

# MM5

## 8

Purpose 8-3

Basic Equations of MM5 8-3

Physics Options in MM5 8-7

- Cumulus Parameterizations (ICUPA) 8-7

- PBL Schemes (IBLTYP) 8-8

- Explicit Moisture Schemes (IMPHYS) 8-10

- Radiation Schemes (IFRAD) and Diffusion 8-13

- Surface Schemes (ISOIL) 8-14

Interactions of Parameterizations 8-17

Boundary conditions 8-17

- Lateral boundary conditions (IBOUDY) 8-17

- Lower boundary conditions 8-18

- Upper boundary condition (IFUPR) 8-18

Nesting 8-18

- One-way nesting 8-18

- Two-way nesting 8-18

- Two-way nest initialization options (IOVERW) 8-18

- Two-way nesting feedback options (IFEED) 8-19

Four-Dimensional Data Assimilation (FDDA) 8-20

- Introduction 8-20

- FDDA Method 8-20

- Uses of FDDA 8-20

- Data used in FDDA 8-21

How to run MM5 8-22

- Compiling MM5 8-22

- Running MM5 8-22

- Running MM5 Batch Job on NCAR's IBM 8-23

- Useful make commands 8-23

Input to MM5 8-24

Output from MM5 8-24

MM5 Files and Unit Numbers 8-27

Configure.user Variables 8-28

Script Variables for IBM Batch Deck: 8-30

Namelist Variables 8-30

OPARAM 8-31

LPARAM 8-31

NPARAM 8-34

PPARAM 8-35

FPARAM 8-35

Some Common Errors Associated with MM5 Failure 8-36

MM5 tar File 8-37

Configure.user 8-39

mm5.deck 8-54

8

MM5

## 8.1 Purpose

- This is the numerical weather prediction part of the modeling system.
- MM5 can be used for a broad spectrum of theoretical and real-time studies, including applications of both predictive simulation and four-dimensional data assimilation to monsoons, hurricanes, and cyclones.
- On the smaller meso-beta and meso-gamma scales (2-200 km), MM5 can be used for studies involving mesoscale convective systems, fronts, land-sea breezes, mountain-valley circulations, and urban heat islands.

## 8.2 Basic Equations of MM5

In terms of terrain following coordinates ( $x$ ,  $y$ ,  $\sigma$ ), these are the equations for the nonhydrostatic model's basic variables excluding moisture.

### Pressure

$$\frac{\partial p'}{\partial t} - \rho_0 g w + \gamma p \nabla \cdot \mathbf{V} = -\mathbf{V} \cdot \nabla p' + \frac{\gamma p}{T} \left( \frac{\dot{Q}}{c_p} + \frac{T_0}{\theta_0} D_\theta \right) \quad (8.1)$$

**Momentum (x-component)**

$$\frac{\partial u}{\partial t} + \frac{m}{\rho} \left( \frac{\partial p'}{\partial x} - \frac{\sigma}{p^*} \frac{\partial p^*}{\partial x} \frac{\partial p'}{\partial \sigma} \right) = -\mathbf{V} \cdot \nabla u + v \left( f + u \frac{\partial m}{\partial y} - v \frac{\partial m}{\partial x} \right) - ew \cos \alpha - \frac{uw}{r_{earth}} + D_u \quad (8.2)$$

**Momentum (y-component)**

$$\frac{\partial v}{\partial t} + \frac{m}{\rho} \left( \frac{\partial p'}{\partial y} - \frac{\sigma}{p^*} \frac{\partial p^*}{\partial y} \frac{\partial p'}{\partial \sigma} \right) = -\mathbf{V} \cdot \nabla v - u \left( f + u \frac{\partial m}{\partial y} - v \frac{\partial m}{\partial x} \right) + ew \sin \alpha - \frac{vw}{r_{earth}} + D_v \quad (8.3)$$

**Momentum (z-component)**

$$\frac{\partial w}{\partial t} - \frac{\rho_0}{\rho} \frac{g}{p^*} \frac{\partial p'}{\partial \sigma} + \frac{gp'}{\gamma p} = -\mathbf{V} \cdot \nabla w + g \frac{p_0}{p} \frac{T'}{T_0} - \frac{gRd}{c_p} \frac{p'}{p} + e(u \cos \alpha - v \sin \alpha) + \frac{u^2 + v^2}{r_{earth}} + D_w \quad (8.4)$$

**Thermodynamics**

$$\frac{\partial T}{\partial t} = -\mathbf{V} \cdot \nabla T + \frac{1}{\rho c_p} \left( \frac{\partial p'}{\partial t} + \mathbf{V} \cdot \nabla p' - \rho_0 g w \right) + \frac{Q}{c_p} + \frac{T_0}{\theta_0} D_\theta \quad (8.5)$$

**Advection terms** can be expanded as

$$\mathbf{V} \cdot \nabla A \equiv mu \frac{\partial A}{\partial x} + mv \frac{\partial A}{\partial y} + \dot{\sigma} \frac{\partial A}{\partial \sigma} \quad (8.6)$$

where

$$\dot{\sigma} = -\frac{\rho_0 g}{p^*} w - \frac{m\sigma}{p^*} \frac{\partial p^*}{\partial x} u - \frac{m\sigma}{p^*} \frac{\partial p^*}{\partial y} v \quad (8.7)$$

**Divergence term** can be expanded as

$$\nabla \cdot \mathbf{V} = m^2 \frac{\partial}{\partial x} \left( \frac{u}{m} \right) - \frac{m\sigma}{p^*} \frac{\partial p^*}{\partial x} \frac{\partial u}{\partial \sigma} + m^2 \frac{\partial}{\partial y} \left( \frac{v}{m} \right) - \frac{m\sigma}{p^*} \frac{\partial p^*}{\partial y} \frac{\partial v}{\partial \sigma} - \frac{\rho_0 g}{p^*} \frac{\partial w}{\partial \sigma} \quad (8.8)$$

## Notes about the equations:

- Appendix A shows derivations of Equations 8.1, 8.4, 8.5 and 8.7, and shows the coordinate transformation from  $z$  to sigma coordinates.
- In the model, Equation 8.1 does not include the last term with parentheses on the right. This is neglected and it represents a pressure increase due to heating which forces the air to expand.
- Equations 8.2-8.4 include terms ( $eu$  and  $ew$ ) representing the usually neglected component of the Coriolis force, where  $e = 2\Omega \cos \lambda$ ,  $\alpha = \phi - \phi_c$ ,  $\lambda$  is latitude,  $\phi$  is longitude, and  $\phi_c$  is central longitude.
- The  $u \frac{\partial m}{\partial y}$ ,  $v \frac{\partial m}{\partial x}$  and  $r_{\text{earth}}$  terms represent curvature effects, and  $m$  is map-scale factor.
- Equations 8.2, 8.3 and 8.8 include terms to account for the sloped sigma surfaces when calculating horizontal gradients.
- Prognostic equations also exist for water vapor and microphysical variables such as cloud and precipitation (if used). These include the advection and various source/sink terms.

## Spatial finite differencing -

The above equations are finite differenced on the B grid mentioned in Chapter 1. Second-order centered finite differences represent the gradients except for the precipitation fall term which uses a first-order upstream scheme for positive definiteness. Often horizontal averaging is required to determine the gradient in the correct position. Vertical interpolations allow for the variable vertical grid size. More details are in Grell et al. (1994), NCAR Tech. Note 398.

## Temporal finite differencing -

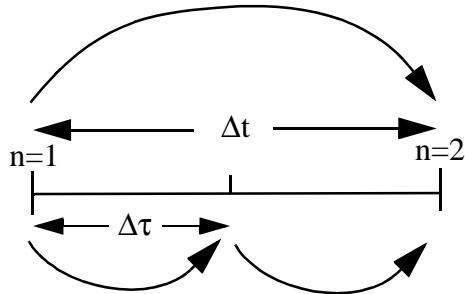
A second-order leapfrog time-step scheme is used for these equations, but some terms are handled using a time-splitting scheme. Note that Equations 8.1-8.4 contain extra terms on the left of the equals sign. This designates so-called fast terms that are responsible for sound waves that have to be calculated on a shorter time step. In the leapfrog scheme, the tendencies at time  $n$  are used to step the variables from time  $n-1$  to  $n+1$ . This is used for most of the right-hand terms (advection, coriolis, buoyancy). A forward step is used for diffusion and microphysics where the tendencies are calculated at time  $n-1$  and used to step the variables from  $n-1$  to  $n+1$ . Some radiation and cumulus options use a constant tendency over periods of many model timesteps and are only recalculated every 30 minutes or so.

However for certain terms the model timestep is too long for stability and these have to be predicted with a shorter step. Examples of this are the sound-wave terms shown in the equations, the precipitation fall term and the PBL tendencies which also may be split in certain situations. When the timestep is split, certain variables and tendencies are updated more frequently. For sound waves  $u$ ,  $v$ ,  $w$  and  $p'$  all need to be updated each short step using the tendency terms on the left of 8.1-8.4 while the terms on the right are kept fixed. For sound waves there are usually four of these steps between  $n-1$  and  $n+1$ , after which  $u$ ,  $v$ ,  $w$  and  $p'$  are up to date.

Certain processes are treated implicitly for numerical stability. An implicit time scheme is one in which the tendencies of variables depend not only on the present and past values, but also the

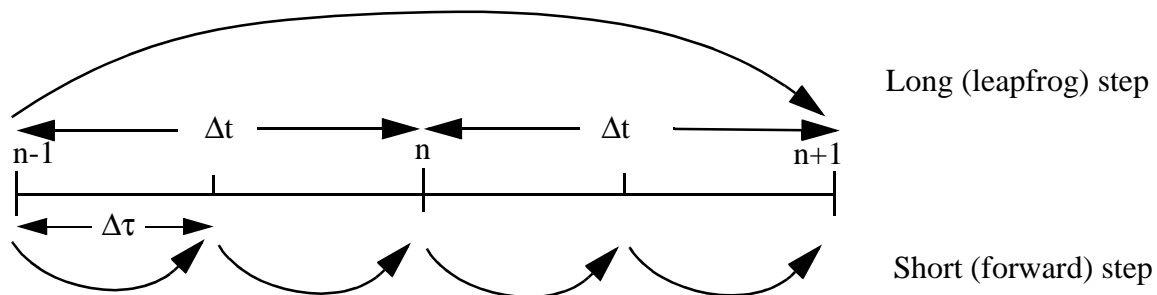
future values. These schemes are often numerically stable for all timesteps, but usually require a matrix inversion to implement them. In MM5 implicit schemes are used only in 1-d column calculations for vertical sound waves and vertical diffusion, so that the matrix is tridiagonal making it straightforward to solve directly.

*First time step:*



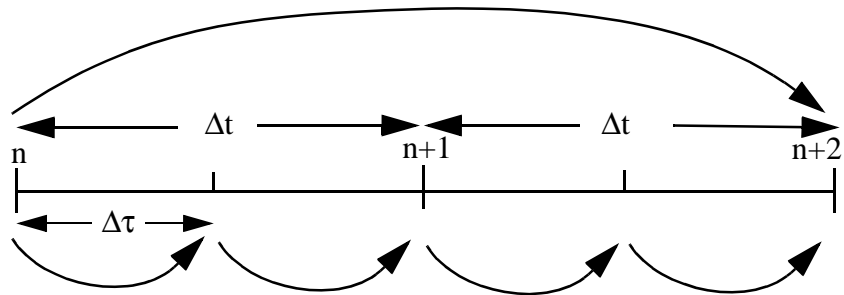
*Time step  $n$ :*

T, qv, qc, etc., advection, physics, boundary, coriolis, diffusion terms



u, v, w, p' advanced (pressure gradients, divergence terms)

*Time step  $n+1$ :*



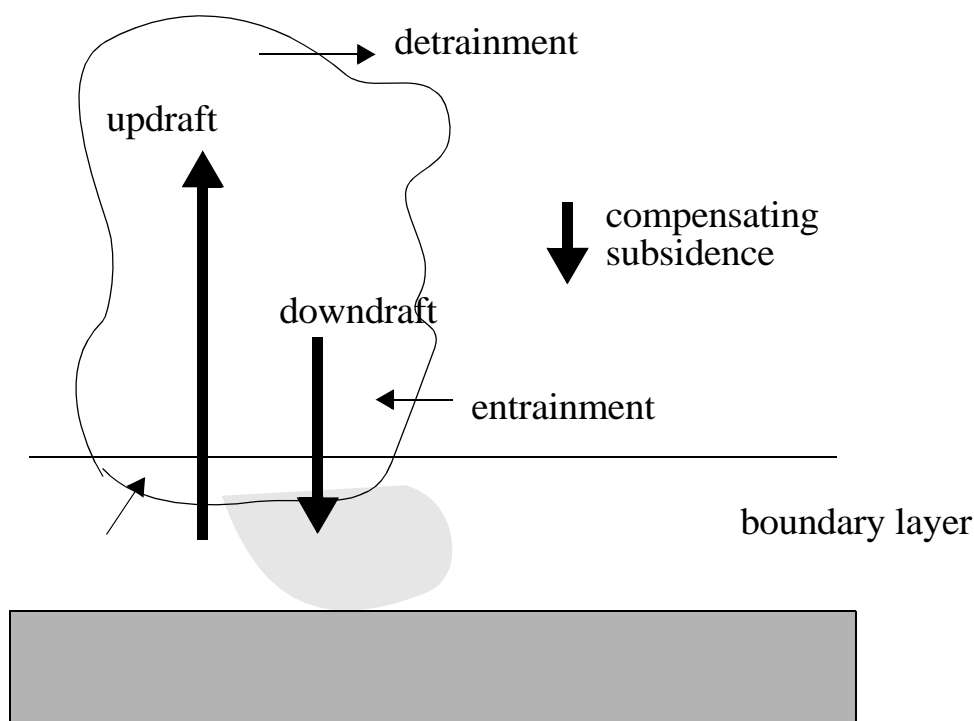
## 8.3 Physics Options in MM5

### 8.3.1 Cumulus Parameterizations (ICUPA)

#### 1. None -

Use no cumulus parametrization at grid sizes  $< 5\text{-}10$  km.

#### Illustration of Cumulus Processes



#### 2. Anthes-Kuo -

Based on moisture convergence, mostly applicable to larger grid sizes  $> 30$  km. Tends to produce much convective rainfall, less resolved-scale precip, specified heating profile, moistening dependent upon relative humidity.

#### 3. Grell -

Based on rate of destabilization or quasi-equilibrium, simple single-cloud scheme with updraft and downdraft fluxes and compensating motion determining heating/moistening profile. Useful for smaller grid sizes  $10\text{-}30$  km, tends to allow a balance between resolved scale rainfall and convective rainfall. Shear effects on precipitation efficiency are considered. See Grell et al. (1994).

#### 4. Arakawa-Schubert -

Multi-cloud scheme that is otherwise like Grell scheme. Based on a cloud population, allowing for entrainment into updrafts and allows for downdrafts. Suitable for larger scales,  $> 30$  km grid sizes, possibly expensive compared to other schemes. Shear effects on precipitation efficiency are considered. Also see Grell et al. (1994).

## **5. Fritsch-Chappell -**

Based on relaxation to a profile due to updraft, downdraft and subsidence region properties. The convective mass flux removes 50% of available buoyant energy in the relaxation time. Fixed entrainment rate. Suitable for 20-30 km scales due to single-cloud assumption and local subsidence. See Fritsch and Chappell (1980) for details. This scheme predicts both updraft and downdraft properties and also detrains cloud and precipitation. Shear effects on precipitation efficiency are also considered.

## **6. Kain-Fritsch -**

Similar to Fritsch-Chappell, but using a sophisticated cloud-mixing scheme to determine entrainment/detrainment, and removing all available buoyant energy in the relaxation time. See Kain and Fritsch (1993) for details. This scheme predicts both updraft and downdraft properties and also detrains cloud and precipitation. Shear effects on precipitation efficiency are also considered.

