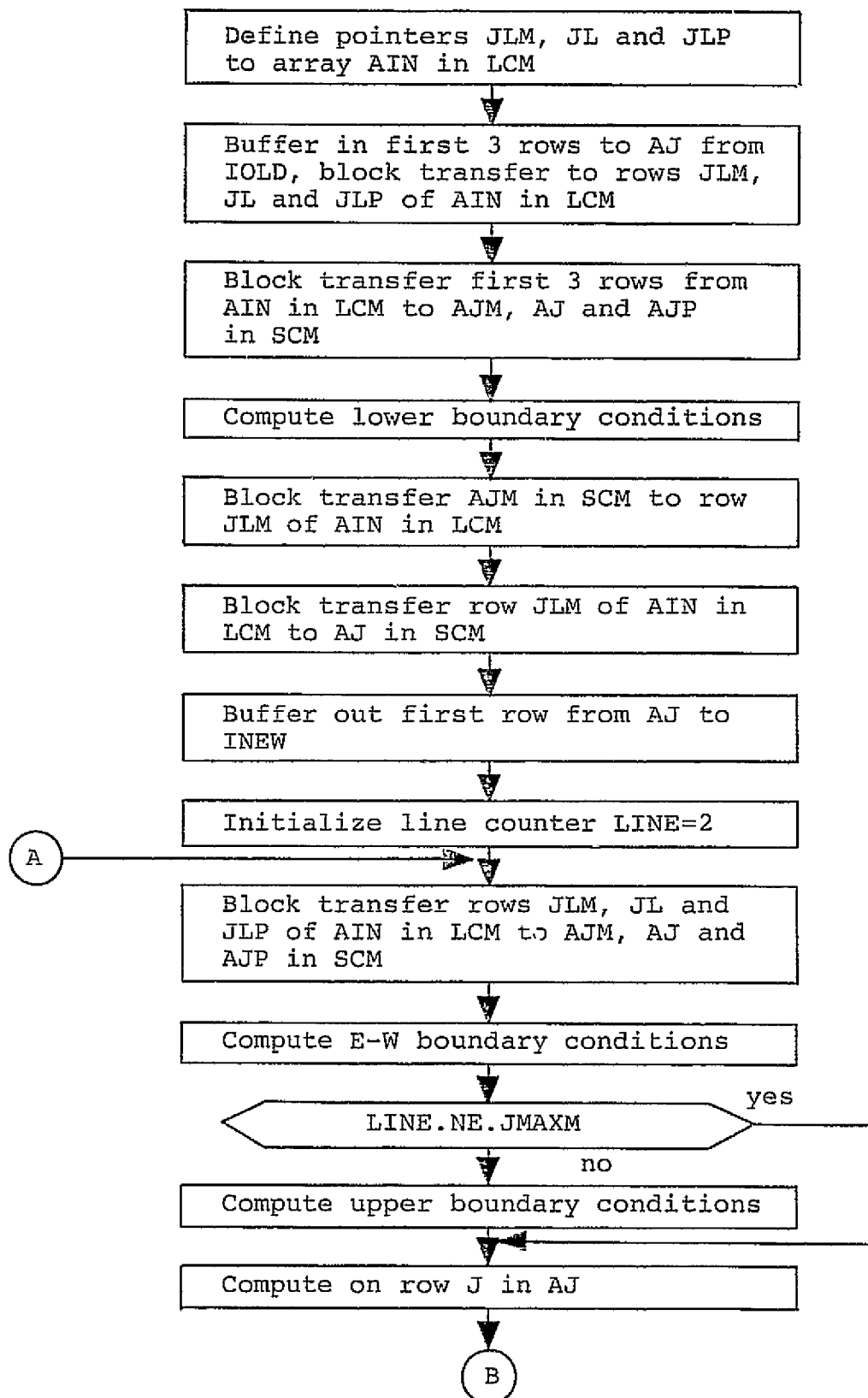


FIGURE III-7: DATA FLOW FOR ONE SWEEP OF THE DATA FILE - MODEL PEFHFV

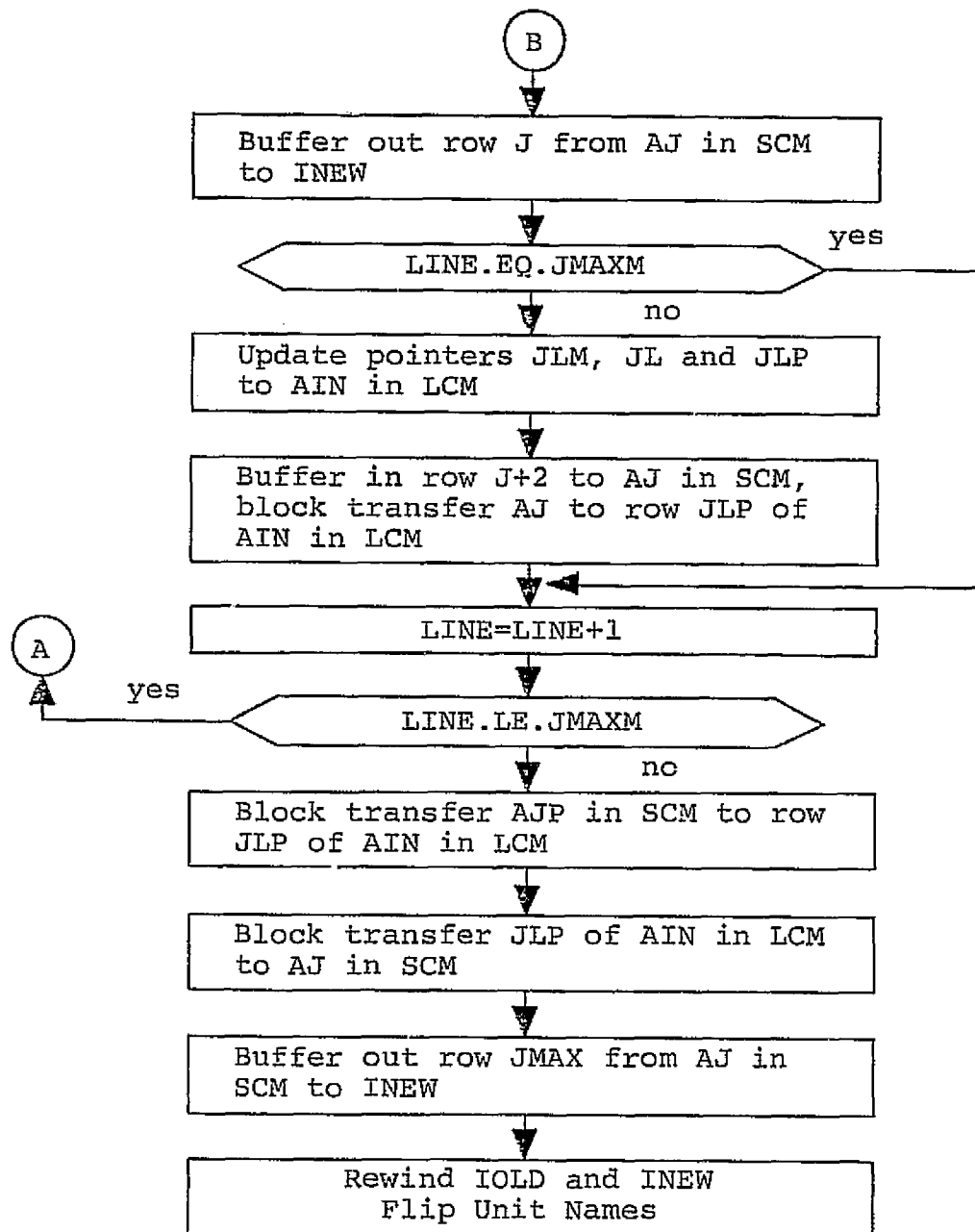


FIGURE III-7: (Continued)

TABLE III-3: OFFSETS OF VARIABLES IN ARRAYS
AJM, AJ AND AJP FOR PEFHCV AND PEFHVV

<u>DESCRIPTION</u>	<u>NAME</u>	<u>5 LAYER</u>		<u>10 LAYER</u>	
		<u>Level</u>	<u>No.</u>	<u>Level</u>	<u>No.</u>
Initial terrain pressure, π_0 , and π reduction to sea level (packed)	PTIR	1	0	1	0
Map factor, m	MF	1	187	1	187
Geopotential heights, ϕ	PHI	5	374	10	374
U wind, n and n-1 time levels	UN	5	1309	10	2244
V wind, n and n-1 time levels	VN	5	2244	10	4114
Temperature, n and n-1 time levels	TN	5	3179	10	5984
Moisture, n and n-1 time levels	QN	3	4114	6	7854
Terrain pressure, π , n+1 and n time levels	PTNP	1	4675	1	8976
Terrain pressure, π , n and n-1 time levels	PTN	1	4862	1	9163
Terrain geopotential	GZ	1	5049	1	9350
Albedo and land-sea-ice indicator (packed)	ALSI	1	5236	1	9537
Sea surface temperature and sea surface vapor pressure (packed)	SSTQSS	1	5423	1	9724
Tendency truncator	RC	1	5610	1	9911
Sine of latitude, $\sin\phi$	SL	1	5797	1	10,098
Precipitation	PRE	1	5984	1	10,285

TABLE III-3: OFFSETS OF VARIABLES (Continued)

<u>DESCRIPTION</u>	<u>NAME</u>	<u>5 LAYER</u>		<u>10 LAYER</u>	
		<u>Level No.</u>		<u>Layer No.</u>	
Heating Rates	HT	3	6171	3	10,472
Vertical velocities, w	W	4	6732	9	11,033
X direction pressure gradient force, n and n-1 time levels	PXN	5	7480	10	12,716
Y direction pressure gradient force, n and n-1 time levels	PYN	5	8415	10	14,586
	NVARS =		5049		9,350
	NVARL =		9350		16,456

NOTE: IMAX = 187

B. Code Structure - PECHCV and PECHFV

This section describes the program structure of the forecast models PECHCV and PECHFV. It also defines the various files used by the programs, defines the common blocks that are used and describes each of the subprograms that constitute the models.

The codes are written in CDC FORTRAN, which is a dialect of FORTRAN IV, for the CDC FORTRAN Extended Compiler (OPT=2 level). The two programs, PECHCV and PECHFV, have been designed to use rotating mass storage communicating with central memory through two I/O channels.

The forecast models are fairly large codes. Since the various sections of the codes -- initialization, integration and output -- are each fairly large, the codes have been overlaid to more efficiently use central memory. The sections that follow describe the main divisions of the forecast models by overlays. Section 1 describes the control overlay, gives a block diagram of the overlay structure, defines the I/O file structure and defines the common blocks that are global to all overlays.

Section 2 describes the initialization overlay, defines the common blocks local to this overlay, and describes each of the subprograms that are used by this overlay. Similarly, Sections 3 and 4 describe the integration and output overlays, respectively.

1. Main Overlay - Programs PECHCV and PECHFV.

The main overlay of the forecast model programs is very simple, since it only performs the function of declaring three global common blocks, defining various control variables and calling the actual working overlays. The overlay structure of the programs is shown in Figure III-8. Only one level of overlay is used for programs PECHCV and PECHFV.

The various files used by the programs are defined in Table III-4. Tables III-5 and III-6 define the variables that appear in the global common blocks CNTRL, and INDEX, respectively. Common block BUFPOINT consists of the offsets for the variables and contains the integer variables listed in Tables III-1 and III-2 for PECHCV and PECHFV, respectively.

a. Subroutine BUFERR

Subroutine BUFERR prints a diagnostic message when an error occurs on a BUFFER (IN or OUT) operation that tells where the error occurred.

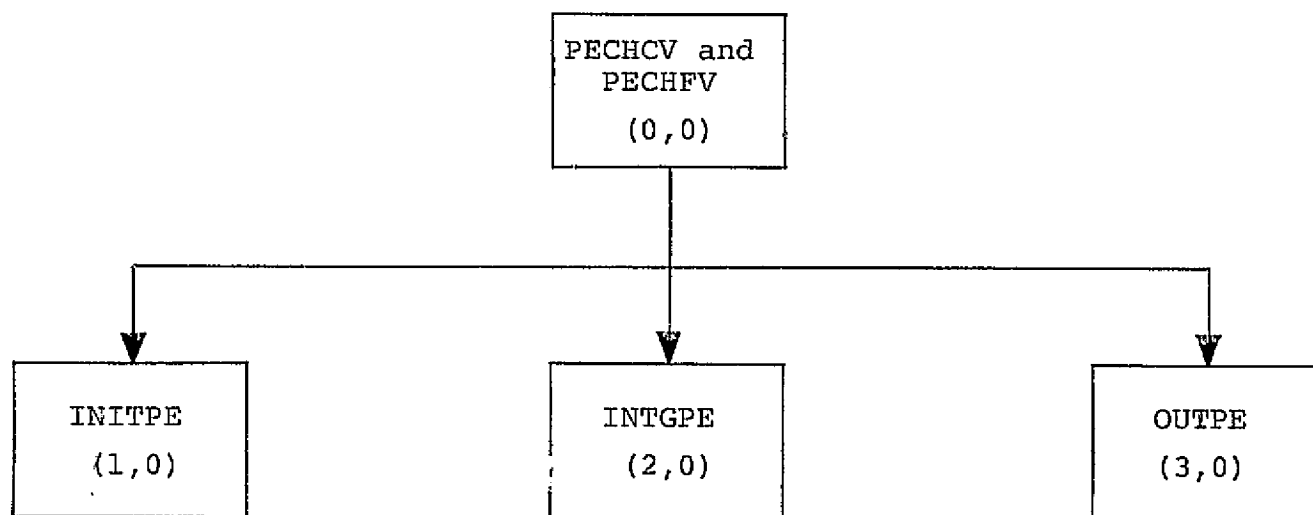


FIGURE III-8: OVERLAY STRUCTURE OF FORECAST MODELS
PECHCV AND PECHFV

TABLE III-4: FILE STRUCTURE OF FORECAST
MODELS PECHCV AND PECHFV

OUTPUT	Used to write various diagnostic messages
TAPE1	Input data file for initialization overlay; generated by analysis programs
TAPE2	Random access file used by initialization and output overlays
TAPE3	Restart data tape; contains a header record and contents of current IOLD file
TAPE4	Output data file; contains forecast fields to be selectively plotted by graphics program and for use by the analysis programs
TAPE8	(IOLD, INEW) integration data
TAPE9	(IOLD, INEW) integration data

TABLE III-5: DEFINITION OF VARIABLES IN COMMON BLOCK
/CNTRL/ FOR PECHCV AND PECHFV

IOLD	File name of integration data (TAPE8 or TAPE9)
INEW	File name of integration data (TAPE8 or TAPE9)
ISW	=1, SSW1 is "on", signifying normal start with data from TAPE1 =2, SSW1 is "off", signifying restart with data from TAPE3
ID(2)	1st word, forecast identifier (date-time group) 2nd word, forecast data field identifier
LIMI	Starting cycle
LIM	Number of integration and output cycles (2 x number of days + 1)
ITAU	Forecast time indicator (hours) = 0 during initialization and first output cycle = 12*LCO for normal integration cycle
ITSPHR	Number of integration time steps per hour
NN	Time step counter, reset after each output cycle, also NN=0 signifies a Matsuno time step
NNOUT	Number of time steps between normal output cycles, usually equal to 12 x ITSPHR
LCO	Output cycle counter, updated by 1 each output cycle
L12	Indicator set at 12-hour output cycle, used to zero rainfall
DAY	Number of days from January 1 of the forecast
HR	Hour of the day, GMT
DT	Time step, Δt
TDT	Twice the time step, $2\Delta t$
MATS	Logical variable used during Matsuno time step = .FALSE. during first half = .TRUE. during second half

TABLE III-6: DEFINITION OF VARIABLES IN COMMON BLOCK
/INDEX/ FOR PECHCV AND PECHFV

ICL	Start of computation on data row (=2)
ICU	End of computation on data row (=IMAX-1)
IMAXM	Length of data row -1
IMAX	Length of data row
JMAXM	Number of data rows -1
JMAX	Number of data rows
KMAXQ	Number of moisture levels
KMAXM	Number of vertical levels -1
KMAX	Number of vertical levels
LINE	Number of row being worked on
NFLD	Size of a data record on TAPE2 (63 x 63)
NVAR	Size of a data record on TAPE8 and TAPE9

2. Initialization Overlay - Program INITPE

The function of the initialization overlay is to take as input data the fields generated by the analysis programs and to generate an initial state for the integration overlay which actually performs the forecast. That is, the interpolation and synthesis of data fields to the PEM sigma surfaces using data produced by the analysis programs as described in Section II-5 are performed by the initialization overlay.

Program INITPE extracts data generated by the analysis programs from the sequential file TAPE1. Using these data, it interpolates and synthesizes the data fields on the sigma surfaces and writes these new fields to the random file TAPE2. They are then sorted into vertical slice format on file IOLD for use by the integration overlay.

Figure III-9 shows the structure of the initialization overlay, giving the names of all the subprograms which constitute this overlay. The following sections describe these routines.

The labeled common block, INCON, is local to this overlay and its variables are described in Table III-7.

Table III-8 defines the data fields on TAPE1 which are produced by the analysis programs. Table III-9 defines the fields on TAPE2 which are produced both by the initialization process to define the initial state of the forecast and by the output overlay to present the results of the forecast.

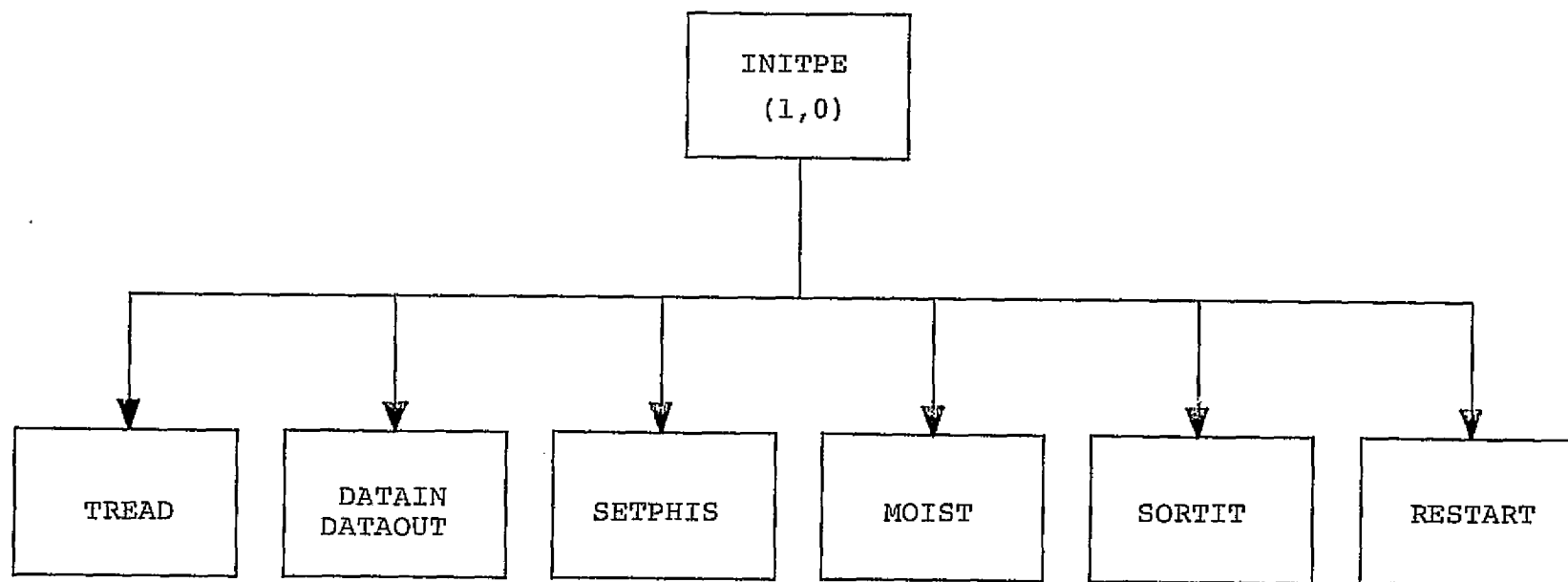


FIGURE III-9: INITIALIZATION OVERLAY, PROGRAMS
PECHCV AND PECHFV

TABLE III-7: DEFINITION OF VARIABLES IN COMMON BLOCK
/INCON/ FOR PECHCV AND PECHFV

PLVL(13)	Pressure levels of input analysis data: 1000, 950, 900, 850, 700, 500, 400, 300, 250, 200, 150, 100, 50
PLVLN(13)	$\ln [PLVL(K)]$, $K = 1, KMAX$
SIG(5)or(10)	Sigma levels 0.9, 0.7, 0.5, 0.3, 0.1 for PECHCV and .95, .85, .75, .65, .55, .45, .35, .25, .15, .05 for PECHFV
ZLN(5)or(10)	$\ln [SIG(K)]$, $K = 1, KMAX$
SLN(5)or(10)	$R \cdot \ln(0.9)$ for $K=1$, $R \cdot \ln[SIG(K)/SIG(K-1)]$; $K=2, KMAX$
G	Gravitational acceleration = 9.80616 m/sec^2
GRDC	Constant used in calculation of $\sin(\text{latitude}) =$ $(31.205)^2$
CTOK	Conversion constant relating degrees Celsius to degrees Kelvin = 273.16
AE	Constant used in calculation of saturation vapor pressure = 21.656
BE	Constant used in calculation of saturation vapor pressure = 5418

TABLE III-8: CONTENTS OF THE INPUT
DATA FILE - TAPE1
CREATED BY THE ANALYSIS PROGRAMS

<u>Record</u>	<u>Data Field</u>
1	Geopotential heights of the 1,000 mb surface
2	" " 950 "
3	" " 900 "
4	" " 850 "
5	" " 700 "
6	" " 500 "
7	" " 400 "
8	" " 300 "
9	" " 250 "
10	" " 200 "
11	" " 150 "
12	" " 100 "
13	" " 50 "
14	Temperatures on the 1,000 mb surface
15	" " 950 "
16	" " 900 "
17	" " 850 "
18	" " 700 "
19	" " 500 "
20	" " 400 "
21	" " 300 "
22	" " 250 "
23	" " 200 "
24	" " 150 "
25	" " 100 "
26	" " 50 "
27	Land-sea-ice indicator
28	Surface pressure, P_s

TABLE III-8: INPUT DATA FILE - TAPE1 (Continued)

<u>Record</u>	<u>Data Field</u>
29	Terrain height
30	Sea surface temperature
31	Albedo
32	U wind on the 1,000 mb surface
33	V " 1,000 "
34	U " 950 "
35	V " 950 "
36	U " 900 "
37	V " 900 "
38	U " 850 "
39	V " 850 "
40	U " 700 "
41	V " 700 "
42	U " 500 "
43	V " 500 "
44	U " 400 "
45	V " 400 "
46	U " 300 "
47	V " 300 "
48	U " 250 "
49	V " 250 "
50	U " 200 "
51	V " 200 "
52	U " 150 "
53	V " 150 "
54	U " 100 "
55	V " 100 "

NOTE: Geopotential heights are in meters;
temperatures are in degrees centigrade;
surface pressure is in mb;
terrain height is in meters;
winds are in meters per second.
All fields are 63 x 63 arrays with a
20 word ID prefixed.