## **7. Betts-Miller -**

Based on relaxation adjustment to a reference post-convective thermodynamic profile over a given period. This scheme is suitable for  $> 30$  km, but no explicit downdraft, so may not be suitable for severe convection. See Betts (1986), Betts and Miller (1986), Betts and Miller (1993) and Janjic (1994) for details.

## **8. Kain-Fritsch 2 -**

A new version of Kain-Fritsch that includes shallow convection. This is similar to one that is being run in test mode in the Eta model (Kain 2002).

## **Shallow Cumulus - (ISHALLO=1)**

Handles non-precipitating clouds. Assumed to have strong entrainment and small radius, no downdrafts, and uniform clouds. Based on Grell and Arakawa-Schubert schemes. Equilibrium assumption between cloud strength and sub-grid (PBL) forcing.

## **8.3.2 PBL Schemes (IBLTYP) and Diffusion**

### **0. None -**

No surface layer, unrealistic in real-data simulations.

### **1. Bulk PBL -**

Suitable for coarse vertical resolution in boundary layer, e.g.  $> 250$  m vertical grid sizes. Two stability regimes.

### **2. High-resolution Blackadar PBL -**

Suitable for high resolution PBL, e.g. 5 layers in lowest km, surface layer  $< 100$  m thick. Four stability regimes, including free convective mixed layer. Uses split time steps for stability.

### **3. Burk-Thompson PBL -**

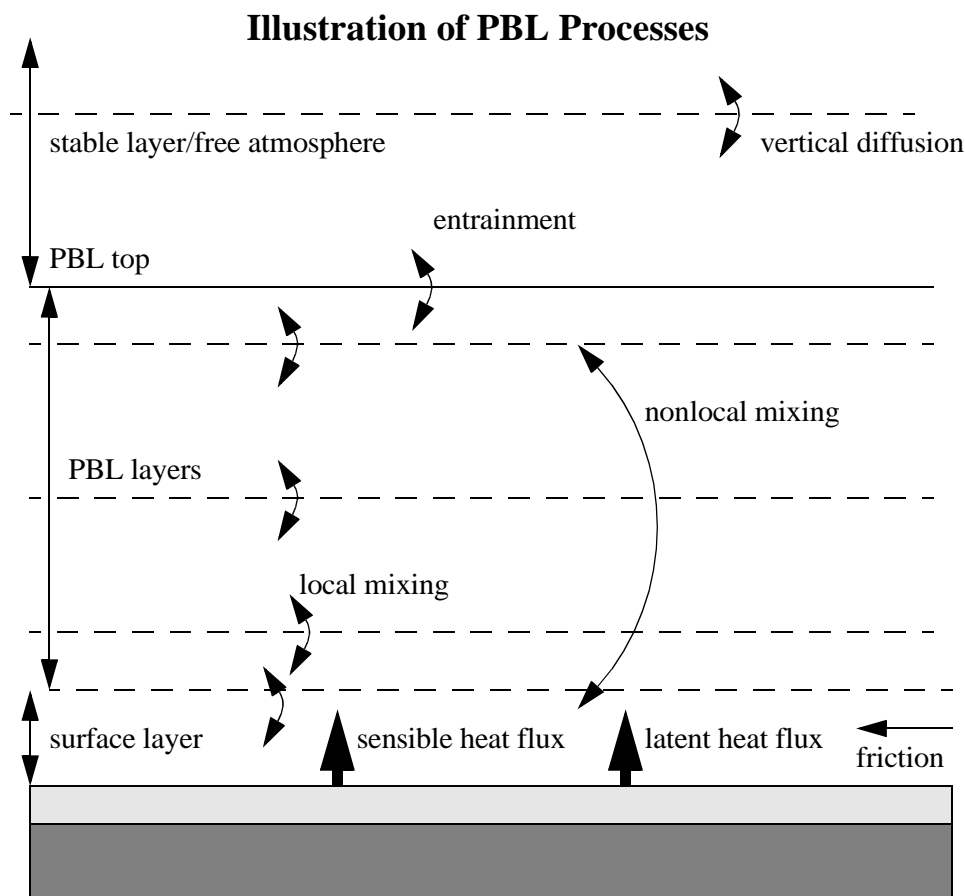
Suitable for coarse and high-resolution PBL. Predicts turbulent kinetic energy for use in vertical mixing, based on Mellor-Yamada formulas. See Burk and Thompson (1989) for details. This is the only PBL option that does not call the SLAB scheme, as it has its own force-restore ground



temperature prediction

#### 4. Eta PBL -

This is the Mellor-Yamada scheme as used in the Eta model, Janjic (1990, MWR) and Janjic (1994, MWR). It predicts TKE and has local vertical mixing. The scheme calls the SLAB routine or the LSM for surface temperature and has to use ISOIL=1 or 2 (not 0) because of its long time step. Its cost is between the MRFPBL and HIRPBL schemes. Before SLAB or the LSM the scheme calculates exchange coefficients using similarity theory, and after SLAB/LSM it calculates vertical fluxes with an implicit diffusion scheme.



#### 5. MRF PBL -

or Hong-Pan PBL, suitable for high-resolution in PBL (as for Blackadar scheme). Efficient scheme based on Troen-Mahrt representation of countergradient term and K profile in the well mixed PBL, as implemented in the NCEP MRF model. See Hong and Pan (1996) for details. This scheme either calls the SLAB routine or the LSM and should have ISOIL=1 or 2. Vertical diffusion uses an implicit scheme to allow longer time steps.

#### 6. Gayno-Seaman PBL -

This is also based on Mellor-Yamada TKE prediction. It is distinguished from others by the use of liquid-water potential temperature as a conserved variable, allowing the PBL to operate more accurately in saturated conditions (Ballard et al., 1991; Shafran et al. 2000). Its cost is comparable with the Blackadar scheme's because it uses split time steps.

## 7. Pleim-Chang PBL -

This scheme only works with ISOIL=3 (see later). The PBL scheme is a derivative of the Blackadar PBL scheme called the Asymmetric Convective Model (Pleim and Chang, 1992, *Atm. Env.*), using a variation on Blackadar's non-local vertical mixing.

### Moist vertical diffusion - (IMVDIF=1)

IBLTYP = 2, 5 and 7 have this option. It allows diffusion in cloudy air to mix toward a moist adiabat by basing its mixing on moist stability instead of the dry stability. From Version 3.5 it can mix cloudy air upwards into clear air in addition to just internally in cloudy layers.

### Thermal roughness length - (IZOTOPT=0,1,2)

IBLTYP = 2 and 5 have the option of using a different roughness length for heat/moisture than that used for momentum. This is the thermal roughness length. IZOTOPT=0 is the default (old) scheme, IZOTOPT=1 is the Garratt formulation, and IZOTOPT=2 is the Zilitinkevich formulation (used by the Eta model). Changing the thermal roughness length affects the partitioning of sensible and latent heat fluxes, and affects the total flux over water.

### Horizontal diffusion - (ITPDIF=0,1,2)

ITPDIF=0,1 are two methods of doing horizontal temperature diffusion. ITPDIF=1 (default) is to only horizontally diffuse the perturbation from the base-state temperature. This partially offsets the effect of the coordinate slope over topography which is needed due to the fact that the diffusion is along model levels. ITPDIF=0 diffuses the full temperature (like all other fields) instead. In Version 3.7, a new option is ITPDIF=2. This applies to temperature, moisture and cloud water, and is a purely horizontal diffusion accounting more accurately for coordinate slope and valley walls (Zangl, 2002 MWR)..

## 8.3.3 Explicit Moisture Schemes (IMPHYS)

### 1. Dry -

No moisture prediction. Zero water vapor.

### 2. Stable Precip -

Nonconvective precipitation. Large scale saturation removed and rained out immediately. No rain evaporation or explicit cloud prediction.

### 3. Warm Rain -

Cloud and rain water fields predicted explicitly with microphysical processes. No ice phase processes.

### 4. Simple Ice (Dudhia) -

Adds ice phase processes to above without adding memory. No supercooled water and immediate melting of snow below freezing level. This also can be run with a look-up table (MPHYSTBL=1) version for efficiency.

### 5. Mixed-Phase (Reisner 1) -

Adds supercooled water to above and allows for slow melting of snow. Memory added for cloud ice and snow. No graupel or riming processes. See Reisner et al. (1998) for details. Since version 3.7 an optimized version of this code is also available (MPHYSTBL=2). This also can be run with

a look-up table (MPHYSTBL=1) version for efficiency.

### **6. Goddard microphysics -**

Includes additional equation for prediction of graupel. Suitable for cloud-resolving models. See Lin et al. (JCAM, 1983), Tao et al. (1989, 1993) for details. Scheme was updated for Version 3.5 to include graupel or hail properties.

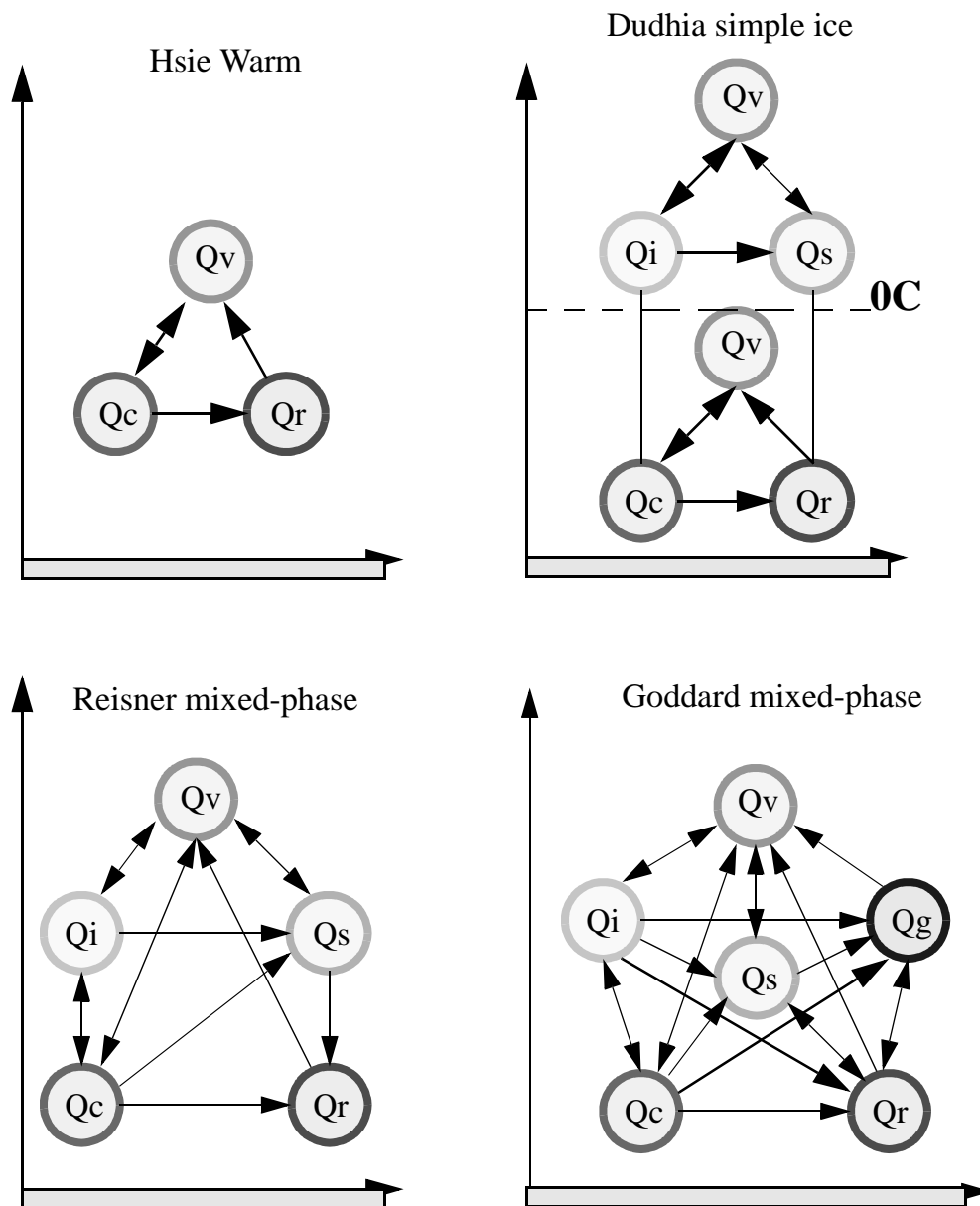
### **7. Reisner graupel (Reisner 2) -**

Based on mixed-phase scheme but adding graupel and ice number concentration prediction equations. Also suitable for cloud-resolving models. Scheme was updated significantly between Version 3.4 and 3.5, and again between 3.5 and 3.6. 3.6 also has a capability for calling the scheme less frequently than every time-step, but this is not standard and requires code editing to implement (Web pages will show the procedure).

### **8. Schultz microphysics -**

A highly efficient and simplified scheme (based on Schultz 1995 with some further changes), designed for running fast and being easy to tune for real-time forecast systems. It contains ice and graupel/hail processes.

## Illustration of Microphysics Processes



---

### 8.3.4 Radiation Schemes (IFRAD)

#### 0. None -

No mean tendency applied to atmospheric temperature, unrealistic in long-term simulations.

#### 1. Simple cooling -

Atmospheric cooling rate depends just on temperature. No cloud interaction or diurnal cycle.

#### 0 or 1. Surface radiation -

This is used with the above two options. It provides diurnally varying shortwave and longwave flux at the surface for use in the ground energy budget. These fluxes are calculated based on atmospheric column-integrated water vapor and low/middle/high cloud fraction estimated from relative humidity.

#### 2. Cloud-radiation scheme -

Sophisticated enough to account for longwave and shortwave interactions with explicit cloud and clear-air. As well as atmospheric temperature tendencies, this provides surface radiation fluxes. May be expensive but little memory requirement. In Version 3.7 namelist switches LEVSLP and OROSHAW can be used with this option. LEVSLP enables slope effects on solar radiation, and OROSHAW allows shadowing effects on nearby grid-cells.

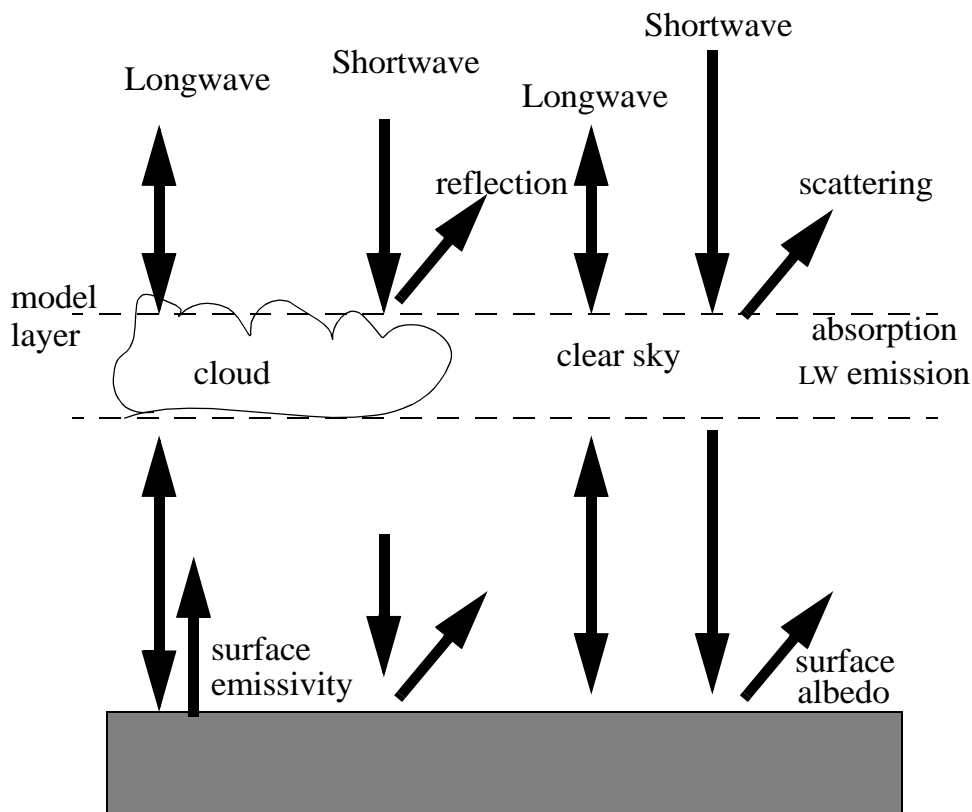
#### 3. CCM2 radiation scheme -

Multiple spectral bands in shortwave and longwave, but cloud treated based on either resolved clouds (ICLOUD=1) or RH-derived cloud fraction (ICLOUD=2). Suitable for larger grid scales, and probably more accurate for long time integrations. Also provides radiative fluxes at surface. See Hack et al. (1993) for details. As with other radiation schemes ICLOUD=0 can be used to remove cloud effects on the radiation. Up until Version 3.5, this scheme was only able to interact with RH-derived clouds.

#### 4. RRTM longwave scheme -

This is combined with the cloud-radiation shortwave scheme when IFRAD=4 is chosen. This longwave scheme is a new highly accurate and efficient method provided by AER Inc. (Mlawer et al. 1997). It is the Rapid Radiative Transfer Model and uses a correlated-k model to represent the effects of the detailed absorption spectrum taking into account water vapor, carbon dioxide and ozone. It is implemented in MM5 to also interact with the model cloud and precipitation fields in a similar way to IFRAD=2.

### Illustration of Free Atmosphere Radiation Processes



### 8.3.5 Surface Schemes (ISOIL)

#### None - (ITGFLG=3)

No ground temperature prediction. Fixed surface temperature, not realistic.

#### 0. Force/restore (Blackadar) scheme -

Single slab and fixed-temperature substrate. Slab temperature based on energy budget and depth assumed to represent depth of diurnal temperature variation (~ 10-20 cm).

#### 1. Five-Layer Soil model -

Temperature predicted in 1,2,4,8,16 cm layers (approx.) with fixed substrate below using vertical diffusion equation. Thermal inertia same as force/restore scheme, but vertically resolves diurnal temperature variation allowing for more rapid response of surface temperature. See Dudhia (1996 MM5 workshop abstracts) for details. Cannot be used with Burk-Thompson PBL (IBLTYP=3).

#### 2. Noah Land-Surface Model -

[Note: this was the OSU LSM until MM5 Version 3.5, and from 3.6 it is updated and renamed as the Noah LSM, a unified model between NCAR, NCEP and AFWA].

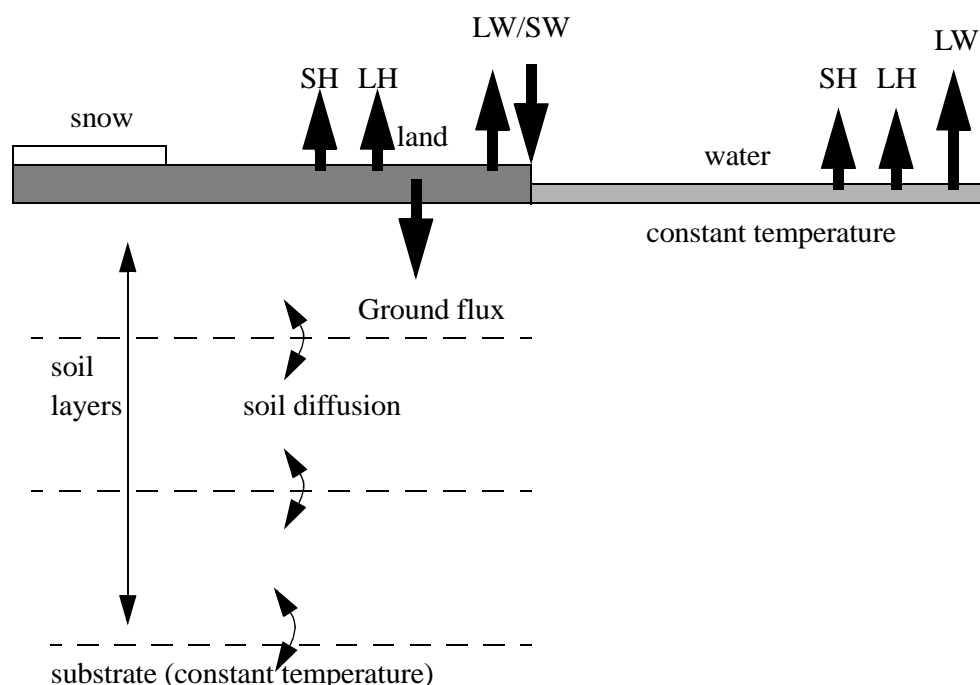
The land-surface model is capable of predicting soil moisture and temperature in four layers (10,

30, 60 and 100 cm thick), as well as canopy moisture and water-equivalent snow depth. It also outputs surface and underground run-off accumulations. The LSM makes use of vegetation and soil type in handling evapotranspiration, and has effects such as soil conductivity and gravitational flux of moisture. In MM5 it may be called instead of the SLAB model in the MRF and Eta PBL schemes, taking surface-layer exchange coefficients as input along with radiative forcing, and precipitation rate, and outputting the surface fluxes for the PBL scheme. This scheme uses a diagnostic equation to obtain a skin temperature, and the exchange coefficients have to allow for this by use of a suitable molecular diffusivity layer to act as a resistance to heat transfer. See Chen and Dudhia (2001). It also handles sea-ice surfaces. All the aforementioned processes were in the OSU LSM. The Noah LSM has some modifications, and additional processes to better handle snow cover, predict physical snow depth, and frozen soil effects. In addition to soil moisture, soil water is a separate 4-layer variable, and soil moisture is taken to be the total of soil water and soil ice. Physical snow height is also diagnosed and output. The Noah LSM can also optionally use satellite-derived climatological albedo, supplied by REGRID, instead of relating albedo to land-use type. See Appendix D for practical guidance on setting up the modeling system to use the LSM.

### 3. Pleim-Xiu Land-Surface Model

This is coupled to the Pleim-Xiu PBL (IBLTYP=7). It is a combined land-surface and PBL model. It represents soil moisture and temperature in two layers (surface layer at 1 cm, and root zone at 1 m) as well as canopy moisture. It handles soil surface, canopy and evapotranspiration moisture fluxes. It also makes use of percentage land-use and soil data from Terrain to aggregate soil and vegetation properties, rather than using a single dominant type. Soil moisture can be initialized from land-use moisture availability, a soil moisture input grid (as with the Noah LSM), or via nudging using model minus observed surface temperature error to correct soil moisture. The model also has optional plant-growth and leaf-out algorithms making it suitable for long-term simulations. See Xiu and Pleim (2000).

#### Illustration of Surface Processes



**Bucket Soil Moisture Model - (IMOIAV=1,2)**

This can be run with ISOIL=0 or 1. It keeps a budget of soil moisture allowing moisture availability to vary with time, particularly in response to rainfall and evaporation rates. The soil moisture can be initialized from land-use type and season (LANDUSE.TBL) as before (IMOIAV=1), or a 10-cm soil moisture input as with the Noah LSM (IMOIAV=2).

**Snow Cover Model - (IFSNOW=0,1,2)**

When the LSM is not used this switch determines how snow cover is handled. IFSNOW=0 means snow cover is ignored. IFSNOW=1 uses the input snow-cover (0/1) flag to determine the land-surface properties such as albedo and soil moisture. These stay fixed in the simulation. Since Version 3.5 there is an option (IFSNOW=2) to predict snow cover using an input water-equivalent snow depth. It updates water-equivalent snow depth according to a heat and moisture budget in the SLAB routine, and accumulates snow from the microphysical schemes (currently IMPHYS=4,5, or 7). In Version 3.7 this can be used with IMOIAV=1 or 2, the bucket soil moisture.

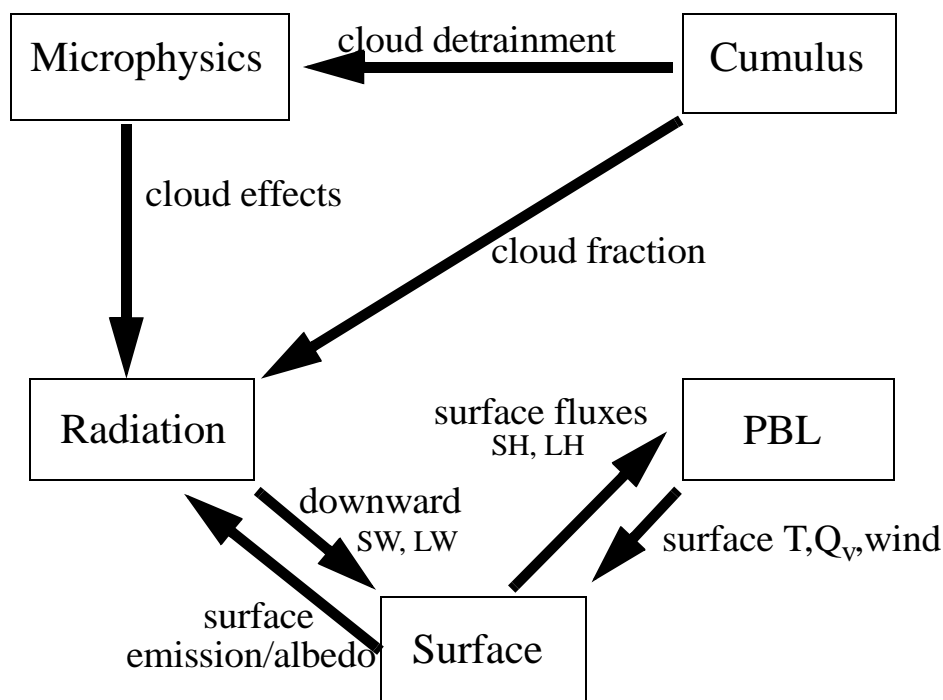
**Polar Mods - (IPOLAR=1)**

The so-called Polar Mods were developed by the Byrd Polar Research Center at Ohio State University to better handle Antarctic conditions for forecasting purposes. IPOLAR=1 is a compile-time option, therefore it is in the configure.user file. The use of the Polar Mods has several effects, and should be applied only with ISOIL=1. The main changes are (i) to increase the number of prognostic soil layers from 5 to 7, and (ii) to allow for sea-ice fraction effects on the heat and moisture fluxes and mean ground temperature. Sea-ice fraction can either be diagnosed from sea-surface temperature (IEXSI=1), or read in from a dataset (IEXSI=2). It is also recommended that the Eta PBL is used with this option, as that has been modified to account for ice-surface fluxes. The soil model is modified to account for snow and ice properties for heat conduction. The Polar Mods also slightly modify the Simple Ice and Reisner 1 microphysics schemes to use the Meyers formula for ice number concentration. In release 3.7 the MRF PBL also has modifications to work with this option.



## 8.4 Interactions of Parameterizations

### Direct Interactions of Parameterizations



## 8.5 Boundary conditions

### 8.5.1 Lateral boundary conditions (IBOUDY)

#### 0. Fixed -

This will not allow time variation at lateral boundaries. Not recommended for real-data applications.

#### 2. Time-dependent/Nest -

Outer two rows and columns have specified values of all predicted fields. Recommended for nests where time-dependent values are supplied by the parent domain. Not recommended for coarse mesh where only one outer row and column would be specified.

#### 3. Relaxation/inflow-outflow -

Outer row and column is specified by time-dependent value, next four points are relaxed towards the boundary values with a relaxation constant that decreases linearly away from the boundary. Recommended for coarse mesh where boundary values are supplied by the BDYOUT\_DOMAIN1 file. Fields without boundary values (such as some moisture variables) are specified zero on inflow and zero-gradient on outflow boundaries.

## 8.5.2 Lower boundary conditions

The LOWBDY\_DOMAINx file provides sea-surface temperature, substrate temperature, and optionally snow cover and sea-ice. The switch ISSTVAR allows multiple times in this file (created by INTERPF) to be read in as the model runs, which is the method of updating these fields in long-term simulations.

## 8.5.3 Upper boundary condition (IFUPR)

### 0. No upper boundary condition -

Rigid lid with no vertical motion at the model top. This may be preferable for very coarse mesh simulations (50 km or more grids).

### 1. Upper radiative condition -

Top vertical motion calculated to reduce reflection of energy from the model top preventing some spurious noise or energy build-up over topography. This is recommended for grid-lengths below 50 km. It works better for hydrostatic gravity wave scales, rather than inertial or nonhydrostatic scales.

## 8.6 Nesting

### 8.6.1 One-way nesting

When a single-domain or multiple-domain run completes, its domain output can be put into NESTDOWN to create an input file with higher resolution (any integer ratio in dx) and new lateral and lower boundary files. See NESTDOWN chapter. NESTDOWN allows the addition of higher resolution elevation and land-use data. This is known as a one-way nest because it is forced purely by the coarse mesh boundaries, and obviously has no feedback on the coarse-mesh run.

When INTERPB becomes available, it will be possible to put model output on pressure levels and reanalyze with observations as well as choosing different vertical levels for the nest by using INTERPF and NESTDOWN.

### 8.6.2 Two-way nesting

Multiple domains can be run in MM5 at the same time. Up to nine domains on four levels of nest are allowed with each nest level one third of its parent domain's grid-length. Each domain takes information from its parent domain every timestep, and runs three timesteps for each parent step before feeding back information to the parent domain on the coincident interior points. Figure 1.3 illustrates the staggering with the 3:1 ratio. The feedback distinguishes two-way nesting from one-way nesting, and allows nests to affect the coarse mesh solution, usually leading to better behavior at outflow boundaries. However there is significant overhead cost associated with the boundary interpolation and feedback at every timestep, particularly with distributed-memory machines.

### 8.6.3 Two-way nest initialization options (IOVERW)

IOVERW is the overwrite switch that determines whether a nested input file is used to replace coarse mesh information or whether the coarse domain is just interpolated to start the nest.

## **0. Interpolation -**

No nested input file is required. All the information including topography is interpolated from the coarse mesh to start the nest. This is suitable for nests that start later than the coarse mesh or for moving and overlapping nests. This could be used in situations where improved topography is not essential such as over water or smooth terrain.

## **1. Nest input file -**

This requires an MMINPUT file to be read in for the nest. The input file contains all the meteorological and terrain fields at a higher resolution, and so may provide a more accurate initial analysis. This should only be applied when the coarse mesh and nest both start at the same time, because an analysis at a later time is unlikely to match the coarse-mesh boundary conditions.

## **2. Terrain input file -**

This only requires the TERRAIN file for the nest. The meteorological fields are interpolated from the coarse mesh, but the terrain and land-use are replaced with the higher resolution fields from TERRAIN. A vertical adjustment is carried out to put the interpolated fields on terrain-following levels consistent with the new nest terrain. This has the benefit of allowing fine-topography nests to start later than the coarse mesh.

### **8.6.4 Two-way nesting feedback options (IFEED)**

These options determine how a nest feeds back its interior information to its parent domain.

#### **0. No feedback -**

Feedback is turned off, similar to one-way nests except boundary conditions are updated by parent domain every timestep. Not recommended except for tests.

#### **1. 9-point weighted average -**

Feedback uses a weighted average of nest points onto coarse mesh point, not just coincident value. Not the primary recommended choice because terrain elevation is not consistent with this feedback.

#### **2. 1-point feedback with no smoothing -**

Coincident point is fed back. Not recommended except for tests.

#### **3. 1-point feedback with smoother-desmoothing -**

Coincident point is fed back, and coarse mesh fields are then filtered using smoother-desmoothing to remove two-grid-length noise. Recommended option.