TABLE III-9: CONTENTS OF THE RANDOM ACCESS FILE -
TAPE2 FOR PECHCV AND PECHFV 10, SIGMA
LEVEL VERSION OF THE FORECAST MODEL

<u>Record</u>	<u>Data Field</u>
1	Analyzed geopotential heights of 1000 mb surface (meters)
2	" 950
3	" 900
4	" 850
5	" 700
6	" 500
7	" 400
8	" 300
9	" 250
10	" 200
11	" 150
12	" 100
13	" 50
14	
15	
16	Analyzed temperatures of 1000 mb surface (°C)
17	" 950
18	" 900
19	" 850
20	" 700
21	" 500
22	" 400
23	" 300
24	" 250
25	" 200
26	" 150
27	" 100
28	" 50
29	
30	
31	Analyzed surface pressure, Ps (mb)
32	Terrain height, Z_T (meters)
33	Sea surface temperature (°C)
34	Land-sea-ice indicator
35	Sin of latitude, $\sin\phi$
36	Map factor, m
37	Tendency truncator
38	Albedo
39	Terrain geopotential, gZ_T
40	

TABLE III-9: RANDOM ACCESS FILE - TAPE2
(Continued)

<u>Record</u>	<u>Data Field</u>
41	Temperature on $\sigma = 0.95$ surface ($^{\circ}\text{K}$)
42	" 0.85 "
43	" 0.75 "
44	" 0.65 "
45	" 0.55 "
46	" 0.45 "
47	" 0.35 "
48	" 0.25 "
49	" 0.15 "
50	" 0.05 "
51	Vapor pressures on $\sigma = 0.95$ surface (mb)
52	" 0.85 "
53	" 0.75 "
54	" 0.65 "
55	" 0.55 "
56	" 0.45 "
57	Terrain pressure, π (mb)
58	Pressure difference, $P_s - \pi$ (mb)
59	$\ln(\pi)$
60	Forecast P_s (mb)
61	Geopotentials of $\sigma = 0.95$ surface (g.meters)
62	" 0.85 "
63	" 0.75 "
64	" 0.65 "
65	" 0.55 "
66	" 0.45 "
67	" 0.35 "
68	" 0.25 "
69	" 0.15 "
70	" 0.05 "
71	U winds on 950 mb surface (m/sec)
72	" 850
73	" 700
74	" 500
75	" 400
76	" 300
77	" 250
78	" 200
79	" 150
80	" 100

TABLE III-9: RANDOM ACCESS FILE - TAPE2
(Continued)

<u>Record</u>	<u>Data Field</u>			
81	V winds on 950 mb surface (m/sec)			
82	"	850		
83	"	700		
84	"	500		
85	"	400		
86	"	300		
87	"	250		
88	"	200		
89	"	150		
90	"	100		
91	U winds on $\sigma = 0.95$ surface (m/sec)			
92	"	0.85	"	
93	"	0.75	"	
94	"	0.65	"	
95	"	0.55	"	
96	"	0.45	"	
97	"	0.35	"	
98	"	0.25	"	
99	"	0.15	"	
100	"	0.05	"	
101	V winds on $\sigma = 0.95$ surface (m/sec)			
102	"	0.85	"	
103	"	0.75	"	
104	"	0.65	"	
105	"	0.55	"	
106	"	0.45	"	
107	"	0.35	"	
108	"	0.25	"	
109	"	0.15	"	
110	"	0.05	"	
111	Forecast geopotential height of 1,000 mb surface (meters)			
112	"	"	950	"
113	"	"	900	"
114	"	"	850	"
115	"	"	700	"
116	"	"	500	"
117	"	"	400	"
118	"	"	300	"
119	"	"	250	"
120	"	"	200	"
121	"	"	150	"
122	"	"	100	"

TABLE III-9: RANDOM ACCESS FILE - TAPE2
(Continued)

<u>Record</u>	<u>Data Field</u>
123	Forecast temperatures on 1,000 mb surface (°C)
124	" 950 "
125	" 900 "
126	" 850 "
127	" 700 "
128	" 500 "
129	" 400 "
130	" 300 "
131	" 250 "
132	" 200 "
133	" 150 "
134	" 100 "
135	Forecast U winds on 1,000 mb surface (m/sec)
136	" 950 "
137	" 900 "
138	" 850 "
139	" 700 "
140	" 500 "
141	" 400 "
142	" 300 "
143	" 250 "
144	" 200 "
145	" 150 "
146	" 100 "
147	Forecast V winds on 1,000 mb surface (m/sec)
148	" 950 "
149	" 900 "
150	" 850 "
151	" 700 "
152	" 500 "
153	" 400 "
154	" 300 "
155	" 250 "
156	" 200 "
157	" 150 "
158	" 100 "
159	Sea surface saturation vapor pressure (mb)
160	Forecast precipitation (cm)
161	Vapor pressure on 1,000 mb surface (mb)
162	" 900 "
163	" 700 "
164	" 500 "
165	" 300 "

a. Subroutine SETPHIS

Subroutine SETPHIS computes the initial geopotentials of the sigma surfaces. Input to the routine consists of the temperatures on the sigma surfaces and the moistures on the lower sigma surfaces. The geopotentials are computed using the form of the hydrostatic equation given by Equations [II.107] and [II.108] given in Section II-D-1. Finally, the computed geopotentials are stored on the random access file TAPE2 for use in the forecast.

b. Subroutine MOIST

Subroutine MOIST computes the relative humidity on a pressure surface using the procedure based on the relative vorticity as described in Section II-E-6. Input to the routine consists of the geopotential heights of the pressure surface. The routine then computes the relative vorticity on the pressure surface and, using the empirical relation given by Equation [II.133], computes the relative humidity on the pressure surface.

Program INITPE calls this subroutine five times to generate moistures on the 1,000, 900, 700, 500 and 300 mb surfaces. Data on these pressure surfaces is then used to obtain initial moisture data on the lower sigma surfaces through logarithmic pressure interpolation.

These data fields are stored on the random access file TAPE2 for use in the forecast.

c. Subroutine WIND

Subroutine WIND computes the geostrophic wind components at 50 mb for use in the initialization of the winds on the model sigma surfaces. Input to the routine consists of the geopotential heights of the 50 mb pressure surface.

d. Subroutine TREAD

Subroutine TREAD acts as a generalized input data routine to obtain analysis data from TAPE1. A storage array address and the data field ID are input to TREAD from the calling routine. TREAD then searches a table (array ITP) to find the location of the desired data field and reads the data field from TAPE1 into the storage array. Various data field orderings on TAPE1 may be accommodated simply by changing the information stored in array ITP.

e. Subroutine DATAIO

Subroutine DATAIO is a generalized routine for writing and reading data fields to and from the random access file TAPE2. Two entry points exist for this routine: DATAOUT, which uses the system routine WRITMS for writing to the file, and DATAIN, which uses the system routine READMS for reading from the file.

f. Subroutine RSTART

Subroutine RSTART reads the restart data tape, TAPE3, when a forecast run is being restarted and initializes the forecast model in order to continue the forecast. Restarts are controlled by sense switch 1. If SSW1 is "ON", then a normal run is being made; if SSW1 is "OFF", then a restart run is being made.

A restart data tape is written during each output cycle so if the computer goes down or if for some reason a forecast is to be continued at a later time, the forecast can be resumed from the point at which the last restart tape was written. Table III-10 gives the format of the restart data tape TAPE3.

Thus subroutine RSTART reads in the header record from TAPE3 and initializes the required control variables. Next the routine reads in the forecast data records and recreates file IOLD. The forecast run will now proceed from this point in the forecast.

g. Subroutine SORTIT

Since the output of the analysis programs and the initialization routines are data on horizontal planes and the integration overlay performs computations on data arranged in vertical slices, an interface routine to properly structure the data is required. Subroutine SORTIT

TABLE III-10: RESTART DATA TAPE (TAPE3) FORMAT

- 1) Header record (twenty words) -
contains information to identify the forecast

1	ID(1)
2	ITAU
3	IOUT
4	LCO
5	L12
6-10	Unused
11	DAY
12	HR
13-20	Unused

- 2) Forecast data records -
these contain the contents of file IOLD at ITAU
(JMAX records in vertical slice format)

takes the initial data in horizontal plane form which is stored on the random access file TAPE2 and constructs the forecast data file IOLD in vertical slice form for use by the integration overlay.

Due to core storage limitations, more than one pass must be made through the file IOLD to completely initialize the forecast. Each pass consists of reading a maximum of ten planes of data from TAPE2 and then sweeping file IOLD from row one to row JMAX, writing out the initial data.

3. Integration Overlay - Program INTGPE

The function of the integration overlay is to numerically integrate the primitive equations; that is, to step the data fields forward in time to produce a forecast. Input to this overlay consists initially of the data field specification produced by the initialization overlay and, subsequently, the current state of the forecast contained on file IOLD. After initialization, a forecast consists of cycling through the integration and output overlays until the desired forecast time has been reached.

Figure III-10 shows the structure of the integration overlay, giving the names of all the subprograms which constitute this overlay. The function of each of these routines is described in the sections that follow.

Table III-11 defines the variables that appear in the common blocks that are local to the integration overlay for programs PECHCV and PECHFV.

a. Subroutine DEFINE

Subroutine DEFINE has the simple function of defining various constants that are used in the integration overlay and zeroing the buffer arrays of this overlay. The subroutine is called once per integration-output cycle upon entering the integration overlay.

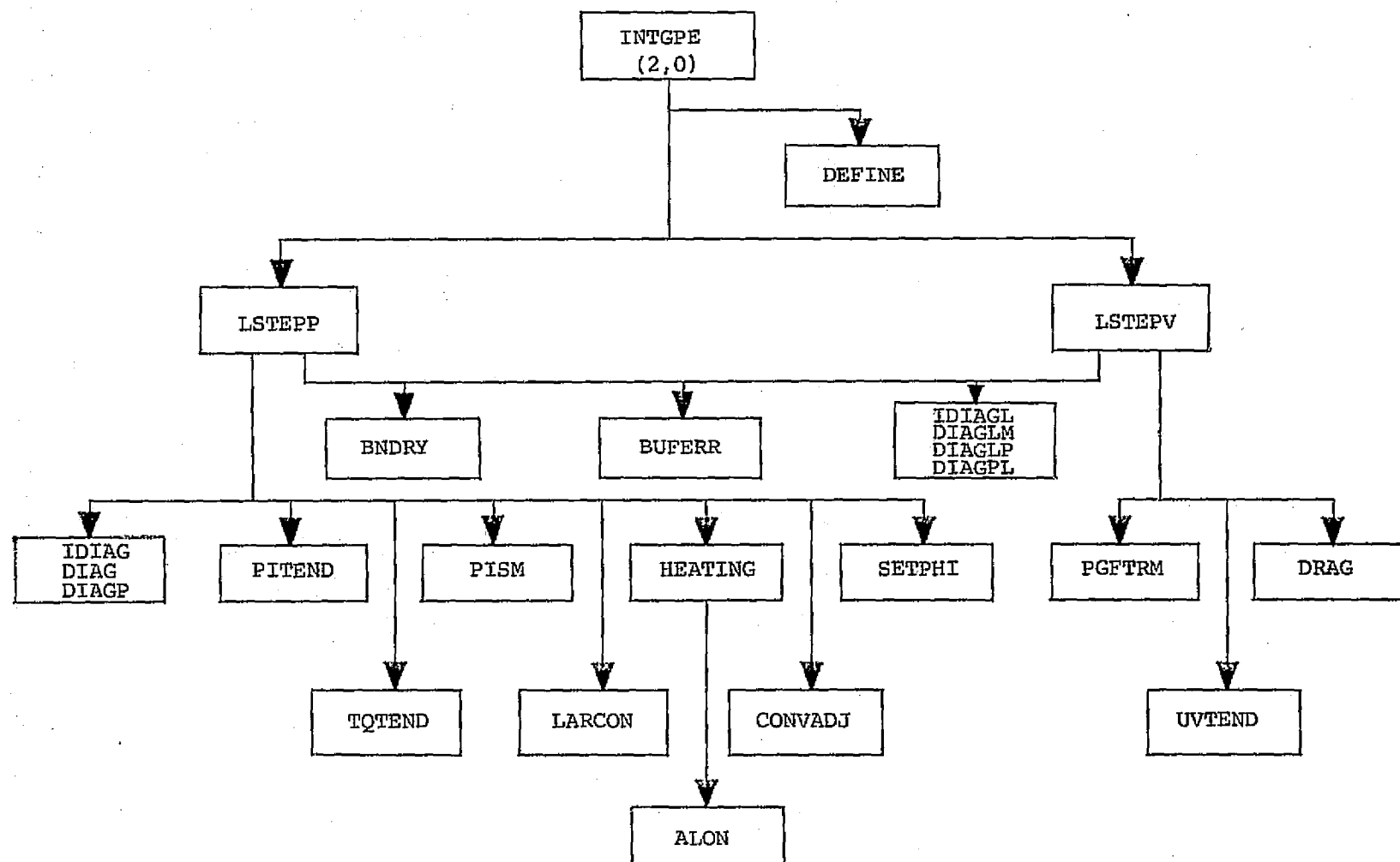


FIGURE III-10: INTEGRATION OVERLAY, PROGRAM INTGPE

TABLE III-11: DEFINITION OF VARIABLES IN
COMMON BLOCKS LOCAL TO THE
PROGRAMS PECHCV AND PECHFV

Common Block /BUFF/

AO(NVAR,4)	Circular input buffer for rows JMN, JN, JPN and JIN
AN(NVAR,2)	Circular output buffer for rows JMNP and JNP

Common Block /TEND/

UT(63,KMAX)	$\frac{\partial \pi u}{\partial t}$, tendencies in πu for all levels on row JN
VT(63,KMAX)	$\frac{\partial \pi v}{\partial t}$, tendencies in πv for all levels on row JN
TT(63,KMAX)	$\frac{\partial \pi T}{\partial t}$, tendencies in πT for all levels on row JN
QT(63,KMAXQ)	$\frac{\partial \pi q}{\partial t}$, tendencies in πq for all levels on row JN
PIT(63)	$\frac{\partial \pi}{\partial t}$, tendency in terrain pressure, π , on row JN