#### **4. 1-point feedback with heavy smoothing -**

Coincident point is fed back, and coarse mesh fields are then smoothed with a 1-2-1 smoother that removes two-grid-length noise, and damps other short wavelengths strongly. Could be used if nest region appears excessively noisy when viewing coarse mesh output.

## 8.7 Four-Dimensional Data Assimilation (FDDA)

### 8.7.1 Introduction

FDDA is a method of running a full-physics model while incorporating observations. Thus the model equations assure a dynamical consistency while the observations keep the model close to the true conditions and make up for errors and gaps in the initial analysis and deficiencies in model physics. The MM5 model uses the Newtonian-relaxation or nudging technique.

### 8.7.2 FDDA Method

There are two distinct nudging methods. The model can use these individually or combined.

#### **Analysis or Grid Nudging -**

Newtonian relaxation terms are added to the prognostic equations for wind, temperature, and water vapor. These terms relax the model value towards a given analysis. The technique is implemented by obtaining analyses on the model grid over the data assimilation period and these are fed to the model in its standard input format. The model linearly interpolates the analyses in time to determine the value towards which the model relaxes its solution. The user defines the time scale of the relaxation constants for each variable.

#### **Station or Observational Nudging -**

In situations where analysis-nudging is not practical, such as at high resolution or with asynoptic data, obs-nudging is a useful alternative. This method again uses relaxation terms, but the method is similar to objective analysis techniques where the relaxation term is based on the model error at observational stations. The relaxation is such as to reduce this error. Each observation has a radius of influence, a time window and a relaxation time scale to determine where, when and how much it affects the model solution. Typical model grid points may be within the radius of influence of several observations and their contributions are weighted according to distance. To implement this method an observation input file is required that chronologically lists the 3D positions and values of each observation in a specific format.

### 8.7.3 Uses of FDDA

Four-Dimensional Data Assimilation has three basic uses -

- **Dynamic Initialization:** Data assimilation by the above methods is applied during a pre-forecast time period for which additional observations or analyses exist. Then the nudging terms switch off as the forecast begins. This has two advantages over the standard static initialization, (i) It can make use of asynoptic data during the pre-forecast period and generally contains more observational information at the forecast start time, and (ii) There is a reduced spin-up or shock effect at the forecast start owing to the better balance of the initial model conditions.
- **Dynamic Analysis:** This is the same as dynamic initialization except that the intent is to produce a four-dimensionally consistent analysis taking into account dynamical balances that are provided by the model and observations that are introduced by nudging. This analysis may be used to initialize higher-resolution simulations or for kinematic studies such as chemical and tracer transports.

- **Boundary Conditions:** By using data assimilation on the coarse mesh and nesting with a finer mesh, the fine mesh is provided with superior boundary conditions compared to the standard linear interpolation of analyses, because the boundaries have a much higher time resolution of features passing through them into the fine mesh.

Note: For scientific case studies and forecasts the model should have no data assimilation terms as these represent non-physical terms in the equations.

## 8.7.4 Data used in FDDA

### Analysis nudging -

When doing three-dimensional analysis nudging, no additional input data files are required. MM5 can use the same MMINPUT file or a copy of MMINPUT to MMINPUT2 file. If surface FDDA is desired, a user must set F4D = TRUE in the namelist of RAWINS job deck, which enables the job to create (typically) a 3-hourly surface analysis file to be used in MM5. FDDA now works with all the boundary layer options except 0, 1, and 3. It needs information on the boundary-layer top from these schemes.

### Station nudging -

There is no standard software available to create input data file for observational nudging. The input file is a binary file containing 9 real numbers per record and in order of increasing time. The READ statement in the model is the following:

```
READ (NVOL,END=111) TIMEOB,RIO,RJO,RKO,(VAROBS(IVAR),IVAR=1,5)
```

where NVOL is the input fortran unit number, and

TIMEOB:	Julian date in dddhh. Example: 16623.5 - Julian day 166 and hour 2330 UTC
RIO:	y-location - I dot-point location on coarse mesh (may be a fraction of a grid)
RJO:	x-location - J dot-point location on coarse mesh (may be a fraction of a grid)
RKO:	z-location - K half- $\sigma$ level (must be on half $\sigma$ levels)
IVAR(1):	u wind - in m/sec rotated to model grid
IVAR(2):	v wind - in m/sec rotated to model grid
IVAR(3):	temperature - in Kelvin
IVAR(4):	water vapor mixing ratio - in kg/kg
IVAR(5):	Pstar - in cb (only used in hydrostatic model)

A user may include more information at the end of a record which are not read by the model but can be used to identify the station and data type. The no-data value is 99999.. If running the model in nonhydrostatic mode, 99999. can be used to fill up the Pstar spot.

## 8.8 How to run MM5

Get the source code. The current MM5 release resides on NCAR's anonymous ftp site, `ftp.ucar.edu:mesouser/MM5V3/MM5.TAR.gz`. You may download `MM5.TAR.gz` to your working directory from the web page, `ftp://ftp.ucar.edu/mesouser/MM5V3`. Or you can copy it from `~mesouser/MM5V3/MM5.TAR.gz` on NCAR's SCD machines.

There are 2 steps to compiling and running the MM5 system:

- Choosing compilation options and compiling the code.
- Modifying the run-time options and executing the program.

### 8.8.1 Compiling MM5

- Edit the file "*configure.user*"
- Type '*make*'

(see 8.8.3 for running batch job on NCAR's IBM.)

The user chooses those compilation options appropriate to his/her system by editing the "*configure.user*" file. This file is included in every Makefile used in compiling the model so it contains many rules, but the user need only concern with 3 things:

- Find the section of compilation options appropriate for your machine. Uncomment the *RUNTIME\_SYSTEM* variable and the compiler options.
- Make sure that the general utilities required in a UNIX environment for compilation are available and appropriate. For example, there are many versions of the program "make" - if yours has special quirks and/or options, this would be the place to indicate them.
- Set model options in sections 5 and 6 of *configure.user*. These are used to set up domain sizes, 4DDA and physics option for (selective) compilation purposes.

If you wish to compile and run the model on a distributed-memory machine (such as IBM SP2, Cray T3E, SGI Origin 2000 with MPI, and Linux clusters),

- obtain additional tar file, *MPP.TAR.gz*, gunzip and then untar the file in the MM5 top directory;
- edit the *configure.user* file, and select and uncomment the appropriate *RUNTIME\_SYSTEM* and compiler flags;
- type '*make mpp*' to make an executable.

More information is provided for this topic in *README.MPP* in the MM5 tar file, Appendix D in this document, and on Web page: <http://www.mmm.ucar.edu/mm5/mpp.html>

### 8.8.2 Running MM5

- create the "*mm5.deck*" script by typing '*make mm5.deck*' - need to set *RUNTIME\_SYSTEM* correctly to get the right deck.
- edit the *mm5.deck* script to set appropriate namelist values
- run the "*mm5.deck*" script by typing '*mm5.deck*'.

**Basic Run:**

Need to set at least these namelist variables in *mm5.deck*:  
TIMAX, TISTEP, TAPFRQ, NESTIX, NESTJX, NESTI, NESTJ

**Restart Run:**

In addition to above namelist variables, set IFREST = .TRUE., and IXTIMR = restart time (can be found at the end of the *mm5.print.out* file from the previous run).

**One-Way Run:**

Should treat a one-way run in exact manner as if it is a basic run.

**8.8.3 Running MM5 Batch Job on NCAR's IBM**

- If you want to work in batch mode, whether to compile and/or execute, get a copy of *mm5.deck.ibm* from mesouser directory: *~mesouser/MM5V3/IBM* on NCAR's blackforest/babyblue/bluesky. Or, you may get the deck once you obtain the *MM5.TAR.gz* file on your local machine. To do so, first unzip and untar the tar file, edit the *configure.user* file to define *RUNTIME\_SYSTEM*="sp2"; then type '*make mm5.deck*'. This deck has the relevant part of *configure.user* file inside the deck. This deck is designed to be used for both interactive and batch mode.
- If you would like to compile interactively on a IBM, you can either use the above deck, or use the IBM interactive deck, by setting the *RUNTIME\_SYSTEM*="IBM", and followed by typing '*make mm5.deck*'. The *mm5.deck* generated this way has an appearance of other workstations decks. Compiling on IBM is similar to what one does on all other workstations.
- When you use the interactive deck to compile, you will still need to use the batch deck to submit a batch job for executing. Before you submit the batch job, remember to tar up your entire directory structure, and save it to some place (whether it is NCAR's MSS, or your local archive). Your batch job needs to access this tar file (default name *mm5exe.tar*) for executing.

**Note:** The *mmlif* (namelist file) for running MM5 is now generated from both your *configure.user* file (section 6 of the *configure.user*) and *mm5.deck*.

**8.8.4 Useful make commands****make clean**

This removes all generated files and returns the code to its original state. Use it before doing recompilation.

**make code**

This creates \*.f files from \*.F files and places them in directory *pick/*. Useful for looking at code in a single directory. All files related to options selected in *configure.user* file will be created.

## 8.9 Input to MM5

Files from INTERPF program for a basic run:

- Model initial condition file(s): *MMINPUT\_DOMAINx* (*MMINPUT\_DOMAIN2*, 3.. are optional)
- Lateral and lower boundary condition files for the coarsest domain: *BDYOUT\_DOMAIN1*, *LOWBDY\_DOMAINx* (*LOWBDY\_DOMAIN2*, 3, ..., are optional. The model will use them if they are present).
- Nest terrain file(s) from program TERRAIN: *TERRAIN\_DOMAIN2*, 3, etc. if using *IOVERW* = 2 option.

Files from MM5 program, if it is a restart run:

- Model save file(s) from previous run: rename *SAVE\_DOMAINx* to *RESTART\_DOMAINx*

Files from *RAWINS/LITTLE\_R*, if running gridded 4DDA option with surface analysis

- FDDA surface analysis: *SFCFDDA\_DOMAINx*

Files generated by user, if running observational nudging option

- FDDA 4D obs file(s): *MM5OBS\_DOMAINx*

*mm5lif*: a namelists file containing user-specified options; created when *mm5.deck* is executed.

*LANDUSE.TBL*: user-modifiable landuse characteristics (in ASCII), provided.

*RRTM\_DATA*: RRTM radiation scheme data file, provided.

*BUCKET.TBL*: user-modifiable constants used in bucket soil moisture model, provided.

*VEGPARM.TBL*: user-modifiable constants used in Noah LSM, provided.

*SOILPARM.TBL*: user-modifiable constants used in Noah LSM, provided.

*GENPARM.TBL*: user-modifiable constants used in Noah LSM, provided.

Note that the workstation *mm5.deck* expects all input files (named as above) to be present in the *Run/* directory. See the *mm5.deck* for details.

## 8.10 Output from MM5

A number of files are written out during MM5 integration. These are

- history files (*MMOUT\_DOMAINx*), if *IFTAPE* = 1, and the output frequency is set by *TAPFRQ* (and *INCTAP*).
- restart files (*SAVE\_DOMAINx*), if *IFSAVE* = *.TRUE.*, and the output frequency is set by *SAVFRQ*.

Output from each domain will be written to different files. For example, domain 1's history file is written to *MMOUT\_DOMAIN1*, and its restart file to *SAVE\_DOMAIN1*. Each output file contains data for all output times for that domain. On NCAR's IBMs, we recommend that one uses *BUF-FRQ* to limit output file sizes not exceeding 6,000 Mb (which is the file size limit for MSS).

For each time period the model history output includes:



- A general header record describing the model configuration
- A subheader describing the field following, and the field. This is repeated for all fields in a output.

3D forecast fields dimensioned by (IX, JX, KX or KX+1) for that domain include (note that the variables are NO LONGER coupled in Version 3):

1	U: U-wind (m/s)	dot
2	V: V-wind (m/s)	dot
3	T: Temperature (K)	cross
4	Q: Water vapor mixing ratio (kg/kg) (if IMPHYS $\geq$ 2)	cross
5	CLW: Cloud water mixing ratio (kg/kg) (if IMPHYS $\geq$ 3)	cross
6	RNW: Rain water mixing ratio (kg/kg) (if IMPHYS $\geq$ 3)	cross
7	ICE: Ice cloud mixing ratio (kg/kg) (if IMPHYS $\geq$ 5)	cross
8	SNOW: Snow mixing ratio (kg/kg) (if IMPHYS $\geq$ 5)	cross
9	GRAUPEL: Graupel (kg/kg) (if IMPHYS $\geq$ 6)	cross
10	NCI: Number concentration of ice (if IMPHYS=7)	cross
11	TKE: Turbulent k.e. (J/kg) (if IBLTYP=3,4,6)	cross
12	RAD TEND: Atmospheric radiation tendency (K/day) (if FRAD $\geq$ 2)	cross
13	W: Vertical velocity (m/s) ( <i>on full <math>\sigma</math> -levels</i> )	cross
14	PP: Perturbation pressure (Pa)	cross

2D forecast fields dimensioned (IX, JX) include:

1	PSTARCRS: Pstar (cb)	cross
2	GROUND T: Ground temperature (K)	cross
3	RAIN CON: Accum. convective rainfall (cm)	cross
4	RAIN NON: Accum. nonconv. rainfall (cm)	cross
5	PBL HGT: PBL height (m)	cross
6	REGIME: PBL regime (category, 1-4)	cross
7	SHFLUX: Surface sensible heat flux (W/m <sup>2</sup> )	cross
8	LHFLUX: Surface latent heat flux (W/m <sup>2</sup> )	cross
9	UST: Frictional velocity (m/s)	cross
10	SWDOWN: Surface downward shortwave radiation (W/m <sup>2</sup> )	cross
11	LWDOWN: Surface downward longwave radiation (W/m <sup>2</sup> )	cross
12	MAVAIL: Surface moisture availability (if IMOIAV=1,2)	cross
13	SOIL T x: Soil temperature in a few layers (K) (if ISOIL=1,2)	cross
14	SOIL M x: Soil moisture in a few layers (m <sup>3</sup> /m <sup>3</sup> ) (if ISOIL=2)	cross
15	SOIL W x: Soil water in a few layers (m <sup>3</sup> /m <sup>3</sup> ) (if ISOIL=2)	cross
16	SFCRNOFF: Surface runoff (mm) (if ISOIL=2)	cross
17	UGDRNOFF: Underground runoff (mm) (if ISOIL=2)	cross
18	SNOWCOVR: Snow cover (variable if ISOIL=2)	cross
19	SNOWH: physical snow height (m)	cross

	(if ISOIL=2, or IFSNOW=2)	
20	WEASD: Water-equivalent snow depth (mm) (if ISOIL=2, or IFSNOW=2)	cross
21	CANOPYM: Canopy moisture (m) (if ISOIL=2)	cross
22	GRNFLX: ground head flux (W/m2) (if ISOIL=2, 3)	cross
23	ALB: albedo (fraction) (if ISOIL=2)	cross
24	ALBSNOMX: maximum snow albedo (%) (if ISOIL=2, and RDMAXALB=T)	cross
25	MONALBnn: monthly albedo (%) (if ISOIL=2, or RDBRDALB=T)	cross
26	ALBEDO: background albedo (%) (if IFSOIL=2, RDBRDALB=T)	cross
27	VEGFRC: Vegetation coverage (if ISOIL = 2)	cross
28	SWOUT: top outgoing shortwave radiation (if FRAD>=2)	cross
29	LWOUT: top outgoing longwave radiation (if FRAD>=2)	cross
30	T2: 2 m temperature (K) (if IBLTYP=2, 4, 5)	cross
31	Q2: 2 m mixing ratio (kg/kg) (if IBLTYP=2, 4, 5)	cross
32	U10: 10 m u component of wind (m/sec) (if IBLTYP=2,4,5)	cross
33	V10: 10 m v component of wind (m/sec) (if IBLTYP=2,4,5)	cross
34	M-O LENG: Monin-Obukov length (m) (if ISOIL=3)	cross
35	NET RAD: surface net radiation (W/m2) (if ISOIL=3)	cross
36	ALBEDO: surface albedo (fraction) (if ISOIL=3)	cross
37	RA: aerodynamic resistance (s/m) (if ISOIL=3)	cross
38	RS: surface resistance (s/m) (if ISOIL=3)	cross
39	LAI: leaf area index (area/area) (if ISOIL=3)	cross
40	VEGFRC: vegetation fraction (fraction) (if ISOIL=3)	cross
41	ZNT: roughness length (m) (if ISOIL=3)	cross
42	ISLTYP: soil texture type (if ISOIL=3)	cross
43	SUMFB: mass flux updraft (if ICUPA=8)	cross
44	SPSRC: source layer updraft (if ICUPA=8)	cross
45	SEAICEFR: seaice fraction (if IPOLAR=1)	cross
46	TGSI: seaice temperature (if IPOLAR=1)	cross