Common Block /LINEAR/

DPIDX(63,KMAX)	Kurihara modification pressure gradient terms in x direction for all levels on row JN
DPIDY(63,KMAX)	Kurihara modification pressure gradient terms in y direction for all levels on row JN
W(63,KMAXM)	Vertical velocities for all levels of row JN

TABLE III-11: Definition of Variables (Continued)

Common Block /LOCAL/

UPM(63)	$\frac{u\pi}{m}$ at a particular level on row JN
VPM(63,3)	$\frac{v\pi}{m}$ at a particular level on rows JMN, JN and JPN
PILN(63,3)	$\ln(\pi)$ on rows JMN, JN and JPN
PIKAP(63)	π^K on row JN
TCOR(63,3,KMAX)	temperatures corrected for vapor pressure at all levels on rows JMN, JN and JPN
<u>NOTE:</u> TCOR is equivalenced to the following variables:	
AB(63,KMAX)	$\frac{\Delta\sigma m^2}{2\Delta x} (\delta\alpha + \delta\beta)$ at all levels on row JN
QC(63,KMAXQ)	vapor pressure, q , at all levels on row JN
QCS(63,KMAXQ)	saturation vapor pressure, q_s , at all levels on row JN
TC(63,KMAX)	temperature at all levels on row JN
PC(63,KMAXQ)	pressure at moisture levels on row JN
RF(63)	precipitation on row JN

TABLE III-11: DEFINITION OF VARIABLES (Continued)

Common Block /CON/

D	$\Delta x = \Delta y = 381000$ for coarse-mesh models and 127000 meters for fine-mesh models
DSQ	$\Delta x^2 = \Delta y^2$
TD	$2\Delta x = 2\Delta y$
OFD	$1/4\Delta x$
DSIG	$\Delta\sigma = 0.2$ for five-layer models and 0.1 for ten-layer models
TDSIG	$2\Delta\sigma$
SIG(KMAX)	Sigma levels $0.9, 0.7, 0.5, 0.3, 0.1$ for five- layer models, and $0.95, 0.85, 0.75, 0.65, 0.55,$ $0.45, 0.35, 0.25, 0.15, 0.05$ for ten-layer models
CT(KMAXM)	$\ln [\text{SIG}(K)/\text{SIG}(K-1)]$; $K = 2, KMAX$
SLN(KMAX)	$R \cdot \ln(0.9)$ or $\ln(0.95)$ for $K=1$, $R \cdot \text{CT}(K)$ for $K=2, KMAX$
S9*	$\ln(0.9)$
S7*	$\ln(0.7)$
S5*	$\ln(0.5)$
S3*	$\ln(0.3)$
S1*	$\ln(0.1)$
AE	Constant used in calculation of saturation vapor pressure = 21.656
BE	Constant used in calculation of saturation vapor pressure = 5418.0
G	Gravitational acceleration = 9.80616 m/sec^2

TABLE III-11: Definition of Variables (Continued)

R	Gas constant = 287.04
XKAP	$\kappa = R/C_p = 0.2858$
FC	Coriolis parameter = $2\Omega = 1.45842 \times 10^{-4}$
CD	Drag coefficient = 0.0015
DK	Diffusion coefficient = $1 \times 10^6 \text{ m}^2/\text{sec}^2$ for coarse-mesh models and 1×10^5 for fine-mesh models
CK(KMAX)	[1000/SIG(K)] **XKAP, K=1, KMAX

* These five constants are appropriate to the five-layer models. They are replaced with $S95 = \ln(0.95)$, etc. for the ten-layer models.

Note: Symbolic dimensions are given where appropriate since the size of the arrays varies with the forecast models.

b. Subroutine LSTEPP

Subroutine LSTEPP performs the first half of a pressure-gradient-force averaged time step, integrating forward one time step the pressure tendency, thermodynamic energy and moisture equations. The routine controls the data flow for one sweep of the data base and calls the routines necessary to compute diagnostics, boundary conditions, forcing functions and diabatic effects. Figure III-10 shows the controlling function of subroutine LSTEPP.

Figure III-3, which appears in Section III-2-a shows the data flow for one sweep of the data base for models PECHCV and PECHFV. This figure concentrates on the data flow aspect of the routine and the computational portion is denoted by the box labeled "computation on row J". Figure III-11 is an expansion of this box for LSTEPP and shows the step-by-step computations and the subroutines that are called.

c. Subroutine PITEND

Subroutine PITEND computes the Kurihara pressure gradient terms for use in the thermodynamic energy equation, the pressure tendencies and the vertical velocities. The terrain pressure tendency is computed using Equation [II.104] and the vertical velocity is computed using Equation [II.106].

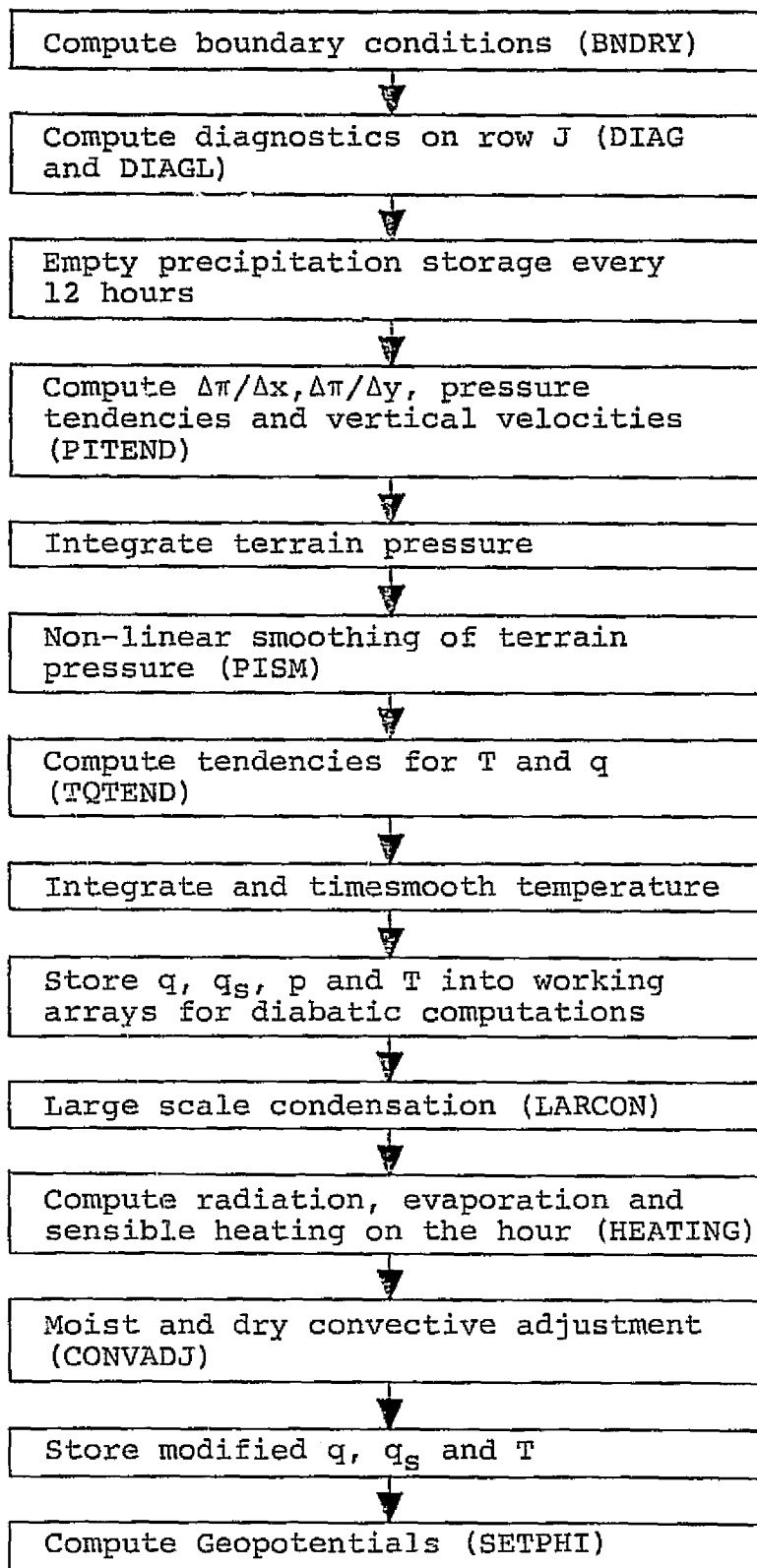


FIGURE III-11: COMPUTATION FOR ONE TIME STEP IN LSTEPP

d. Subroutine PISM

Subroutine PISM performs the non-linear terrain pressure smoothing that is described in Section II-D-2. The non-linear smoothing operator given by Equation [II.118] is applied along an entire row immediately after the terrain pressure has been advanced in time.

e. Subroutine TQTEND

Subroutine TQTEND computes the tendencies for the temperature and vapor pressure using Equations [II.105] and [II.109], respectively.

The computation of the temperature tendency requires evaluation of the horizontal Arakawa advective operator, the vertical flux and the adiabatic heating/cooling term. Computation of the vapor pressure tendency involves the horizontal Arakawa advective operator and the vertical flux.

f. Subroutine LARCON

Subroutine LARCON computes the amount of precipitation resulting from the removal of excess moisture from super-saturated regions of the atmosphere and the temperature change due to the release of latent heat. The precipitation algorithm used in this computation is described in Section II-C-2.

g. Subroutine HEATING

Subroutine HEATING performs the diabatic heating computations which include shortwave and longwave radiation and sensible heating from the planetary boundary layer as described in Section II-C-1. Evaporation of moisture from the surface into the lowest sigma level is also computed.

Since the radiation and boundary layer computations are complicated and time-consuming, they are only performed once every model-hour (normally every four time steps, when a fifteen-minute time step is used). Temperature tendencies are then computed and saved for use over the next hour of integration.

h. Function ALON

Function ALON computes the longitude of a Northern Hemispheric polar stereographic (I,J) grid point in radians ranging from 0 - 2π clockwise from the Greenwich Meridian.

i. Subroutine CONVADJ

Subroutine CONVADJ performs the moist and dry convective adjustment as described in Section II-C-3. The moist convective adjustment consists of Arakawa's parameterized cumulus convection which may result in precipitation and adjustment of the temperature and moisture in the lower three sigma layers for the five-layer models and in

the lower six sigma layers for the-ten layer models. The dry convective adjustment process consists of checking the model atmosphere for static stability and modifying the temperature structure when necessary to achieve stability.

j. Subroutine SETPHI

Subroutine SETPHI computes the new geopotentials after all the prognostic variables have been updated on the first half of the time step using Equations [II.107] and [II.108].

k. Subroutine LSTEPV

Subroutine LSTEPV performs the second half of a pressure-gradient-force averaged time step integrating the momentum equations forward one time step. The routine controls the data flow for one sweep of the data base and calls the routines necessary to compute diagnostics, boundary conditions and forcing functions. Figure III-10 shows the controlling function of subroutine LSTEPV.

Figure III-3, which appears in Section III-2-a shows the data flow for one sweep of the data base for models PECHCV and PECHFV. This figure concentrates on the data flow aspect of the routine and the computational portion is denoted by the box labeled "computation on row J". Figure III-12 is an expansion of this box for LSTEPV and shows the step-by-step computation and the subroutines that are called.

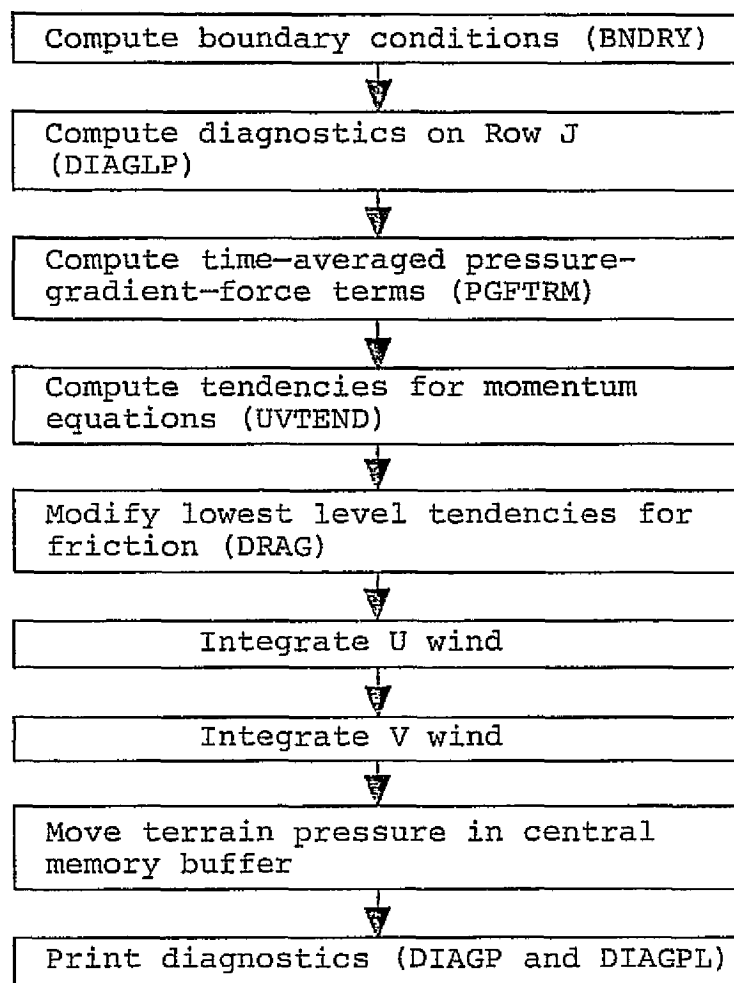


FIGURE III-12: COMPUTATION FOR ONE TIME STEP IN LSTEPV

1. Subroutine PGFTRM

Subroutine PGFTRM computes the time-averaged pressure-gradient-force terms as given by Equation [II.111] for use in the momentum equations. The Kurihara modification is used to evaluate the terms on local pressure surfaces rather than directly on the sigma surfaces.

The routine also stores the vertical velocities into a working array for use in subsequent computations.

m. Subroutine UVTEND

Subroutine UVTEND computes the tendencies for the momentum equations using Equations [II.94] and [II.95].

The computation of the tendencies for the momentum equations involves the horizontal Arakawa advective operator, the vertical flux, the time-averaged pressure-gradient-force term, the Coriolis force and the diffusion term.

n. Subroutine DRAG

Subroutine DRAG modifies the u and v momentum tendencies in the lowest layer to account for frictional dissipation in the gross boundary layer. The friction terms are evaluated using Equations [II.98] and [II.99].

o. Subroutine BNDRY

Subroutine BNDRY computes the horizontal boundary conditions for the hemispheric polar stereographic grid. As explained in Section II-B, the horizontal boundaries consist of rigid impermeable vertical walls placed on the grid on the next to outermost row of grid points. The boundary conditions computed are given by Equation [II.9].

Three entry points exist for this subroutine: LBNDRY, EWBNDRY and UBNDRY. LBNDRY computes values for the first row of the grid and is called once for each sweep of the data base. EWBNDRY computes values for the end points of each row of the grid and is required for each row as the grid is swept. UBNDRY computes values for the last row of the grid and is called once for each sweep of the data base.

p. Subroutine DIAGN

Subroutine DIAGN computes and prints various layer mean diagnostic quantities in order to monitor the forecast. Table III-12 defines the diagnostic quantities computed by DIAGN. In addition to the mean values computed by the routine, vertical column values are saved and printed for one arbitrary point.

There are three entry points in DIAGN: IDIAG, which initializes the diagnostic computations; DIAG, which computes diagnostics on a line-by-line basis; and DIAGP, which prints the layer mean diagnostics at the end of the time step.

TABLE III-12: DIAGNOSTIC QUANTITIES COMPUTED
BY SUBROUTINE DIAGN

$$\bar{U}_k = \frac{1}{N} \sum_{i,j} u_{i,j,k}, \text{ layer mean of the } u \text{ wind}$$

$$\bar{V}_k = \frac{1}{N} \sum_{i,j} v_{i,j,k}, \text{ layer mean of the } v \text{ wind}$$

$$\bar{T}_k = \frac{1}{N} \sum_{i,j} T_{i,j,k}, \text{ layer mean of the temperature}$$

$$\bar{\Phi}_k = \frac{1}{N} \sum_{i,j} \Phi_{i,j,k}, \text{ layer mean of the geopotential}$$

$$\overline{HT}_k = \frac{1}{N} \sum_{i,j} HT_{i,j,k}, \text{ layer mean of the diabatic heating rate}$$

$$\bar{W}_k = \frac{1}{N} \sum_{i,j} |W|_{i,j}, \text{ layer mean of the vertical velocity}$$

$$\bar{Q}_k = \frac{1}{N} \sum_{i,j} q_{i,j,k}, \text{ layer mean of the vapor pressure}$$

$$\overline{KE}_k = \frac{1}{N} \sum_{i,j} P_{i,j,k} (u_{i,j,k}^2 + v_{i,j,k}^2) / (2 \cdot R \cdot T_{i,j,k}),$$

layer mean of kinetic energy

$$\overline{DIVSQ}_k = \frac{1}{N} \sum_{i,j} [u_{i+1,j,k} - u_{i-1,j,k} + v_{i,j+1,k} - v_{i,j-1,k}]^2 / (2\Delta x)^2,$$

layer mean of the square of the divergence

$$\overline{VORSQ}_k = \frac{1}{N} \sum_{i,j} [(v_{i+1,j,k} - v_{i-1,j,k} - u_{i,j+1,k} + u_{i,j-1,k}) / (2\Delta x)]^2,$$

layer mean of the square of the vorticity

$$\bar{\Pi} = \frac{1}{N} \sum_{i,j} \Pi_{i,j}, \text{ hemispheric mean of the terrain pressure}$$

$$\bar{P} = \frac{1}{N} \sum_{i,j} P_{i,j}, \text{ hemispheric mean of precipitation}$$

NOTE: $N = (ICU-ICL+1) \cdot (JMAX-2)$

q. Subroutine DIAGNL

Subroutine DIAGNL computes and prints various zonal band diagnostic quantities in order to monitor the forecast. The zonal bands are ten degrees wide and range from the equator to the north pole. Table III-13 defines the diagnostic quantities computed by DIAGNL.

There are four entry points in DIAGNL: IDIAGL, which initializes the diagnostic computations; DIAGLM, which computes all quantities except the eddy kinetic energy; DIAGLP, which computes the eddy kinetic energy; and DIAGPL, which prints the zonal band diagnostics at the end of the time step.

r. Function POW

Function POW computes the power function, x^y , to six-digit accuracy to achieve a savings in computer time and is used to compute π^k and in the heating routine.

s. Subroutine SIN COS

Subroutine SIN COS computes the sine and cosine of an argument to six-digit accuracy to achieve a savings in computer time and is used in the diagnostic routine DIAGNL.

TABLE III-13: DIAGNOSTIC QUANTITIES COMPUTED BY
SUBROUTINE DIAGNL

$$\bar{U}_{L,K} = \frac{1}{N_L} \sum_{i,j} -u_{i,j,k} \cos \lambda_{i,j} + v_{i,j,k} \sin \lambda_{i,j}, \text{ zonal mean of}$$

the tangential component of the wind

$$\bar{V}_{L,K} = \frac{1}{N_L} \sum_{i,j} u_{i,j,k} \sin \lambda_{i,j} + v_{i,j,k} \cos \lambda_{i,j}, \text{ zonal mean of}$$

the normal component of the wind

$$\overline{KE}_{L,K} = \frac{1}{N_L} \sum_{i,j} P_{i,j,k} (u_{i,j,k}^2 + v_{i,j,k}^2) / (2 \cdot R \cdot T_{i,j,k}), \text{ zonal}$$

mean of the kinetic energy

$$\overline{KE'}_{L,K} = \frac{1}{N_L} \sum_{i,j} P_{i,j,k} [(u_{i,j,k} - \bar{U}_{L,K})^2 + (v_{i,j,k} - \bar{V}_{L,K})^2] / (2 \cdot R \cdot T_{i,j,k}),$$

zonal mean of the eddy kinetic energy

$$\bar{T}_{L,K} = \frac{1}{N_L} \sum_{i,j} T_{i,j,k}, \text{ zonal mean of the temperature}$$

$$\bar{W}_{L,K} = \frac{1}{N_L} \sum_{i,j} |W|_{i,j,k}, \text{ zonal mean of the vertical velocity}$$

$$\overline{DIVSQ}_{L,K} = \frac{1}{N_L} \sum_{i,j} [(u_{i+1,j,k} - u_{i-1,j,k} + v_{i,j+1,k} - v_{i,j-1,k}) / 2\Delta x]^2,$$

zonal mean of the square of the divergence

$$\overline{VORSQ}_{L,K} = \frac{1}{N_L} \sum_{i,j} [(v_{i+1,j,k} - v_{i-1,j,k} - u_{i,j+1,k} + u_{i,j-1,k}) / 2\Delta x]^2,$$

zonal mean of the square of the vorticity

NOTE: L denotes the zonal band and N_L denotes the number
of grid points lying in that band

4. Output Overlay - Program OUTPE

The function of the output overlay is to take the results of the forecast contained on file IOLD and to produce meteorologically meaningful output; that is, to create surface pressure, precipitation and geopotential heights, temperatures and winds at standard levels for viewing by meteorologists. In addition, the output overlay writes a restart data tape on each output cycle (nominally set at twelve-hour intervals).

Figure III-13 shows the structure of the output overlay, giving the names of all of the subprograms which constitute this overlay. The function of each of these routines is described in the sections that follow.

Table III-14 defines the fields which are produced by the output overlay and written to TAPE4.

Table III-15 shows the common blocks local to this overlay.

a. Subroutine SORTOUT

Since the output of the forecast contained on file IOLD is data arranged in vertical slice format and the required output is data on horizontal planes, an interface routine to properly structure the data is required. Subroutine SORTOUT takes forecast data in vertical slice format

III-71

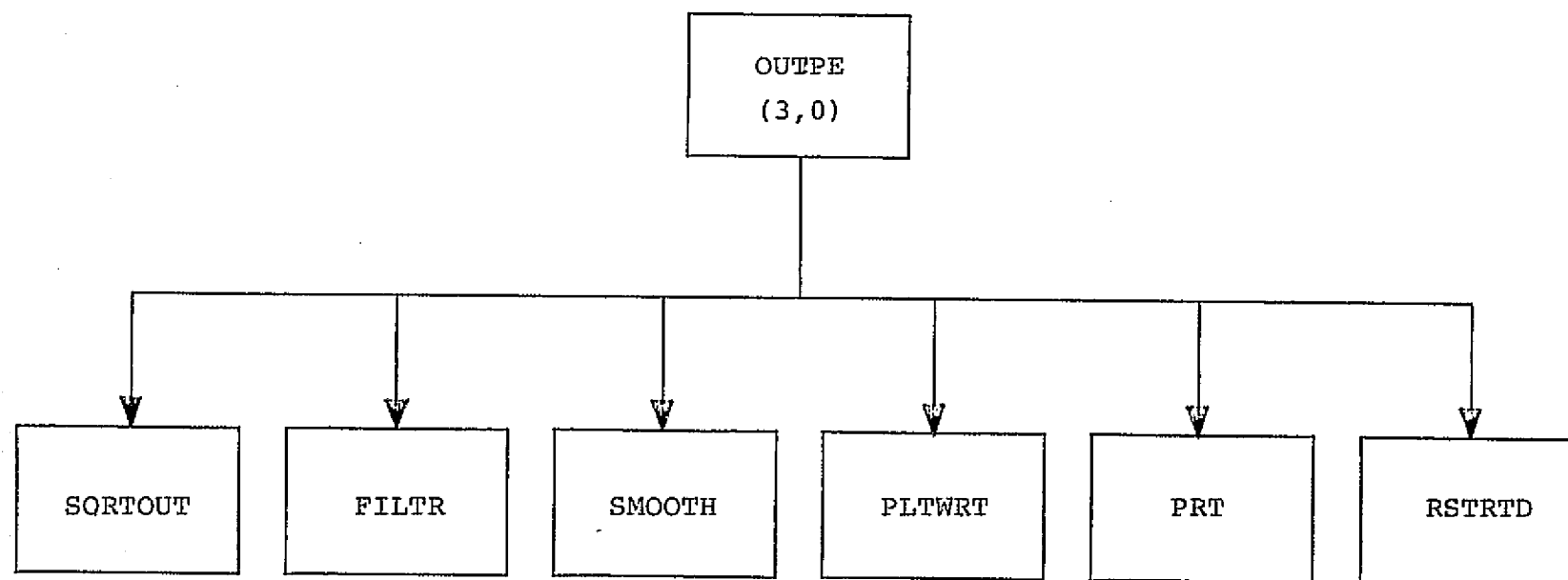


FIGURE III-13: OUTPUT OVERLAY, PROGRAM OUTPE -
MODELS PECHCV AND PECHFV

TABLE III-14: CONTENTS OF THE OUTPUT DATA FILE -
TAPE4 PER OUTPUT CYCLE

<u>Record</u>	<u>Data Field</u>						
1	Forecast Surface Pressure, Ps (mb)						
2	Precipitation (cm)						
3	Geopotential Heights of 1,000 mb Surface (meters)						
4	"	"	"	950	"	"	"
5	"	"	"	900	"	"	"
6	"	"	"	850	"	"	"
7	"	"	"	700	"	"	"
8	"	"	"	500	"	"	"
9	"	"	"	400	"	"	"
10	"	"	"	300	"	"	"
11	"	"	"	250	"	"	"
12	"	"	"	200	"	"	"
13	"	"	"	150	"	"	"
14	"	"	"	100	"	"	"
15	Temperatures on the 1,000 mb Surface (°C)						
16	"	"	"	950	"	"	"
17	"	"	"	900	"	"	"
18	"	"	"	850	"	"	"
19	"	"	"	700	"	"	"
20	"	"	"	500	"	"	"
21	"	"	"	400	"	"	"
22	"	"	"	300	"	"	"
23	"	"	"	250	"	"	"
24	"	"	"	200	"	"	"
25	"	"	"	150	"	"	"
26	"	"	"	100	"	"	"
27	U winds on 1000 mb surface (m/sec)						
28	V	"	"	1000	"	"	"
29	U	"	"	950	"	"	"
30	V	"	"	950	"	"	"