2D constant fields dimensioned (IX, JX) include:

47	TERRAIN: Terrain elevation (m)	cross
48	MAPFACCR: Map scale factor	cross
49	MAPFACDT: Map scale factor	dot
50	CORIOLIS: Coriolis parameter (/s)	cross
51	RES TEMP: Substrate temperature (K)	cross
52	LATITCRS: Latitude (deg)	cross
53	LONGICRS: Longitude (deg)	cross
54	LANDUSE: Land-use category	cross
55	SNOWCOVR: Snow cover (if ISOIL < 2)	cross
56	TSEASFC: Sea surface temperature (K)	cross
57	SEAICE: Seaice (dimensionless) (if ISOIL=2)	cross

Other special output:

58	SIGMAH: Model half-sigma levels
59	ALBD: Surface albedo from LANDUSE.TBL
60	SLMO: Surface moisture availability from LANDUSE.TBL
61	SFEM: Surface emissivity from LANDUSE.TBL

- 62 SFZ0: Surface roughness from LANDUSE.TBL
- 63 THERIN: Surface thermal inertia from LANDUSE.TBL
- 64 SFHC: Soil heat capacity from LANDUSE.TBL
- 65 SCFX: Snow cover effect from LANDUSE.TBL

If one sets IFTSOUT = .TRUE., and defines TSLAT and TSLON for the time-series locations, one will obtain time-series output in fort.26 for domain 1, fort.27 for domain 2 and so on for serial runs (for MPI runs, the time series is (unfortunately) scattered in various rsl.out.\* files. The time-series output contains the following data:

```
xtime, time-step, its, jts, t-sfc, q-sfc, u-sfc, v-sfc,
pstar, pp-sfc, rainc, rainnc, clw, glw, hfx, qfx, gsw, t-ground
```

where

*xtime* : model time (unit *minutes*)  
*time-step* : the *nth* time series  
*its, jts* : I, J locations in model grid for time-series points  
*t-sfc* : 2-m or lowest  $\sigma$  level temperature (unit *K*)  
*q-sfc* : 2-m or lowest  $\sigma$  level mixing ratio (unit *kg/kg*)  
*u-sfc, v-sfc* : the 10-m or lowest- $\sigma$  level winds (unit *m s<sup>-1</sup>*), rotated to earth coordinates  
*pstar* : reference *p*\* (unit *cb*, or 10\*hPa)  
*pp-sfc* : perturbation pressure at the lowest- $\sigma$  level (unit *Pa*)  
*rainc, rainnc* : accumulative convective and non-convective surface precipitation (unit *cm*)  
*clw* : column integrated cloud liquid/ice (unit *mm*)  
*glw, gsw* : surface downward long-wave and shortwave radiation (unit *W m<sup>-2</sup>*)  
*hfx, qfx* : surface sensible and latent heat (\* latent heat of vaporization) fluxes (unit *W m<sup>-2</sup>*)  
*t-ground* : ground or skin temperature (if ISOIL = 2) temperature (unit *K*).

## 8.11 MM5 Files and Unit Numbers

MM5 accesses most files by referring to the file names. Fortran unit numbers associated with the files are assigned as follows:

**Table 8.1 File names, fortran unit numbers, and their description for MM5.**

File name	Unit number	Description
<b>INPUT</b>		
mmlif	fort.10	Input, namelist file
LOWBDY_DOMAIN1	fort.21, 22, ...	Lower boundary file, contains substrate temp and SST
BDYOUT_DOMAIN1	fort.9	Lateral boundary file created by program INTERPF
LANDUSE.TBL	fort.19	Physical properties for landuse categories
BUCKET.TBL	fort.18	Max, min moisture availability range, evaporation rate
VEGPARM.TBL	fort.19	used if ISOIL = 2

File name	Unit number	Description
SOILPARM.TBL	fort.19	used if ISOIL = 2
GENPARM.TBL	fort.19	used if ISOIL = 2
RRTM_DATA	fort.20	RRTM radiation scheme data
MMINPUT_DOMAINx (TERRAIN_DOMAIN2..)	fort.11, 12, ... 19	Initial condition files created by program INTERPF (or NESTDOWN); or Terrain output files for nests
MMINPUT(2)_DOMAINx	fort.31, 32, ... 39	3D analysis nudging files (same as initial condition files)
SFCFDDA_DOMAINx	fort.71, 72, ... 79 (fort.81, 82, ... 89)	Surface analysis nudging files created by program LITTLE_R/RAWINS
MM5OBS_DOMAINx	fort.61,62, ... 69	Observation nudging files created by user's own program
RESTART_DOMAINx	fort.91, 92 ... 99	Rrestart files (same as SAVE_DOMAINx files)
<b>OUTPUT</b>		
MMOUT_DOMAINx	fort.41,42, ... 49	MM5 model history files
SAVE_DOMAINx	fort.51, 52, ..., 59	restart files
SHUTDO_DOMAINx	fort.61, 62, ..., 69	Shutdown restart files
fort.26, fort.27....	fort.26, 27, ..., 29	time series output (IFTSOUT=T)

## 8.12 Configure.user Variables

The '*configure.user*' is the first file one needs to edit (if one is running Cray batch job, one would need to edit the mm5.deck only and these variables appear inside the deck). Except for the first variable, the rest are used for setting up model's memory - these variables are referred to as pre-compilation variables. Sections 1, 4 and make rules will be explained in Chapter 9.

<b>RUNTIME_SYSTEM</b>	computer system to run model on.
<b>FDDAGD</b>	=1, for 4DDA grid analysis nudging; =0, no 4DDA.
<b>FDDAOB</b>	=1, for 4DDA observation nudging; =0, no obs 4DDA.
<b>MAXNES</b>	maximum number of domains in simulation. Note though, there are only 4 default nest levels (i.e. 1 coarse domain and 3 nests)
<b>MIX,MJX,MKX</b>	maximum number of grid points in I, J, and K.
<b>IMPHYS</b>	options for explicit schemes:

---

	=1, dry; =2, removal of super-saturation; =3, warm rain (Hsie); =4, simple ice (Dudhia); =5, mixed phase (Reisner); =6, mixed phase with graupel (Goddard); =7, mixed phase with graupel (Reisner); =8, mixed phase with graupel (Schultz)
<b>MPHYSTBL</b>	=0, not using look-up table version; =1, use look-up table version of explicit scheme options 4 and 5; =2, use new optimized version (with vmass libraries).
<b>ICUPA</b>	options for cumulus parameterization schemes: =1, none; =2, Anthes-Kuo; =3, Grell; =4, Arakawa-Schubert; =5, Fritsch-Chappell; =6, Kain-Fritsch; =7, Betts-Miller; =8, Kain-Fritsch 2 (with shallow convection).
<b>IBLTYP</b>	options for planetary boundary layer schemes: =0, no PBL; =1, bulk PBL; =2, Blackadar PBL; =3, Burk-Thompson PBL; =4, Eta PBL; =5, MRF PBL; =6, Gayno-Seaman PBL; =7, Pleim-Chang PBL.
<b>FRAD</b>	options for atmospheric radiation schemes: =0, none; =1, simple cooling; =2, cloud (Dudhia) (require IMPHYS $\geq$ 3); =3, CCM2; =4, RRTM longwave scheme.
<b>IPOLAR</b>	=0, none; =1, polar physics (ISOIL .ne. 2)
<b>ISOIL</b>	=1, use the multi-layer soil model (require IBLTYP=2, 4, 5, 6); =0, no soil model; =2, Noah LSM model (requires IBLTYP=4, 5); =3, Pleim-Xiu LSM (requires IBLTYP=7).

**ISHALLO**                      =1, use shallow convective scheme (not well tested);  
                                 =0, no.

### 8.13 Script Variables for IBM Batch Deck:

<b>ExpName</b>	experiment name used in setting MSS pathname for output.
<b>InName</b>	input MSS pathname.
<b>RetPd</b>	mass store retention period (days).
<b>compile</b>	=yes, compile the mm5 code; =no, expect an existing executable.
<b>execute</b>	=yes, execute the model; =no, compile the code only.
<b>UseMySource</b>	=yes, use your own source code; =no, use mesouser version of the source code.
<b>CaseName</b>	MSS pathname for this run.
<b>STARTsw</b>	= NoReStart: start model run at hour zero (initialize). = ReStart: restart model run.
<b>FDDAsw</b>	= NoFDDA, no FDDA input files, = Anly, gridded FDDA input files, = Obs, obsFDDA input files, = Both, gridded and obs FDDA input files.
<b>InBdy</b>	MSS name of lateral boundary file.
<b>InLow</b>	MSS name for lower boundary condition file.
<b>InMM</b>	MSS name(s) of model input files.
<b>InRst</b>	MSS name(s) of model restart files.
<b>In4DSfc</b>	MSS name of surface analysis used for 4DDA.
<b>In4DObs</b>	MSS name of fdda obs files.
<b>Host</b>	= username@host.domain:/usr/tmp/username, host computer to rcp user's program tar file.
<b>OutMM</b>	MSS name for output.

### 8.14 Namelist Variables

A namelist file, called *mmlif*, is created when *mm5.deck* is executed. In MM5, this file is created

partially from the *configure.user* file, and partially from *mm5.deck*.

### 8.14.1 OPARAM

<b>TIMAX</b>	= forecast length in minutes.
<b>TISTEP</b>	= time step in seconds for the coarsest domain (recommend 3*dx(km)).
<b>IFREST</b>	=TRUE, for restart, =FALSE, for initial run.
<b>IXTIMR</b>	= integer time in minutes for restart.
<b>IFSAVE</b>	=TRUE, if saving data for restart, = FALSE, for no restart output.
<b>SVLAST</b>	= TRUE, if only saving the last time; = FALSE, save multiple times.
<b>SAVFRQ</b>	= frequency of restart output in minutes.
<b>IFTAPE</b>	= 1, for model output; =0, no model output.
<b>TAPFRQ</b>	= frequency of model history file output in minutes.
<b>BUFFRQ</b>	= how frequency to split model output files in minutes (ignored if < TAPFRQ).
<b>INCTAP</b>	= multipliers of TAPFRQ for outputting.
<b>IFRSFA</b>	= TRUE, if it is a restart run, using FDDA and multiple input files. Use with CDATEST.
<b>IFSKIP</b>	= TRUE, skip input files to start the model - DO NOT use this when restart.
<b>CDATEST</b>	= DATE (yyyy-mm-dd_hh:mm:ss) of the start file, used with IFSKIP/IFRSFA.
<b>IFPRT</b>	= 1, for printed output fields; = 0, for no printed output fields
<b>PRTRFQ</b>	= frequency of printed output fields in minutes
<b>MASCHK</b>	= integer frequency in number of time steps for budget/rainfall prints (coarsest mesh) - may not give correct answer on parallel computers.
<b>IFTSOUT</b>	= TRUE, if output time series; = FALSE, do not output time series
<b>TSLAT</b>	= latitudes of time series output locations
<b>TSLON</b>	= longitudes of time series output locations

### 8.14.2 LPARAM

1) Defined in *mm5.deck*:

<b>RADFRQ</b>	= frequency in minutes of radiation calculations (surface and atmospheric)
<b>IMVDIF</b>	=1, for moist vertical diffusion in clouds (requires IMPHYS>2, and IBLTYP=2, 5 or 7), = 0, vertical diffusion is dry
<b>IVQADV</b>	= 0, vertical moisture advection uses log interpolation (old method),

	= 1, vertical moisture advection uses linear interpolation (affects all moisture variables)
<b>IVTADV</b>	= 0, vertical temperature advection uses log interpolation (old method), = 1, vertical temperature advection uses linear interpolation
<b>ITHADV</b>	= 0, temperature advection and adiabatic term use temperature (old method), = 1, temperature advection and adiabatic term use potential temperature
<b>ITPDIF</b>	= 1, for diffusion using perturbation temperature in NH model; =2, use horizontal diffusion (new in version 3.7); = 0, not using this function (new in V2)
<b>TDKORR</b>	=2, temperature gradient correction fo horizontal diffusion (ITPDIF=2) at ground level uses ground temperature; =1, temperature gradient correction fo horizontal diffusion (ITPDIF=2) at ground level uses one sided difference of air temperature.
<b>ICOR3D</b>	= 1, for full 3D Coriolis force (requires INHYD=1), = 0, for traditional approximation.
<b>IEXSI</b>	= 0, no seaice = 1, seaice fraction diagnosed from sea-surface temperature (requires IPO- LAR=1) = 2, seaice fraction read from from LOWBDY file (requires IPOLAR=1)
<b>IFUPR</b>	= 1, for upper radiative boundary condition (NH run only). = 0, rigid upper boundary in nonhydrostatic runs.
<b>LEVSLP</b>	nest level (corresponding to LEVIDN) at which solar radiation starts to account for orography (to switch off, set to a large value). Only available for SWRAD (IFRAD=2,4).
<b>OROSHAW</b>	=1, include the effect of orography shadowing. This only has an effect if LEVSLP is also set. This option is not available for MM5 MPP runs.Only available for SWRAD (IFRAD=2,4). =0, do not include the effect of orography shawdowing.
<b>ITADVM</b>	=1, use instability limiter for temperature advection. =0, do not use instability limiter for temperature advection
<b>IQADVM</b>	=1, use instability limiter for QV/CLW advection. =0, do not use instability limiter for QV/CLW advection
<b>IBOUDY</b>	Boundary condition options: = 0, fixed, and for coarse domain only = 2, time-dependent (used for 2-way nested boundary condition) = 3, relaxation inflow/outflow, for coarse domain only



---

<b>IFDRY</b>	= 1, for fake dry run with no latent heating release (requires IMPHYS>1, and ICUPA=1)
<b>ISSTVAR</b>	= 1, update SST during a simulation (and snow cover and sea ice, if they are available). Must have at least SST field in the input; = 0, do not update SST (and snow cover and sea ice) during a simulation.
<b>IMOIAV</b>	used for bucket moisture scheme. = 0, do not use bucket scheme. = 1, use bucket scheme, and soil moisture is initialized with moisture availability values in LANDUSE.TBL. = 2: use bucket scheme, and soil moisture is initialized with soil moisture fields from MMINPUT files.
<b>IFSNOV</b>	= 1, snow cover effects (requires input SNOWC field from REGRID) = 2, snow-cover prediction (requires input WEASD field from REGRID, and use of IMPHYS = 4,5, and 7)
<b>ISFMTHD</b>	method for calculation of 2m/10m diagnostics = 0, old method = 1, new method for stable conditions (IBLTYP = 2 and 5 only)
<b>IZ0TOPT</b>	Thermal roughness length option for IBLTYP = 2 and 5 only. = 0, default (old) scheme = 1, Garratt formulation = 2, Zilitinkevich formulation
<b>ISFFLX</b>	= 1, compute surface heat and moisture fluxes; =0, no fluxes.
<b>ITGFLG</b>	= 1, ground temperature predicted; = 3, constant ground temperature.
<b>ISFPAR</b>	= 1, use TERRAIN-generated land-use categories; = 0, use only 2 (land/water) categories.
<b>ICLOUD</b>	= 1, consider cloud effects on surface radiation when FRAD=0,1 ; consider clouds in both surface and atmospheric radiation when FRAD=2,3,4; = 0, do not consider cloud effect on radiation; = 2, (IFRAD=3 only) radiation interacts with RH-derived cloud fraction only.
<b>IEVAP</b>	= 1, normal evaporative cooling; = 0, no evaporative effects; = -1, no precip evaporative cooling, (for IMPHYS=3,4, and 5).
<b>ISMIRD</b>	Soil moisture initialization method for IBLTYP = 7 (Pleim-Xiu scheme) only. = 0, use moisture availability from LANDUSE.TBL; = 2, use soil moisture input from REGRID.
<b>ISTLYR</b>	bottom of soil layers expected as input for SOIL=2,3
<b>ISMLYR</b>	bottom of soil layers expected as input for SOIL=2,3
<b>RDMAXALB</b>	whether to read in max snow albedo for ISOIL = 2 (Noah LSM) only.