TABLE III-14: OUTPUT DATA FILE - TAPE4 (Continued)

<u>Record</u>	<u>Data Field</u>						
31	U winds on	900 mb surface	(m/sec)				
32	V	"	"	900	"	"	"
33	U	"	"	850	"	"	"
34	V	"	"	850	"	"	"
35	U	"	"	700	"	"	"
36	V	"	"	700	"	"	"
37	U	"	"	500	"	"	"
38	V	"	"	500	"	"	"
39	U	"	"	400	"	"	"
40	V	"	"	400	"	"	"
41	U	"	"	300	"	"	"
42	V	"	"	300	"	"	"
43	U	"	"	250	"	"	"
44	V	"	"	250	"	"	"
45	U	"	"	200	"	"	"
46	V	"	"	200	"	"	"
47	U	"	"	150	"	"	"
48	V	"	"	150	"	"	"
49	U	"	"	100	"	"	"
50	V	"	"	100	"	"	"

NOTE: Each data field has an identification block of twenty words appended to it, which uniquely identifies the forecast base and forecast time plus parameters for plotting of the fields by the graphics package.

TABLE III-15: COMMON BLOCKS LOCAL TO
THE OUTPUT OVERLAY -
MODELS PECHCV AND PECHFV

COMMON /WORK/
FPI(63,63,12)

Used as temporary storage
for the forecast fields on
horizontal planes.

COMMON /BUFFP/ AO(3989)

Used as input/output buffer
area for file IOLD.

on sigma surfaces and constructs data on standard pressure surfaces and stores it on the random access file TAPE2.

In order to construct a data field for output purposes, the forecast values must be interpolated or extrapolated to standard pressure levels as the file IOLD is swept, row by row. This process creates horizontal plane data fields. The interpolation process is straightforward using logarithmic interpolation in pressure to generate output data on pressure surfaces. However, when a pressure surface is below the terrain, simple extrapolation can give very unrealistic results and a more sophisticated procedure is required.

The method described below is a variation of the one in use at FNWC and is based on the work of Bellamy (1945). It basically uses the standard heights and temperatures of pressure surfaces and modifies them according to the conditions at $\sigma = 1.0$. Define:

$$Z_{1.0} = \left(\frac{288.16}{6.5 \times 10^{-3}} \right) \left[1 - \left(\frac{\pi}{1013.25} \right)^{\left(\frac{1.0}{5.2561} \right)} \right] \quad [\text{III.9}]$$

$$S_{1.0} = \frac{T_{1.0}}{288.16 - Z_{1.0} (6.5 \times 10^{-3})} - 1 \quad [\text{III.10}]$$

$$Z_{\text{SFC}} = \left(\frac{288.16}{6.5 \times 10^{-3}} \right) \left[1 - \left(\frac{P_s}{1013.25} \right)^{\left(\frac{1.0}{5.2561} \right)} \right] \quad [\text{III.11}]$$

and

$$\bar{S} = \frac{Z_T}{Z_{1.0} - Z_{SFC}} - 1 \quad [III.12]$$

where

$$T_{1.0} = 1.5 T_{0.9} - 0.5 T_{0.7} \quad [III.13]$$

Then if the difference $Z_{1.0} - Z_{SFC}$ is less than 500 m.,
the height and temperature at standard level L is given by:

$$Z(L) = [Z_{STD}(L) - Z_{SFC}] [1 + S_{1.0}] \quad [III.14]$$

and

$$T(L) = T_{STD}(L) [1 + S_{1.0}] \quad [III.15]$$

Otherwise, by

$$Z(L) = [Z_{STD}(L) - Z_{SFC}] [1 + \frac{1}{2}(3\bar{S} - S_{1.0})] \quad [III.16]$$

and

$$T(L) = T_{STD}(L) [1 + \bar{S}] \quad [III.17]$$

where $Z_{STD}(L)$ and $T_{STD}(L)$ are the standard height and temperature, respectively, of the standard pressure level, L. For graphical presentation, the surface pressure, geopotential heights and temperatures are then smoothed to remove small-scale computational noise from the forecast values. The forecast fields are then written to the random access file TAPE2. Using subroutine PLTWRT which prefixes a twenty-word identification record to the fields, the forecast fields are written to the forecast file TAPE4.

b. Subroutine PLTWRT

Subroutine PLTWRT prefixes a twenty-word identification record to a data field and then writes the field to a specified output unit. The identification record contains forecast and field identifiers and contouring information for the graphical display package.

c. Subroutine FILTR

Subroutine FILTR is a non-linear smoothing routine used to remove computational noise from selected output forecast fields. Three types of filter can be chosen: a medium wave filter with a cutoff wave length of six grid lengths; a short wave filter with a cutoff wave length of four grid lengths; and a long wave filter with a cutoff wave length of thirty grid lengths.

d. Subroutine SMOOTH

Subroutine SMOOTH is a diffusive type of data field smoother that only smooths points whose latitude is less than thirty degrees. These points are then smoothed according to the formula

$$F'_{i,j} = F_{i,j} + 0.1 \nabla^2 (F) \quad [\text{III.18}]$$

Multiple passes are made over the data field.

e. Subroutine PRT

Subroutine PRT is a specialized printer contour routine which contours 63 x 63 data fields on four printer pages. It is used to output a selected few data fields to get a quick look at the quality of the forecast.

f. Subroutine RSTRTD

Subroutine RSTRTD writes restart data to TAPE3 during each output cycle for possible restart use in the event of computer malfunction or if it should be desired to continue a forecast at a later date. The format of TAPE3 is given in Table III-10, which appears in Section III-B-2.

C. Code Structure - PEFHCV and PEFHFV

This section describes the program structure of the forecast models PEFHCV and PEFHFV. It also defines the various files used by the programs, defines the common blocks that are used and describes each of the subprograms that constitute the models.

The codes are written in CDC Fortran, which is a dialect of FORTRAN IV, for the CDC FORTRAN Extended Compiler (OPT=2 level). The two programs, PEFHCV and PEFHFV, have been designed to use rotating mass storage along with central memory and either ECS or LCM.

The forecast models are fairly large codes. In fact, the models consist of a set of two programs. PEPREP sorts out the analysis data into a vertical slice-pressure surface format. The main forecast model code consists of PEFHCV or PEFHFV. Section 1 describes the analysis data program, PEPREP, giving a block diagram of the program structure, and describing each of the subprograms.

The forecast models are fairly large codes. Since the various sections of the codes -- initialization, integration and output -- are each fairly large, the main forecast model codes have been overlaid to more efficiently use central memory. Section 2 describes the control overlay and gives a block diagram of the overlay structure, defines the I/O file structure and defines the common blocks that are global to all overlays.

Section 3 describes the initialization overlay defining the common blocks local to this overlay and describing each of the subprograms that are used by this overlay. Similarly, Sections 4 and 5 describe the integration and output overlays, respectively.

1. Analysis Data - Program PEPREP

Due to the large size of the data base for the 187 x 187 forecast models (approximately one million words), it is much more efficient to sort out the analysis data into a pressure surface - vertical slice format prior to doing the sigma surface interpolation in the initialization section of the forecast model. Program PEPREP takes analyzed data on pressure surfaces and geophysical data, computes moistures and 50 mb winds, and, finally, sorts out this large mass of data producing a vertical slice - pressure surface format data file.

The structure of the program PEPREP is shown in Figure III-14 and the various files used by PEPREP are defined in Table III-16. In addition, the program uses both rotating mass storage and either ECS or LCM (approximately 250,000 words) as temporary data storage. The contents of the input data file, TAPE1, have been defined previously in Table III-8 in Section III-2 with the exception that the 187 x 187 analysis data records are packed 2:1 prefixed by a 20 word identification header.

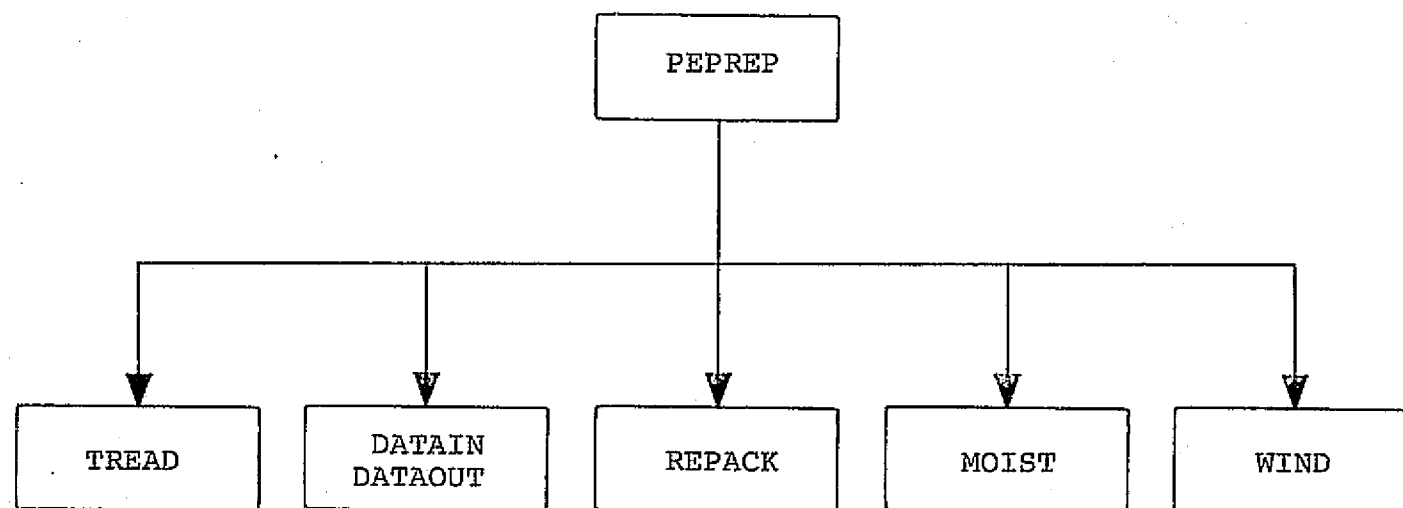


FIGURE III-14: STRUCTURE OF ANALYSIS DATA PROGRAM - PEPREP

TABLE III-16: FILE STRUCTURE OF ANALYSIS
DATA PROGRAM - PEPREP

OUTPUT	Used to write various diagnostic messages
TAPE1	Input data file generated by analysis program
TAPE2	Random access file used for intermediate storage of data fields
TAPE5, TAPE6	(JOLD, JNEW) Vertical slice analysis data on pressure surfaces for input to the main forecast models PEFHCV or PEFHFV

a. Subroutine TREAD

Subroutine TREAD acts as a generalized input data routine to obtain analysis data from TAPE1. A storage array address and the data field ID are input to TREAD from the calling routine. TREAD then searches a table (array ITP) to find the location of the desired data field and reads the data field from TAPE1 into the storage array. Various data field orderings on TAPE1 may be accommodated simply by changing the information stored in array ITP.

b. Subroutine DATAIO

Subroutine DATAIO is a generalized routine for writing and reading data fields to and from the random access file TAPE2. Two entry points exist for this routine: DATAOUT, which uses the system routine WRITMS for writing to the file, and DATAIN, which uses the system routine READMS for reading from the file.

c. Subroutine REPACK

Subroutine REPACK repacks analysis data in a 2:1 format for use in generating the data records for the vertical slice - pressure surface file. For example, if one has two packed analysis data arrays consisting of the u and v wind components, then REPACK puts these two arrays into one 187 x 187 packed array with the u component of the wind in

the upper half of the word and the v component in the lower half of the word. The routine is also used for repacking other data fields.

d. Subroutine MOIST

Subroutine MOIST computes the relative humidity on a pressure surface using the procedure based on the relative vorticity as described in Section II-E-6. Input to the routine consists of the geopotential heights of the pressure surface. The routine then computes the relative vorticity on the pressure surface and, using the empirical relation given by Equation [II.133], computes the relative humidity on the pressure surface. Program PREPREP calls this subroutine five times to generate moistures on the 1,000, 900, 700, 500 and 300 mb surfaces.

e. Subroutine WIND

Subroutine WIND computes the geostrophic wind components at 50 mb for use in the initialization of the winds. Input to the routine consists of the geopotential heights of the 50 mb pressure surface.