---

= FALSE, do not use max snow albedo;  
= TRUE, use max snow albedo present in MMINPUT file.

**RDBRDALB**    whether to read in climatological month albedo for ISOIL = 2 (Noah LSM).  
= FALSE, do not use climatological monthly albedo;  
= TRUE, use climatological monthly albedo present in MMINPUT file.

2) Defined in *configure.user*, or internally produced:

**IFRAD**        see 'Configure.user variables'  
**ICUPA**        see 'Configure.user variables'  
**IBLTYP**       see 'Configure.user variables'  
**ISHALLO**     see 'Configure.user variables'  
**ISOIL**        see 'Configure.user variables'  
**IPOLAR**      see 'Configure.user variables'

### 8.14.3 NPARAM

**LEVIDN**       = level of nest for each domain (0 for domain 1 - default valid values are 0-3)  
**NUMNC**       = id number of parent domain for each domain (1 for domain 1)  
**NESTIX**       = I-dimension of each domain.  
**NESTJX**       = J-dimension of each domain.  
**NESTI**        = south-west corner point I for each domain.  
**NESTJ**        = south-west corner point J for each domain.  
**XSTNES**       = starting time in minutes for each domain.  
**XENNES**       = ending time in minutes for each domain.  
**IOVERW**       = 1, for initializing a nest from the nest input file, usually at model starting time;  
                  = 0, for interpolating to a nest from parent mesh, usually during model integration;  
                  = 2, for initializing domain with high resolution terrain, usually during model integration.  
**IACTIV**       = 1, if this domain is active when restart;  
                  = 0, if this domain is inactive.  
**IMOVE**        = 0, if domain does not move; =1, if domain will move.  
**IMOVCO**       = number of first move (always 1 at beginning, may change for restarts).  
**IMOVEI**       = increment in I (parent domain grids) of this move for this domain.

---

<b>IMOVEJ</b>	= increment in J (parent domain grids) of this move for this domain.
<b>IMOVET</b>	= time in minutes of this move for this domain (relative to beginning of the coarse mesh run).

Note: the default number of moves is 10.

<b>IFEED</b>	feedback from nest to coarse mesh in 2-way nests:
	= 0, no feedback;
	= 1, 9-point weighted average;
	= 2, 1-point feedback, with no smoothing;
	= 3, 1-point feedback, with smoother/desmoother (recommended);
	= 4, 1-point feedback, with heavy smoothing

#### 8.14.4 PPARAM

<b>ZZLND</b>	= roughness length over land (m) (if ISFPAR=0)
<b>ZZWTR</b>	= roughness length over water (m) (if ISFPAR=0)
<b>ALBLND</b>	= albedo over land (if ISFPAR=0)
<b>THILND</b>	= thermal inertia of land (cal-1 cm-2 K-1 s-0.5, if ISFPAR=0)
<b>XMAVA</b>	= moisture availability over land (if ISFPAR=0)
<b>CONF</b>	= non-convective precip saturation criterion (fraction $\leq 1$ for IMPHYS=1)
<b>SOILFAC</b>	= a factor to make 5-layer soil model time step more conservative. Higher number makes soil timestep shorter (range typically 1.0 - 2.0). Used in IBLTYP=1, 2, 4, 5, and 6.
<b>CZO,OZO</b>	= constants in Charnock relation for water roughness length. Used in IBLTYP = 2, 5 and 6.
<b>CKH</b>	= factor to control background diffusion coefficient used in the model. Default value is 1., which gives the same diffusion as versions before 3.5 if one uses 3xDX as the time step.

#### 8.14.5 FPARAM

<b>FDASTA</b>	(MAXSES); time (min) for initiation of FDDA.
<b>FDAEND</b>	(MAXSES); time (min) for termination of FDDA.
<b>I4D</b>	(MAXSES, 2); will FDDA analysis nudging be employed, (0=no; 1=yes).
<b>DIFTIM</b>	(MAXNES, 2); time (min) between input analyses for analysis nudging.
<b>IWIND</b>	(MAXSES, 2); will the wind field be nudged from analyses, (0=no; 1=yes).
<b>GV</b>	(MAXSES, 2); analysis-nudging coefficient ( $s^{-1}$ ) for wind.
<b>ITEMP</b>	(MAXSES, 2); will the temperature be nudged from analyses, (0=no; 1=yes).

<b>GT</b>	(MAXSES, 2); analysis-nudging coefficient ( $s^{-1}$ ) for temperature.
<b>IMOIS</b>	(MAXSES, 2); will the mixing ratio be nudged from analyses, (0=no; 1=yes).
<b>GQ</b>	(MAXSES, 2); analysis-nudging coefficient ( $s^{-1}$ ) for mixing ratio.
<b>IROT</b>	(MAXSES); will vorticity be nudged from analyses, (0=no; 1=yes).
<b>GR</b>	(MAXSES, 2); analysis-nudging coefficient ( $m^2 s^{-1}$ ) for vorticity.
<b>INONBL</b>	(MAXSES, 4); will PBL fields be nudged from 3-D analyses when not using surface-analysis nudging within PBL. (0=yes; 1=exclude certain variables depending on integer value of second index).
<b>RINBLW</b>	radius of influence (km) for surface-analysis nudging where the horizontal weighting function depends on surface data density.
<b>NPFG</b>	coarse-grid time-step frequency for select diagnostic print of analysis nudging.
<b>I4DI</b>	(MAXSES); will FDDA observation nudging be employed, (0=no; 1=yes).
<b>ISWIND</b>	(MAXSES); will the wind field be nudged from observations, (0=no; 1=yes).
<b>GIV</b>	(MAXSES); observation-nudging coefficient ( $s^{-1}$ ) for wind.
<b>ISTEMP</b>	(MAXSES); will the temperature be nudged from observations, (0=no; 1=yes).
<b>GIT</b>	(MAXSES); observation-nudging coefficient ( $s^{-1}$ ) for temperature.
<b>ISMOIS</b>	(MAXSES); will the mixing ratio be nudged from observations, (0=no; 1=yes).
<b>GIQ</b>	(MAXSES); observation-nudging coefficient ( $s^{-1}$ ) for mixing ratio.
<b>RINXY</b>	default horizontal radius of influence (km) for distance-weighted nudging corrections (for observation nudging).
<b>RINSIG</b>	vertical radius of influence (on sigma) for distance -weighted nudging corrections (for observation nudging).
<b>TWINDO</b>	(time window)/2 (min) over which an observation will be used for nudging.
<b>NPFI</b>	coarse-grid time-step frequency for select diagnostic print of observation nudging.
<b>IONF</b>	observation-nudging frequency in coarse grid time steps for observation-nudging calculations.
<b>IDYNIN</b>	for dynamic initialization using a ramp-down function to gradually turn off the FDDA before the pure forecast (1=yes, 0=no).
<b>DTRAMP</b>	the time period in minutes over which the nudging (obs nudging and analysis nudging) is ramped down from one to zero. Set dtramp negative if FDDA is to be ramped down BEFORE the end-of-data time (DATEND), and positive if the FDDA ramp-down period extends beyond the end-of-data time.g calculations.

## 8.15 Some Common Errors Associated with MM5 Failure

When an MM5 job is completed, always check for at least the following:

- The “STOP 99999” print statement indicates that MM5 completed without crashing.

- When running a batch job on NCAR's computer, check to be sure that the mswrite commands were all completed successfully by the shell, and that the files were written to the pathnames you expected.
- Check the top of "mm5.print.out" file to see if all domains are correctly initiated if running a multiple-domain job, and if the physics options are correctly specified.

If an MM5 job has failed, check for some of the common problems:

- If your model stops immediately after it prints out 'NON-HYDROSTATIC RUN' with an 'Segmentation fault' or sometimes 'Bus error', it is a good indication that the model is not getting enough memory to run. On most machine, typing 'unlimit' before you run the model will be the solution.
- "Read past end-of-file": This is usually followed by a fortran unit number. Check this unit number with Table 8.1 to find out which file MM5 has problem with. Check all the MSREAD statements in the printout to be sure that files were read properly from the MSS. Also check to make sure that the file sizes are not zero. Double-check experiment names, MSS pathnames.
- "Unrecognized namelist variable": This usually means there are typos in the namelist.
- Unmatched physics option: for instance, the following should appear in the print output:

```
STOP SEE ERRORS IN PRINT-OUT
```

If one browses through the output, one may find things like:

```
ERROR: IFRAD=2 REQUIRES IRDDIM=1 AND IMPHYS>3
```

which tells a user what a user needs to do to correct the problem.

- Uncompiled options:

```
STOP SEE ERRORS IN PRINT-OUT
```

If one browses through the output, one may find things like:

```
ERROR: IFRAD=2, OPTION NOT COMPILED
```

which tells a user the option you choose has not been compiled.

- When restarting a job, do not re-compile. If you do re-compile, do not change anything in the *configure.user* file.
- If the job stopped and there is a long list of "CFL>1...", it usually means the time step (TISTEP in namelist) is too big. Shorten the TISTEP and re-submit.
- If doing a multi-domain run, please check these namelist variables carefully:

```
LEVIDN = 0,1,1,1,1,1,1,1,1,1, ; level of nest for each domain
```

```
NUMNC = 1,1,1,1,1,1,1,1,1,1, ; ID of mother domain for each nest
```

## 8.16 MM5 tar File

The mm5.tar file contains the following files and directories:

CHANGES	Description of changes to the MM5 program
Makefile	Makefile to create MM5 executable
README	General information about the MM5 directory and

README.MPP	how to run MM5
Diff/	General information on how to compile and run on DM machines
Run/	Diff files for each new release
Templates/	Where MM5 runs
Util/	MM5 job decks for different machines
configure.user	Utility programs for cpp
configure.user.linux	Make rules and model configuration
domain/	Above for PC running Linux OS on single and OMP processors
dynamics/	
fdda/	
include/	
memory/	
physics/	
pick/	

The file *README* contains basic instructions on how to compile and run the model.

The file *README.MPP* contains basic information and instructions on how to start compiling and running MPP MM5.

The model is executed in the directory *Run*.

The bug fixes and changes to the source code and tar file are described in file *CHANGES* file and *diff.\** files in directory *Diff/*.

All FORTRAN files are in lower-case directories separated according to their functions. See the chart at the end of this chapter for a complete list of FORTRAN files.

When ‘*make code*’ command is executed, all .F and .f files selected for compiling are copied into the *pick/* directory. A single cat command will enable a user to generate a source listing (see the *README* file in directory *pick/*).

## 8.17 Configure.user

(This file is included here for reference only. Use the most up-to-date one from MM5.TAR file.)

```
# Sections
# 1. System Variables
# 3. Fortran options
#   3a. Cray (YMP, J90)
#       Note: set RUNTIME_SYSTEM="CRAY_IA" for Cray interactive job
#   3a2. Cray X1 Single Node OpenMP version
#   3b. IRIX.6.X (SGI_Origin,SGI_R10000,SGI_R8000 which support OpenMP)
#   3b2. IRIX.6.X (SGI_Origin,SGI_R10000,SGI_R8000)
#   3c. IRIX.5.2/5.3, IRIX.6.X (SGI_R4000/SGI_R4400/SGI_R5000)
#       Note: set RUNTIME_SYSTEM="SGI_R4000" for SGI_R4400/SGI_R5000
#   3d. SUN Fortran (solaris,SPARC20/SPARC64)
#   3e. DEC ALPHA (OSF/1)
#   3e2. DEC ALPHA (4100/8400; use OpenMP parallel directives)
#   3f. IBM (AIX)
#   3f2. IBM, OpenMP (AIX)
#   3g. HP (UX)
#   3h. HP (SPP-UX) for HP Exemplar S/X-Class Systems
#   3i1. PC_PGFL (LINUX/Portland Group Inc.)
#   3i2. PC_INTEL (LINUX/INTEL)
#   3j. MAC (OSX/xlf)
# 4. General commands
# 5. Options for making "./include/parame.incl"
# 6. Physics Options (memory related)
# 7. MPP Options (Set no options in section 3)
#   7a. IBM SP2
#   7a.1 IBM SP2 with SMP nodes
#   7b. Cray T3E
#   7c. SGI Origin 2000
#   7d. HP Exemplar
#   7e. Compaq ALPHA/MPI
#   7e.1 ALPHA Linux with MPI
#   7f. Fujitsu VPP
#   7g1. Network of Linux PCs with MPI (PGI)
#   7g2. Network of Linux PCs with MPI (INTEL)
#   7h. NEC SX/5 (under development)
#   7i. Sun MPI
#   7j. Cray X1
# 7k. Cray XD1, PGI Fortran
#
#-----
# 1. System Variables
#-----
SHELL = /bin/sh
RANLIB =echo
.SUFFIXES: .F .i .o .f .c
#-----
# 3. Fortran options
#   Uncomment the ones you need, including RUNTIME_SYSTEM
#-----
LIBINCLUDE = $(DEVTOP)/include
#-----
#   3a. Cray
#       Note: - imsl library is only needed if running Arakawa-Schubert cumulus
#       scheme;
#           and the location of the library may be different on non-NCAR Crays.
#           - if you are using the new program environment on Cray, should set
#           CPP = /opt/ctl/bin/cpp
#           - select the right compilation option for Cray - you may use
#           f90 option on paiute
#           - -x omp is needed for f90 compiler version 3.0.2.6 and above.
#           Check man page.
#-----
```

```
#RUNTIME_SYSTEM = "CRAY_IA"
#FC = f90
#FCFLAGS = -D$(RUNTIME_SYSTEM) -I$(LIBINCLUDE) -O task1 -x omp
#CFLAGS =
#CPP = /opt/ctl/bin/cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS =
#LOCAL_LIBRARIES = -L /usr/local/lib -l imsl
#MAKE = make -i -r
#-----
# 3a2. Cray X1 Single Node OpenMP version
#-----
#RUNTIME_SYSTEM = "crayx1"
## Use these for X1 cross compiler
#X1_CROSS_COMP = "gcc"
#X1_CROSS_CFLG = ""
## Use these for X1 native (trigger) compiler
##X1_CROSS_COMP = "cc"
##X1_CROSS_CFLG = "-hcommand"
#FC = ftn
### OpenMP in SSP mode
#FCFLAGS = -Ossp,task1,inline0 -xcsd,mic -sword_pointer -I$(LIBINCLUDE) -
D$(RUNTIME_SYSTEM)
#LDOPTIONS = -Ossp,task1
### Multi-streaming single MSP mode
###FCFLAGS = -O3 -Ogen_private_callee -xomp,mic -sword_pointer -I$(LIBINCLUDE)
-D$(RUNTIME_SYSTEM)
### LDOPTIONS =
#CFLAGS =
#CPP = cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----
# 3b. IRIX.6.X (SGI_Origin,SGI_R10000,SGI_R8000 which support OpenMP)
# Use OpenMP directives for multi-processor runs.
# - set RUNTIME_SYSTEM = SGI_Origin
# - works with 7.2.1 and above compiler
# - select appropriate XLOCAL0 macro for loader option
#
# - For parallel execution of MM5 set the following environment variables:
# setenv OMP_NUM_THREADS <number_of_processors>
# setenv DSM_PLACEMENT ROUND_ROBIN
# - For parallel execution on a processor set without contention:
# setenv DSM_WAIT SPIN
# setenv OMP_DYNAMIC FALSE
# setenv MPC_GANG OFF
# - For parallel execution on a contented set of processors:
# setenv DSM_WAIT YEILD
# setenv OMP_DYNAMIC TRUE
# setenv MPC_GANG OFF
#-----
#RUNTIME_SYSTEM = "SGI_Origin"
#FC = f77
#ABI = -n32 # 2 GB address space
##ABI = -64 # For 64-bit address space
#IO = -mpio
#PREC = # default 32-bit floating-point presicion.
##PREC = -r8 # 64-bit floating-point precision.
##Conversion program between different precisions of mminput and bdyout avail-
able from wesley@sgi.com
#MP = -mp -MP:old_mp=OFF
##MP = -mp -MP:open_mp=OFF # Use SGI multiprocessing directives
#OPT = -O3 -OPT:roundoff=3:IEEE_arithmetic=3 -OPT:reorg_common=OFF
##debugging#OPT = -g -DEBUG:div_check:subscript_check=ON:trap_uninitialized=ON
##select appropriate XLOCAL loader
```