2. Main Overlay - Programs PEFHCV and PEFHFV

The main overlay of the forecast model programs is very simple, since it only performs the function of declaring three global common blocks, defining various control variables and calling the actual working overlays. The overlay structure of the programs is shown in Figure III-15.

The various files used by the programs are defined in Table III-17. Tables III-18 and III-19 define the variables that appear in the global common blocks CNTRL, and INDEX, respectively. Common block BUFPOINT consists of the offsets for the variables and contains the integer variables listed in Table III-3.

a. Subroutine BUFERR

Subroutine BUFERR prints a diagnostic message when an error occurs on a BUFFER (IN or OUT) operation that tells where the error occurred.

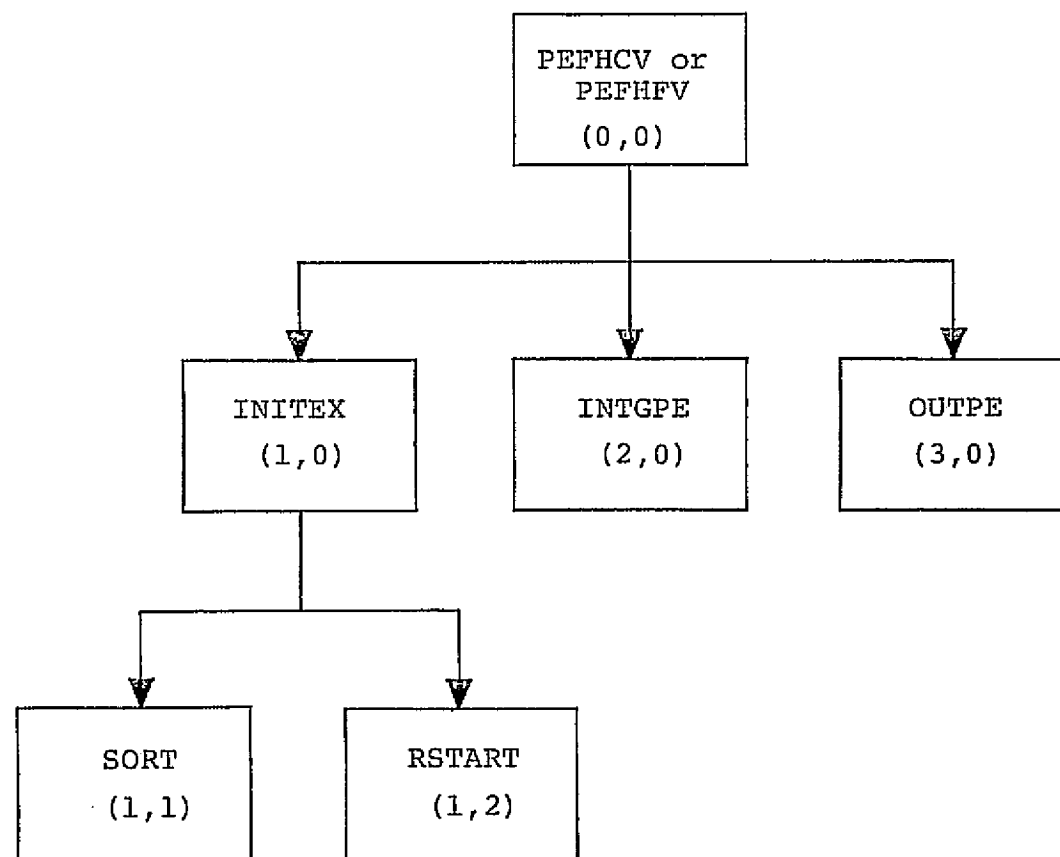


FIGURE III-15: OVERLAY STRUCTURE OF FORECAST MODELS
PEFHCV AND PEFHFV

TABLE III-17: FILE STRUCTURE OF
FORECAST MODELS
PEFHCV AND PEFHFV

OUTPUT	Used to write various diagnostic messages
TAPE2	Random access file used by output overlay (four records only)
TAPE3	Restart data tape; contains a header record and contents of current IOLD file
TAPE 4	Output data file; contains forecast fields to be selectively plotted by graphics program and for use by the analysis programs
TAPE5	Input data file for initialization overlay; generated by program PEPREP
TAPE8	(IOLD, INEW) integration data
TAPE9	(IOLD, INEW) integration data

TABLE III-18: DEFINITION OF VARIABLES IN COMMON
BLOCK /CNTRL/-MODELS PEFHCV AND PEFHFV

IOLD	File name of integration data (TAPE8 or TAPE9)
INEW	File name of integration data (TAPE8 or TAPE9)
ISW	=1, SSW1 is "on", signifying normal start with data from TAPE1 =2, SSW1 is "off", signifying restart with data from TAPE3
ID(2)	1st word, forecast identifier (date-time group) 2nd word, forecast data field identifier
LIMI	Starting cycle
LIM	Number of integration and output cycles (2 x number of days + 1)
ITAU	Forecast time indicator (hours) = 0 during initialization and first output cycle = 12*LCO for normal integration cycle
ITSPHR	Number of integration time steps per hour
NN	Time step counter, reset after each output cycle, also NN=0 signifies a Matsuno time step
NNOUT	Number of time steps between normal output cycles, usually equal to 12 x ITSPHR
LCO	Output cycle counter, updated by 1 each output cycle
L12	Indicator set at 12-hour output cycle, used to zero rainfall
DAY	Number of days from January 1 of the forecast
HR	Hour of the day, GMT
DT	Time step, Δt
TDT	Twice the time step, $2\Delta t$
MATS	Logical variable used during Matsuno time step = .FALSE. during first half = .TRUE. during second half
MFRQ	Number of hours between Matsuno time steps

TABLE III-19: DEFINITION OF VARIABLES IN COMMON
BLOCK /INDEX/-MODELS PEFHCV AND PEFHFV

ICL	Start of computation on data row (=2)
ICU	End of computation on data row (=IMAX-1)
IMAXM	Length of data row -1
IMAX	Length of data row
JMAXM	Number of data rows -1
JMAX	Number of data rows
KMAXQ	Number of moisture levels
KMAXM	Number of vertical levels -1
KMAX	Number of vertical levels
LINE	Number of row being worked on
NFLD	Size of a data record on TAPE2 (63 x 63)
NFLDEX	Expanded data record (187 x 187)
NVARS	Size of the AJM and AJP buffers
NVARL	Size of the AJ buffer and a data record on TAPE 8 or TAPE9

3. Initialization Overlay - Program INITEX

The function of the initialization overlay is to take as input data the analysis data in vertical slice - pressure surface format generated by Program PEPREP and to generate an initial state for the integration overlay which actually performs the forecast. That is, the interpolation of data fields to the PEM sigma surfaces as described in Section II-E is performed by the initialization overlay.

Figure III-16 shows the structure of the initialization overlay, giving the names of the subprograms which constitute the overlay.

Program INITEX performs the simple executive function of either calling overlay RSTART to initialize from a previously written restart file, or by calling overlay SORT to initialize from the analysis data file written by Program PEPREP.

a. Program SORT

Program SORT takes the analysis data in vertical slice - pressure surface format as generated by program PEPREP and generates the integration data file in vertical slice - sigma surface format. The interpolation to sigma surfaces proceeds one vertical slice at a time sweeping the input data file and creating the integration data file. The interpolation process for converting pressure surface data to sigma surfaces is described in Section II-E.

16-III

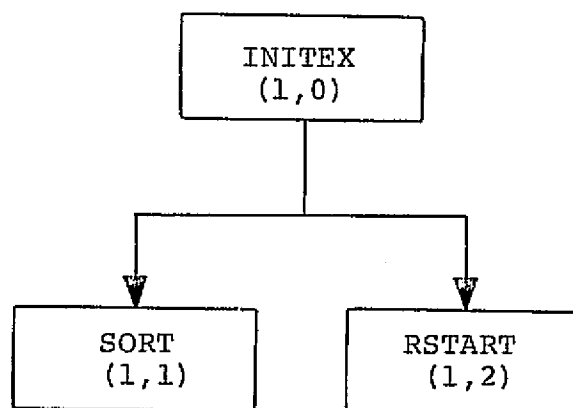


FIGURE III-16: INITIALIZATION OVERLAY STRUCTURE -
MODELS PEFHCV AND PEFHFV

b. Program RSTART

Program RSTART reads the restart data file, TAPE3, when a forecast run is being restarted and initializes the forecast model in order to continue the forecast. Restarts are controlled by sense switch 1. If SSW1 is "ON", then a normal run is being made; if SSW1 is "OFF", then a restart run is being made.

A restart data tape is written during each output cycle so if the computer goes down, or if for some reason, a forecast is to be continued at a later time, the forecast can be resumed from the point at which the last restart tape was written. Table III-10 gives the format of the restart data tape TAPE3.

Thus, program RSTART reads in the header record from TAPE3 and initializes the required control variables. Next the routine reads in the forecast data records and recreates file IOLD. The forecast run will now proceed from this point in the forecast.

4. Integration Overlay - Program INTGPE

The function of the integration overlay is to numerically integrate the primitive equations; that is, to step the data fields forward in time to produce a forecast. Input to this overlay consists initially of the data field specification produced by the initialization overlay and, subsequently, the current state of the forecast contained on file IOLD. After initialization, a forecast consists of cycling through the integration and output overlays until the desired forecast time has been reached.

Figure III-10 in Section III-B-3 shows the structure of the integration overlay, giving the names of all the subprograms which constitute this overlay. Table III-20 defines the variables that appear in the common blocks that are local to the integration overlay for programs PEFHCV and PEFHFV. Table III-11 in Section III-B-3 defines the variables in common block /CON/ which is the same for all four forecast model programs.

With the exception of the indexing method used to access variables in the working arrays, the subprograms in the integration overlay are identical to those occurring in models PECHCV and PECHFV. Therefore, the descriptions of the subprograms are not repeated in this section and the reader is referred to Section III-B-3.

TABLE III-20: DEFINITION OF VARIABLES IN
COMMON BLOCKS LOCAL TO THE
PROGRAMS PEFHCV AND PEPHFV

Common Block /LCM/

AIN(NVARL,3) Circular input buffer for rows JM,
J and JP; resides in either ECS or LCM

Common Block /BUFFS/

AJM(NVARS) Buffer for partial row JM in central
memory

AJ(NVARL) Buffer for full row J in central memory

AJP(NVARS) Buffer for partial row JP in central
memory

Common Block /LOCAL/

PIKAP(187) π^K on row J

W(187,KMAXM) Vertical velocities for all levels on
row J

DUM(187,KMAX+4) Dummy array

NOTE: DUM is equivalenced to the
following variables

a) PILN(187,3) $\ln(\pi)$ on rows JM, J and JP

b) UPM(187) $\frac{u\pi}{m}$ on row J at a particular level

VPM(187,3) $\frac{v\pi}{m}$ on rows JM, J and JP at a particular
level

AB(187,KMAX) $\frac{\Delta \sigma m^2}{2\Delta x} (\delta\alpha + \delta\beta)$ at all levels on row J

c) PIF(187,3) π reduction factor to sea level

PEPS(187) π change due to nonlinear smoothing
operator

TABLE III-20: Definition of Variables (Continued)

Common Block /LTEND/

TENDP(187,2*KMAX)	Dummy tendency array
DPIDX(187,KMAX)	Pressure-gradient-force terms in x direction for all levels on row J
DPIDY(187,KMAX)	Pressure-gradient-force terms in y direction for all levels on row J
TCOR(187,3,KMAX)	Temperatures corrected for vapor pressure at all levels on rows JM, J and JP

NOTE: /LTEND/ resides in central memory on the Cyber 175 and in LCM on the Cyber 76

NOTE: TENDP is equivalenced to the following variables:

a) PIT(187)	$\frac{\partial \pi}{\partial t}$, terrain pressure tendency on row J
TT(187,KMAX)	$\frac{\partial \pi T}{\partial t}$, tendencies in πT for all levels on row J
QT(187,KMAXQ)	$\frac{\partial \pi q}{\partial t}$, tendencies in πq for moisture levels on row J
b) UT(187,KMAX)	$\frac{\partial \pi u}{\partial t}$, tendencies in πu for all levels on row J
VT(187,KMAX)	$\frac{\partial \pi v}{\partial t}$, tendencies in πv for all levels on row J

NOTE: TCOR is equivalenced to the following variables:

PC(187,KMAXQ)	Pressure at moisture levels on row J
---------------	--------------------------------------

TABLE III-20: Definition of Variables (Continued)

QC(187,KMAXQ)	Vapor pressure at moisture levels on row J
QCS(187,KMAXQ)	Saturation vapor pressure at moisture levels on row J
TC(187,KMAX)	Temperature at all levels on row J
RF(187)	Rainfall on row J

NOTE: The following common block only appears in the model PEFHCV.

Common Block /SCM/

IBUF(NVARL)	Input buffer for file IOLD
OBUF(NVARL)	Output buffer for file INEW

NOTE: Symbolic dimensions are given where appropriate since the size of the array varies with the forecast model.

5. Output Overlay - Program OUTPE

The function of the output overlay is to take the results of the forecast contained on file IOLD and to produce meteorologically meaningful output; that is, to create surface pressure, precipitation and geopotential heights, temperatures and winds at standard levels for viewing by meteorologists. In addition, the output overlay writes a restart data tape on each output cycle (nominally set at twelve-hour intervals).

Figure III-17 shows the structure of the output overlay, giving the names of all of the subprograms which constitute this overlay. The function of each of these routines is described in the sections that follow.

Table III-14 in Section III-B-4 defines the fields which are produced by the output overlay and written to TAPE4. Table III-21 shows the common blocks local to this overlay.

a. Subroutine SORTOUT

Since the output of the forecast contained on file IOLD is data arranged in vertical slice format and the required output is data on horizontal planes, an interface routine to properly structure the data is required. Subroutine SORTOUT takes forecast data in vertical slice format on sigma surfaces and constructs data on standard pressure surfaces.

86-III

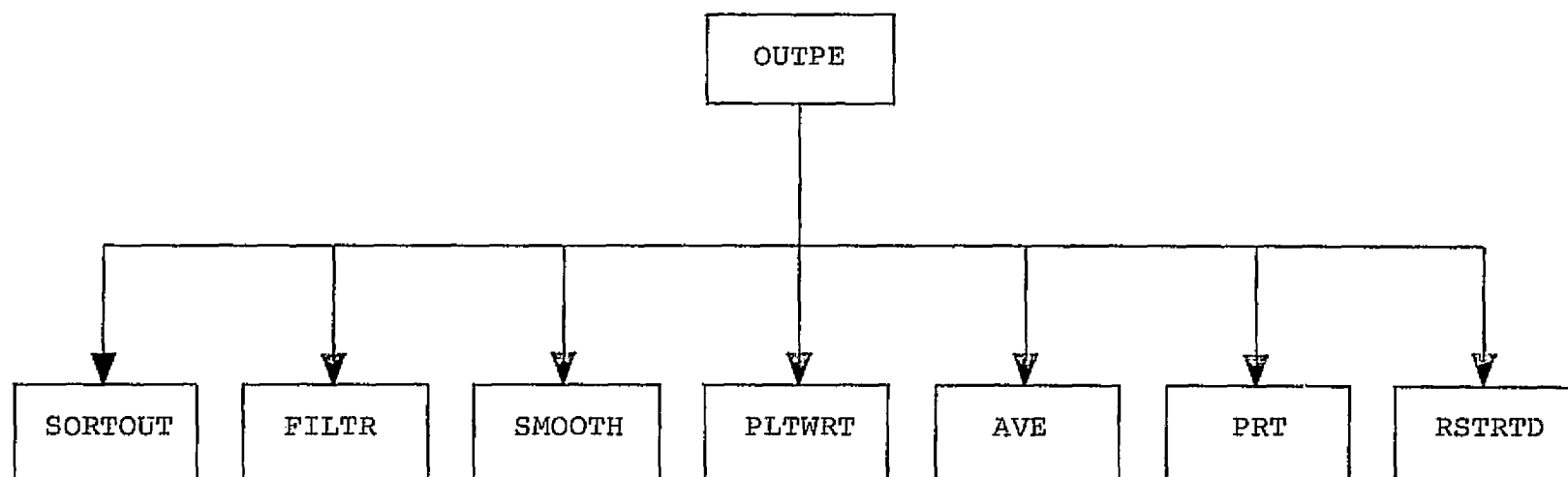


FIGURE III-17: OUTPUT OVERLAY, PROGRAM OUTPE -
MODELS PEFHCV AND PEFHVV

TABLE III-21: COMMON BLOCKS LOCAL TO THE OUTPUT
OVERLAY - MODELS PEFHCV AND PEFHFV

COMMON /WORKX/ DUME(20), FA(187,187)	Used as temporary storage for the forecast fields on horizontal planes and as an input buffer area for files IOLD and the analysis data file.
COMMON /BUFFP/ AO(3969)	Used as a working storage area.
COMMON /LCM/ FL(187,187,3)	Buffer area in ECS or LCM used during the interpolation/ sorting process.

In order to construct a data field for output purposes, the forecast values must be interpolated or extrapolated as explained in Section III-B-4-a to standard pressure levels as the file IOLD is swept, row by row. This process creates horizontal plane data fields. For graphical presentation, the surface pressure, geopotential heights and temperatures are then smoothed to remove small-scale computational noise from the forecast values. Using subroutine PLTWRT which appends a twenty-word identification record to the fields, the forecast fields are written to the forecast file, TAPE4.

b. Subroutine PLTWRT

Subroutine PLTWRT prefixes a twenty-word identification block to a data field, packs the data field into a 2:1 format, and then writes the packed plot field to a specified output unit. The identification block contains forecast and field identifiers and contouring information for the graphical display package.

c. Subroutine FILTR

Subroutine FILTR is a non-linear smoothing routine used to remove computational noise from selected output forecast fields. Three types of filter can be chosen: a medium wave filter with a cutoff wave length of six grid

lengths; a short wave filter with a cutoff wave length of four grid lengths; and a long wave filter with a cutoff wave length of thirty grid lengths.

d. Subroutine SMOOTH

Subroutine SMOOTH is a diffusive type of data field smoother that only smooths points whose latitude is less than thirty degrees. These points are then smoothed according to the formula

$$F'_{i,j} = F_{i,j} + 0.1 \nabla^2 (F) \quad \text{[III.19]}$$

Multiple passes are made over the data field.

e. Subroutine PRT

Subroutine PRT is a specialized printer contour routine which contours 63 x 63 data fields on four printer pages. It is used to output a selected few data fields to get a quick look at the quality of the forecast.

f. Subroutine RSTRTD

Subroutine RSTRTD writes restart data to TAPE3 during each output cycle for possible restart use in the event of computer malfunction or if it should be desired to continue a forecast at a later date. The format of TAPE3 is given in Table III-10, which appears in Section III-B-2.

g. Subroutine AVE

Subroutine AVE transforms forecast data from a 187 x 187 grid to a 63 x 63 grid so that contours may be plotted using subroutine PRT. Every third point along the boundaries of the 187 x 187 grid is moved to the boundaries of the 63 x 63 grid. The interior points are averaged via the following algorithm:

$$\begin{aligned}\bar{F} = \frac{1}{16} (& 4F_{i,j} + 2(F_{i+1,j} + F_{i,j+1} + F_{i-1,j} + F_{i,j-1}) \\ & + F_{i+1,j-1} + F_{i+1,j+1} + F_{i-1,j+1} + F_{i-1,j-1}) \quad \text{[III.20]}\end{aligned}$$

REFERENCES

- ARAKAWA, Akio, "Computational Design for Long-Term Numerical Integration of the Equation of Fluid Motion: Two-Dimensional Incompressible Flow, Part I", Journal of Computational Physics, Vol. 1, No. 1, Academic Press Inc., New York, N.Y., January 1966, pp. 114-143.
- ARAKAWA, Akio, Akira Katayama and Yale Mintz, "Numerical Simulation of the General Circulation of the Atmosphere", Proceedings of the WMO/IUGG Symposium on Numerical Weather Prediction, Tokyo, Japan, November 26 - December 4, 1968, Japan Meteorological Agency, Tokyo, March 1969, pp. IV-1 - IV-14.
- ASSELIN, Richard, "Frequency Filter for Time Integration", Monthly Weather Review, Vol. 100, No. 6, June 1972, pp. 487-490.
- BELLAMY, J.C., "The Use of Pressure Altitude and Altimeter Corrections in Meteorology", Journal of Meteorology, II, 1945, pp. 1-79.
- DANARD, Maurice B., "A Simple Method of Including Longwave Radiation in a Tropospheric Numerical Prediction Model", Monthly Weather Review, Vol. 97, No. 1, January 1969, pp. 77-85.
- GERRITY, J.P., Jr. and S.H. Scolnick, "Some Comments on Robert's Time Filter for Time Integration", U.S. Dept. of Commerce NOAA, National Weather Service, Office Note 62, November 1971, 9 pp.
- JOSEPH, Joachim A., "Calculation of Radiation Heating in Numerical General Circulation Models", Numerical Simulation of Weather and Climate Technical Report No. 1, Dept. of Meteorology, Univ. of Calif., Los Angeles, August 1, 1966, 60 pp.
- KAITALA, J.E., "Heating Functions and Moisture Source Terms in the FNWC Primitive Equation Models", unpublished manuscript, April 1974, 84 pp.
- KESSEL, Philip G. and Howard L. Lewit, "Primitive Equation Model: I. Moisture Initialization; II. PE Model Diagnostics", ODSI Final Technical Report, Contract No. N66314-74-C-1390, U.S. Navy Environmental Prediction Research Facility, December 1974.

- KESEL, Philip G. and Francis J. Winninghoff, "The Fleet Numerical Weather Central Operational Primitive Equation Model", Monthly Weather Review, Vol. 100, No. 5, May 1972, pp. 360-373.
- KURIHARA, Yoshio, "Correspondence Note on Finite Difference Expressions for the Hydrostatic Relation and Pressure Gradient Force", Monthly Weather Review, Vol. 96, No. 9, September 1968, pp. 654-656.
- LANGLOIS, W.C. and C.W. Kwok, "Description of the Mintz-Arakawa Numerical General Circulation Model", Numerical Simulation of Weather and Climate Technical Report No. 3, Dept. of Meteorology, Univ. of Calif., Los Angeles, 1969, 95 pp.
- MANABE, Syukuru and Fritz Moller, "On the Radiative Equilibrium and Heat Balance of the Atmosphere", Monthly Weather Review, Vol. 39, No. 12, December 1961, pp. 503-532.
- OLIGER, Joseph, Robert Wellick, Akira Kasahara and Warren Washington, "Description of NCAR Global Circulation Model", National Center for Atmospheric Research, NCAR-TN/STR-56, May 1970.
- PHILIPS, Norman A., "A Coordinate System Having Some Special Advantages for Numerical Forecasting", Journal of Meteorology, Vol. 14, No. 4, April 1957, pp. 184-185.
- ROBERT, Andre J., "The Integration of a Low Order Spectral Form of the Primitive Meteorological Equations", Journal of the Meteorological Society of Japan, Serial 2, Vol. 44, No. 5, Tokyo, October 1966, pp. 237-245.
- SHUMAN, Frederick G. and John B. Hovermale, "An Operational Six-Layer Primitive Equation Model", Journal of Applied Meteorology, Vol. 7, No. 4, August 1968, pp. 525-547.
- SHUMAN, F.G., "Resuscitation of an Integration Procedure", National Meteorological Center Office Note 54, 1971, 15 pp.
- SMAGORINSKY, Joseph, Syukuru Manabe and J. Leith Holloway, Jr., "Numerical Results from a Nine-Level General Circulation Model of the Atmosphere", Monthly Weather Review, Vol. 93, No. 12, December 1965, pp. 727-768.