---

```

#XLOCAL0 =
### Burk-Thompson PBL (IBLTYP=3) option mp directives
##XLOCAL0 = -Wl,-Xlocal,bt1_,-Xlocal,blk1_,-Xlocal,blk2_
### Noah LSM (ISOIL=2) option mp directives
##XLOCAL0 = -Wl,-Xlocal,rite_,-Xlocal,abci_
### Gayno-Seaman PBL (IBLTYP=6) option mp directives
##XLOCAL0 = -Wl,-Xlocal,fog1d_,-Xlocal,surface1_,-Xlocal,surface2_,-Xlo-
cal,surface3_,-Xlocal,comsurfslab_
#FCFLAGS = -I$(LIBINCLUDE) -D$(RUNTIME_SYSTEM) $(ABI) $(IO) $(PREC) $(MP)
$(OPT)
#CFLAGS =
#CPP = /usr/lib/cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS = $(ABI) $(PREC) $(MP) $(OPT) $(XLOCAL0)
#LOCAL_LIBRARIES = -lfastm
#MAKE = make -i -r -P
#-----
# 3b2. IRIX.6.X (SGI_Origin,SGI_R10000,SGI_R8000)
#      Use SGI directives for multi-processor runs.
#      - set RUNTIME_SYSTEM = SGI_R8000
#      - use the appropriate LDOPTIONS if compiling Burk-Thompson PBL,
#        Gayno-Seaman PBL, or Noah land-surface module
#      - use 7.0 and above compiler
#      - do not use -lfastm for R10000 and Origin series for compiler
#        versions 7.0 and 7.1, unless patches are installed. For more
#        information please see MM5 Web page:
#        http://www.mmm.ucar.edu/mm5/mm5v2-sgi.html
#-----
#RUNTIME_SYSTEM = "SGI_R8000"
#FC = f77
#FCFLAGS = -I$(LIBINCLUDE) -O3 -n32 -mips4 -mp -OPT:round-
off=3:IEEE_arithmetic=3
#CFLAGS =
#CPP = /usr/lib/cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS = -n32 -mips4 -mp
###Burk-Thompson (IBLTYP=3) option mp directives
##LDOPTIONS = -n32 -mips4 -mp -Wl,-Xlocal,bt1_,-Xlocal,blk1_,-Xlocal,blk2_
###Noah LSM (ISOIL=2) option mp directives
##LDOPTIONS = -n32 -mips4 -mp -Wl,-Xlocal,rite_,-Xlocal,abci_
### Gayno-Seaman (IBLTYP=6) option mp directives
##LDOPTIONS = -n32 -mips4 -mp -Wl,-Xlocal,fog1d_,-Xlocal,surface1_,-Xlo-
cal,surface2_,-Xlocal,surface3_,-Xlocal,comsurfslab_
#LOCAL_LIBRARIES = -lfastm
##LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----
# 3c. IRIX.6.X (SGI_R4400/SGI_R4000/SGI_R5000)
#-----
#RUNTIME_SYSTEM = "SGI_R4000"
#FC = f77
#FCFLAGS = -I$(LIBINCLUDE) -mips2 -32 -O2 -Nn30000 -Olimit 1500
#CFLAGS =
#CPP = /usr/lib/cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS =
#LOCAL_LIBRARIES = -lfastm
#MAKE = make -i -r
#-----
# 3d. SUN (solaris,SPARC20/SPARC64)
#-----
#RUNTIME_SYSTEM = "SUN"
#FC = f90
#FCFLAGS = -fast -O2 -I$(LIBINCLUDE)
#CFLAGS =
#LDOPTIONS = -fast -O2
#CPP = /usr/lib/cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P

```

---

```
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----
#      3e. DEC_ALPHA (OSF/1)
#-----
#RUNTIME_SYSTEM = "DEC_ALPHA"
#FC = f90
#FCFLAGS = -cpp -D$(RUNTIME_SYSTEM) -I$(LIBINCLUDE) -c -O4 -Olimit 2000 -auto-
#atic \
#      -fpe0 -align dcommons -align records -convert big_endian
###FCFLAGS = -cpp -D$(RUNTIME_SYSTEM) -DIBMopt -DvsLIB -I$(LIBINCLUDE) -c -O4
#-Olimit 2000 -automatic \
###      -fpe0 -align dcommons -align records -convert big_endian
#CFLAGS =
#CPP = cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS = -math_library accurate
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----
#      3e2. DEC_ALPHA (4100/8400 Series)
#      Use OpenMP directives for multi-processor runs.
#      - set RUNTIME_SYSTEM = DEC_ALPHA
#-----
#RUNTIME_SYSTEM = "DEC_ALPHA"
#FC = f90
#FCFLAGS = -omp -cpp -D$(RUNTIME_SYSTEM) -I$(LIBINCLUDE) -c -O4 -Olimit 2000 \
#-automatic -fpe0 -align dcommons -align records -convert big_endian
#CFLAGS =
#CPP = cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS = -omp -math_library accurate
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----
#      3f. IBM (AIX)
#-----
#RUNTIME_SYSTEM = "IBM"
#FC = xlf
#FCFLAGS = -I$(LIBINCLUDE) -O3 -qarch=auto -qmaxmem=-1
#CPP = /usr/lib/cpp
#CFLAGS =
#CPPFLAGS = -I$(LIBINCLUDE) -C -P -Drs6000
#LDOPTIONS = -qmaxmem=-1 -O3 -qarch=auto
#LOCAL_LIBRARIES = -lmass
#MAKE = make -i
#-----
#      3f2. IBM (AIX)
#      - Depending on problem size and machine memory size, the settings
#      of maxstack and maxdata may need to be modified.
#      - If the newer thread-safe mass library is available, add
#      the -lmass_r option to LOCAL_LIBRARIES.
#-----
#RUNTIME_SYSTEM = "IBM"
#FC = xlf_r
#FCFLAGS = -I$(LIBINCLUDE) -O2 -qarch=auto -qmaxmem=-1 -qsmp=omp:noauto -
#qnosave -qstrict -qnoclines
#CPP = /usr/lib/cpp
#CFLAGS =
#CPPFLAGS = -I$(LIBINCLUDE) -C -P -Drs6000
#LDOPTIONS = -qmaxmem=-1 -O2 -qarch=auto -bmaxstack:512000000 -bmax-
#data:2000000000
#LOCAL_LIBRARIES = -lxlsmpl -lmass_r
#LOCAL_LIBRARIES = -lxlsmpl
#MAKE = make -i
#-----
```

---

```

#      3g. HP (UX)
#-----
#RUNTIME_SYSTEM = "HP"
#FC = f77
#FCFLAGS = -I$(LIBINCLUDE) -O
#CPP = /usr/lib/cpp
#CFLAGS = -Aa
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS =
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----
#      3h. HP-SPP (SPP-UX), and HP-SPP_IA
#-----
#RUNTIME_SYSTEM = "HP-SPP"
#FC = f77
#PA8K = +DA2.0N +DS2.0a
#ARCH = ${PA8K}
#PROFILE =
#INLINE = +Olimit +Oinline=_saxpy,vadv,hadv,sinty,sintx,slab,diffut
#PARALLEL = +O3 +Oparallel +Onofail_safe +Onoautopar +Onodynsl
#
## Use the following FCFLAGS to build single-threaded executable
##FCFLAGS = ${PROFILE} ${ARCH} -I$(LIBINCLUDE) +O3 +Oaggressive \
##          +Olibcalls ${INLINE}
#
## Use the following FCFLAGS to build a parallel executable
#FCFLAGS = ${PROFILE} ${ARCH} -I$(LIBINCLUDE) ${PARALLEL} \
#          +O3 +Oaggressive +Olibcalls ${INLINE}
#
#CPP = /usr/lib/cpp
#CFLAGS = ${PROFILE} -Aa
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS = ${FCFLAGS} -Wl,-aarchive shared -Wl,+FPD
#LOCAL_LIBRARIES = -Wl,/usr/lib/pa1.1/libm.a
#MAKE = gmake -j 4 -i -r
#-----
#      3i1. PC PGF77 (LINUX/Portland Group Inc.)
#      pgf77 version 1.6 and above
#      May use pgf90 if the version is 3.1-4
#-----
#RUNTIME_SYSTEM = "linux"
#FC = pgf90
#FCFLAGS = -I$(LIBINCLUDE) -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswa-
pio
##FCFLAGS = -I$(LIBINCLUDE) -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -
byteswapio -mp \
##-Mnosgimp
#CPP = /lib/cpp
#CFLAGS = -O
#CPPFLAGS = -I$(LIBINCLUDE)
#LDOPTIONS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio
##LDOPTIONS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio -mp
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----
#      3i2. PC_INTEL (LINUX/INTEL)
#-----
#RUNTIME_SYSTEM = "linux"
#FC = ifort
#FCFLAGS = -I$(LIBINCLUDE) -O2 -tp p6 -pc 32 -convert big_endian
#CPP = /lib/cpp
#CFLAGS = -O
#CPPFLAGS = -I$(LIBINCLUDE)
#LDOPTIONS = -O2 -tp p6 -pc 32 -convert big_endian
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----

```

---

```

#      3j. MAC (OSX/xlf)
#-----
#RUNTIME_SYSTEM = "macxlf"
#FC = xlf
#FCFLAGS = -I$(LIBINCLUDE) -qarch=auto
#CPP = /usr/bin/cpp
#CFLAGS = -O -DNOUNDERSCORE
#CPPFLAGS = -I$(LIBINCLUDE) -I. -C -P -DIBM -xassembler-with-cpp
#LDOPTIONS = -Wl,-stack_size,10000000,-stack_addr,0xc0000000
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#RANLIB = ranlib
#-----
# 4. General commands
#-----
AR = ar ru
RM = rm -f
RM_CMD = $(RM) *.CKP *.ln *.BAK *.bak *.o *.i core errs ,* *~ *.a \
.emacs_* tags TAGS make.log MakeOut *.f !
GREP = grep -s
CC = cc
#-----
# 5. Options for making ./include/parame.incl
#-----
#
# FDDAGD (integer)                - "1" -> FDDA gridded run
FDDAGD = 0
#
# FDDAOBS (integer)              - "1" -> FDDA obs run
FDDAOBS = 0
#
# MAXNES (integer)               - Max Number of Domains in simulation
MAXNES = 2
#
# MIX,MJX (integer)              - Maximum Dimensions of any Domain
MIX = 49
MJX = 52
# MKX (integer)                  - Number of half sigma levels in model
MKX = 23
#-----
# 6. Physics Options
#   The first MAXNES values in the list will be used for the corresponding
#   model nests; the rest in the list can be used to compile other options.
#   The exception is FRAD, of which only the first value is used in the
model,
#   (i.e., only one radiation option is used for all nests). The rest allow
#   other options to be compiled.
#   Compilation of Arakawa-Schubert cumulus scheme requires imsl.
#-----
# IMPHYS - for explicit moisture schemes (array,integer)
IMPHYS = "4,4,1,1,1,1,1,1,1,1"
#
#                               - Dry,stable,warm rain,simple ice,mix phase,
#                               - 1 ,2 ,3 ,4 ,5
#                               - graupel(gsfcl),graupel(reisner2),schultz
#                               -,6 ,7 ,8
MPHYSTBL = 0
#
#                               - 0=do not use look-up tables for moist
#                               physics
#                               - 1=use look-up tables for moist physics
#                               (currently only simple ice and mix phase
#                               are available)
#                               - 2=optimized exmoisr routine (need vslib,
if not
#                               available set -DvsLIB in compile flags)
#
# ICUPA - for cumulus schemes (array,integer)

```

---

```

#                                     - None, Kuo, Grell, AS, FC, KF, BM, KF2 -
1,2,3,4,5,6,7,8
ICUPA = "3,3,1,1,1,1,1,1,1,1"
#
# IBLTYP - for planetary boundary layer (array, integer)
#                                     - 0=no PBL fluxes, 1=bulk, 2=Blackadar,
#                                     3=Burk-Thompson, 4=Eta M-Y, 5=MRF,
#                                     6=Gayno-Seaman, 7=Pleim-Xiu
IBLTYP = "5,5,0,0,0,0,0,0,0,0"
#
# FRAD - for atmospheric radiation (integer)
#                                     - Radiation cooling of atmosphere
#                                     0=none, 1=simple, 2=cloud, 3=ccm2, rrtm=4
FRAD = "2,0,0,0,0,0"
#
# IPOLAR - (integer) for polar model used only if ISOIL=1
#                                     0=not polar (5-layer soil model)
#                                     1=polar (7-layer snow/soil model)
IPOLAR = 0
#
# ISOIL - for multi-layer soil temperature model (integer)
#                                     - 0=no, 1=yes (only works with IBLTYP=2,4,5,6)
#                                     2=Noah land-surface scheme (IBLTYP=4,5 only)
#                                     3=Pleim-Xiu LSM (IBLTYP=7 only)
ISOIL = 1
#
# ISHALLO (array, integer)           - Shallow Convection Option
#                                     1=shallow convection, 0=No shallow convection
ISHALLO = "0,0,0,0,0,0,0,0,0,0"
#-----
# 7. MPP options
#
# For general information and updated "helpdesk" information see
#   http://www.mmm.ucar.edu/mm5/mpp
#   http://www.mmm.ucar.edu/mm5/mpp/helpdesk
#-----
#
#                                     Presently, of the MPP platforms only the "sp2"
#                                     is supplied with the "make deck" capability.
#
# MPP Software Layer
MPP_LAYER=RSL
#MPP_LAYER=NNTSMS
#
# PROCMIN_NS - minimum number of processors allowed in N/S dim
#
PROCMIN_NS = 1
#
# PROCMIN_EW - minimum number of processors allowed in E/W dim
#
PROCMIN_EW = 1
#
# ASSUME_HOMOGENEOUS_ENVIRONMENT - on a machine with a heterogeneous
# mix of processors (different speeds) setting this compile time
# constant to 0 (zero) allows the program to detect the speed of each
# processor at the beginning of a run and then to attempt to come up with
# an optimal (static) mapping. Set this to 0 for a heterogeneous
# mix of processors, set it to 1 for a homogeneous mix. Unless you
# are certain you have a heterogeneous mix of processors, leave this
# set to 1. Currently, this option is ignored on platforms other
# than the IBM SP.
#
ASSUME_HOMOGENEOUS_ENVIRONMENT = 1
#-----
#   7a. IBM SP2
#       type 'make mpp' for the SP2

```

---

```
#-----
#RUNTIME_SYSTEM = "sp2"
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = mpixlf_r
#MCC = mpcc_r
#MLD = mpixlf_r
#FCFLAGS = -O2 -qmaxmem=-1 -qarch=auto -qfloat=hsflt
#LDOPTIONS = -bmaxdata:0x70000000
#LOCAL_LIBRARIES = -lmassv
##LOCAL_LIBRARIES = -lmass
###LOCAL_LIBRARIES = -lessl
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -DMPI -Drs6000 -DSYSTEM_CALL_OK -DIBMopt
##CPPFLAGS = -DMPI -Drs6000 -DSYSTEM_CALL_OK -DIBMopt -DvsLIB
#CFLAGS = -DNOUNDERSCORE -DMPI
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
#       7a.1 IBM SP with Silver or Winterhawk nodes
#       type 'make mpp' for the SP2
#       - You must compile with XLF or MPXLF version 6.1 or greater.
#       - Check with your system admin before linking to lessl or lmass.
#       - Note for running on blue.llnl.gov:
#       newmpixlf_r is LLNL specific wrapper around HPF 6.1 w/ HPF off.
#       - If the newer thread-safe mass library is available, add
#       the -lmass_r option to LOCAL_LIBRARIES.
#       - For very large domains, use -bmaxdata:2000000000 -bmaxstack:268435456
#       for load options (Peter Morreale/SCD)
#       - If you enable -O3 optimization, add -qstrict as well
#-----
#RUNTIME_SYSTEM = "sp2"
#MPP_TARGET=$(RUNTIME_SYSTEM)
## On llnl.blue.gov, (3/99)
##MFC = time newmpixlf_r
##MCC = mpcc_r
##MLD = newmpixlf_r
## On systems with R6.1 or greater of IBM Fortran.
#MFC = time mpixlf_r
#MCC = mpcc_r
#MLD = mpixlf_r
#FCFLAGS = -O2 -qarch=auto -qcache=auto -qzerosize -qsmp=noauto -qnosave -
qmaxmem=-1 \
#       -qspillsize=2000
#LDOPTIONS = -qsmp=noauto -bmaxdata:0x70000000
##LOCAL_LIBRARIES = -lmass_r
##LOCAL_LIBRARIES = -lessl
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -DMPI -Drs6000 -DSYSTEM_CALL_OK
#CFLAGS = -DNOUNDERSCORE -DMPI
```

---

```

#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
#       7b. T3E
#-----
#RUNTIME_SYSTEM = "t3e"
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = f90
#MCC = cc
#MLD = $(MFC)
##FCFLAGS = -g
#FCFLAGS = -O2
#LDOPTIONS =
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /opt/ctl/bin/cpp -C -P
#CPPFLAGS = -DMPI -DT3E
#CFLAGS = -DNOUNDERSCORE -Dt3e -DT3E -DMPI
#ARCH_OBJS = error_dupt3d.o t3etraps.o set_to_nan.o milliclock.o
#IWORDSIZE = 8
#RWORDSIZE = 8
#LWORDSIZE = 8
#-----
#       7c. Origin 2000
#       Note that the MPP version of MM5 is not supported for compilation under
#       the "modules" environment.  To see if you are using modules to control
#       compiler versions on your machine, type "module list".
#
#       It may be necessary to modify the MPI run time environment on the
#       Origin as follows:
#
#           setenv MPI_MSGS_PER_PROC 4096
#
#       See also http://www.mmm.ucar.edu/mm5/mpp/helpdesk/20000621.txt
#-----
#RUNTIME_SYSTEM = "o2k"
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = f90 -64 -mips4 -w
#MCC = cc -64 -mips4 -w
#MLD = f90 -64 -mips4
##FCFLAGS = -g
#FCFLAGS = -O3 -OPT:roundoff=3:IEEE_arithmetic=3 -OPT:fold_arith_limit=2001
#LDOPTIONS =
#LOCAL_LIBRARIES = -lfastm -lmpi
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -DMPI -DO2K -DDEC_ALPHA -DSYSTEM_CALL_OK
#CFLAGS = -DO2K -DMPI -DDEC_ALPHA
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----

```

---

```
#      7d. HP Exemplar
#-----
#RUNTIME_SYSTEM = "hp"
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = f77
#MCC = mpicc
#MLD = mpif77
##FCFLAGS = +DA2.0N +DS2.0a -g
#FCFLAGS = +DA2.0N +DS2.0a +O3
#LDOPTIONS =
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -DMPI -DSYSTEM_CALL_OK
#CFLAGS = -DNOUNDERSCORE -DMPI
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
#      7e. Compaq ALPHA/MPI/OpenMP (Thanks to Dave Sherden)
#      - For multi-threaded MPI processes (useful on dm-clusters of SMP
#        nodes; such as fir.mmm.ucar.edu), uncomment the definition
#        of the macro: SPECIAL_OMP.
#      - If running with MPICH (public domain MPI) uncomment
#        first set of definitions for MFC, MCC, MLD and LDOPTIONS. If using
#        the Compaq/DEC MPI, uncomment the second set.
#      - On prospect.ucar.edu (ES40), add the -lelan option to LDOPTIONS.
#-----
#RUNTIME_SYSTEM = "alpha"
#MPP_TARGET=$(RUNTIME_SYSTEM)
##### If using OpenMP for SMP parallelism on each MPI process ###
##SPECIAL_OMP = -omp
##### If using MPICH ###
#MFC = f77
#MCC = mpicc
#MLD = mpif77
#LDOPTIONS = $(SPECIAL_OMP)
##### If using DEC MPI (e.g. on fir.mmm.ucar.edu) ###
##### Compaq ES40 Cluster (prospect.ucar.edu) requires -lelan for OpenMP
##MFC = f90
##MCC = cc
##MLD = f90
##LDOPTIONS = -lmpi -lelan $(SPECIAL_OMP)
#LDOPTIONS = -lmpi $(SPECIAL_OMP)
#####
#FCFLAGS = -O4 -Olimit 2000 -fpe0 -align dcommons -align records \
#      -convert big_endian $(SPECIAL_OMP)
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = cpp -C -P
#CPPFLAGS = -DMPI -DDEC_ALPHA -DSYSTEM_CALL_OK
#CFLAGS = -DMPI -DDEC_ALPHA
#ARCH_OBJS = milliclock.o
```



---

```

#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
#       7e.1  ALPHA Linux with MPI (Thanks Greg Lindahl, HPTi)
#             (This has run on jet.fsl.noaa.gov)
#-----
#RUNTIME_SYSTEM = "alpha"
#MPP_TARGET=$(RUNTIME_SYSTEM)
##### If using OpenMP for SMP parallelism on each MPI process ###
##SPECIAL_OMP = -omp
#####
#MFC = fort
#MCC = mpicc
#MLD = mpif77
#UNDERSCORE = -DF2CSTYLE
#LDOPTIONS = $(SPECIAL_OMP) -static
#FCFLAGS = -O5 -arch ev6 -tune ev6 -align dcommons -align records \
#          -convert big_endian $(SPECIAL_OMP)
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -traditional -C -P
#CPPFLAGS = -DMPI -DDEC_ALPHA $(UNDERSCORE) -DSYSTEM_CALL_OK
#CFLAGS = -DMPI -DDEC_ALPHA $(UNDERSCORE)
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
#       7f.  Fujitsu VPP
#
# These options have been updated for the newer VPP5000 system. If you
# find that you have trouble compiling on your system, try removing the
# -KA32 and -Ka4 option from FCFLAGS, LDOPTIONS, CFLAGS and from
# MPP/RSL/RSL/makefile.vpp. Note that to successfully compile the RSL
# library (MPP/RSL/RSL) you need the following two environment variables
# set (syntax may vary with shells other than csh):
#
# Older systems:
#
#     setenv MPIINC_DIR /usr/lang/mpi/include
#     setenv MPILIBS '-Wl,-P -L/usr/lang/mpi/lib -lmpi -lmp'
#
# Newer systems:
#
#     setenv MPIINC_DIR /usr/lang/mpi2/include32
#     setenv MPILIBS '-Wl,-P -L/usr/lang/mpi2/lib32 -lmpi -lmp'
#
# Note for older systems. The configure.user is set up for VPP5000.
# For older (VPP300/700) systems, it may be necessary to remove the
# -KA32 and -Ka4 flags in the settings below.
#
# Note with v3.4: VECTOR=1 works only with IMPHYS=5, IBLTYP=5, and ICUPA=3.
# Other IMPHYS options and ICUPA options will work but won't be vector
# optimized.  IBLTYP=2 will not compile with VECTOR=1.
#
# Debugging VECTOR=1 option on non-vector platforms: see MPP/README_VECDEBUG
#-----
#RUNTIME_SYSTEM = "vpp"
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = frt

```

---

```
#MCC = cc
#MLD = frt
### debugging
### FCFLAGS = -Sw -g -Pdos -lmpi -lmp
### debugging; for debugging without MPI (also need to compile RSL with -
DSTUBS)
### FCFLAGS = -Sw -g -Pdos -Of,-P,-E
#FCFLAGS = -Sw -Wv,-Of,-te,-ilfunc,-noalias,-m3,-P255 \
#           -Oe,-P -Kfast -Pdos -lmpi -lmp -KA32
#FCVFLAGS = -Sw -Wv,-te,-noalias,-ilfunc,-Of,-m3,-P255 \
#           -Of,-e,-P,-u -Kfast -Pdos -lmpi -lmp -KA32
#LDOPTIONS = -Wl,-P -L$(MPILIBS) -lmpi -J -lmp -KA32
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = $(CAT)
#M4 = m4
#CPP = /lib/cpp -C -P
### Uncomment only for debugging without MPI
### CPPFLAGS = -DMPI -Dvpp -I$(MPIINCDIR) -DKMA -DSTUBS -DSYSTEM_CALL_OK
### CFLAGS = -DMPI -Dvpp -I$(MPIINCDIR) -KA32 -Ka4 -DSTUBS
### Normal settings for CPPFLAGS and CFLAGS
#CPPFLAGS = -DMPI -Dvpp -I$(MPIINCDIR) -DKMA -DSYSTEM_CALL_OK
#CFLAGS = -DMPI -Dvpp -I$(MPIINCDIR) -KA32 -Ka4
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#FLIC_MACROS = LMvpp.m4
#VECTOR = 1
#-----
# 7gl. Linux PCs. Need Portland Group pgf77 and MPICH.
#
# The following information has been added to this file with MM5v3.2:
#
# This expects mpif77 and mpicc to be installed on your system in
# $(LINUX_MPIHOME)/bin . These should be configured to use the Portland Group
# pgf77 (v3 or higher) and gcc, respectively. For information on how to
# download, install, and configure mpich on your system, see:
#
#   http://www.mcs.anl.gov/mpi/mpich
#
# Information on Portland Group compiler:
#
#   http://www.pgroup.com
#
# If using a different Fortran compiler, modify FCFLAGS and LDOPTIONS as
# needed. The compiler should be capable of doing little- to big-endian
# conversion and it should understand integer (Cray-style) pointers. It
# is recommended that the same fortran compiler be used to compile
# mpich. Edit the LINUX_MPIHOME macro, below, to point to the top level mpich
# directory. See also:
#
#   http://www.mmm.ucar.edu/mm5/mpp/linuxhelp.html (by Steve Webb, NCAR/RAP)
#
# Note for pgf77 on RedHat Linux6: patches available from Portland Group at:
#
#   http://www.pgroup.com/downloads/rh6patches.html
#-----
#RUNTIME_SYSTEM = "linux"
#MPP_TARGET=$(RUNTIME_SYSTEM)
## edit the following definition for your system
```

---

```

#LINUX_MPIHOME = /usr/local/mpich
#MFC = $(LINUX_MPIHOME)/bin/mpif77
#MCC = $(LINUX_MPIHOME)/bin/mpicc
#MLD = $(LINUX_MPIHOME)/bin/mpif77
#FCFLAGS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio
#LDOPTIONS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio
#LOCAL_LIBRARIES = -L$(LINUX_MPIHOME)/build/LINUX/ch_p4/lib -lfpmpich -lmpich
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P -traditional
#CPPFLAGS = -DMPI -Dlinux -DSYSTEM CALL_OK
#CFLAGS = -DMPI -I$(LINUX_MPIHOME)/include
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
# 7g2. Linux PCs. Need INTEL and MPICH.
#-----
#RUNTIME_SYSTEM = "linux"
#MPP_TARGET=$(RUNTIME_SYSTEM)
### edit the following definition for your system
#LINUX_MPIHOME = /usr/local/mpich-intel
#MFC = $(LINUX_MPIHOME)/bin/mpif77
#MCC = $(LINUX_MPIHOME)/bin/mpicc
#MLD = $(LINUX_MPIHOME)/bin/mpif77
#FCFLAGS = -O2 -convert big_endian -pc32
#LDOPTIONS = -O2 -convert big_endian -pc32
#LOCAL_LIBRARIES = -L$(LINUX_MPIHOME)/build/LINUX/ch_p4/lib -lfpmpich -lmpich
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = /usr/bin/expand
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -traditional -DMPI -Dlinux
#CFLAGS = -DMPI -I/usr/local/mpi/include
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
# 7h. NEC SX-4 (under development)
#-----
#RUNTIME_SYSTEM = sx
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = f90
#MCC = cc
#MLD = $(MFC)
#FCFLAGS = -V -E P -Wf"-init stack=zero heap=zero -O nooverlap" -USX -float0 \
#          -D$(RUNTIME_SYSTEM) -I$(LIBINCLUDE) -Wf"-L transform fmtlist sum-
mary" -g
#FCFLAGS = -V -E P -C vopt -Wf"-init stack=zero heap=zero -O nooverlap" \
#          -ew -USX -float0 -D$(RUNTIME_SYSTEM) -I$(LIBINCLUDE) \
#          -Wf"-L transform fmtlist summary"
#LDOPTIONS = -float0 -lmpi -lmpi-w -g
#CFLAGS =
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed

```

---

```
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -DMPI -Dvpp -I$(LIBINCLUDE) -C -P -DDEC_ALPHA -DSYSTEM_CALL_OK
#CFLAGS = -DMPI -Dvpp -DDEC_ALPHA
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#ASSUME_HOMOGENEOUS_ENVIRONMENT = 1
#FLIC_MACROS = LMvpp.m4
#VECTOR = 1
#-----
#       7i. Sun MPI (tested on k2.ucar.edu)
#-----
#RUNTIME_SYSTEM = "sunmpi"
#MPP_TARGET=$(RUNTIME_SYSTEM)
##### If using OpenMP for SMP parallelism on each MPI process ###
##SPECIAL_OMP = ??
#MFC = mpf90
#MCC = mpcc
#MLD = mpf90
#LDOPTIONS = -fast -O2 -lmpi
#####
#FCFLAGS = -fast -O2 $(SPECIAL_OMP)
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = cpp -C -P
#CPPFLAGS = -DMPI -DSYSTEM_CALL_OK
#CFLAGS = -DMPI
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
#       7j. Cray X1
#-----
#RUNTIME_SYSTEM = "crayx1"
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = ftn
#MCC = cc
#MLD = $(MFC)
## Use these for X1 cross compiler
#X1_CROSS_COMP = "gcc"
#X1_CROSS_CFLG = ""
## Use these for X1 native (trigger) compiler
##X1_CROSS_COMP = "cc"
##X1_CROSS_CFLG = "-hcommand"
#
#FCFLAGS = -x omp,mic -O3 -Ofp3 -Ogen_private_callee -V -ra -sword_pointer -
D$(RUNTIME_SYSTEM)
##FCFLAGS = -x omp,mic -Oscalar2,stream3,vector3 -Ofp3 -Ogen_private_callee -V
-ra -sword_pointer -D$(RUNTIME_SYSTEM)
#
#LDOPTIONS =
#LOCAL_LIBRARIES = -lmalloc
#MAKE = make -i -r
#AWK = awk
```

---

```

#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = cpp -C -P
#CPPFLAGS = -DMPI -D$(RUNTIME_SYSTEM) -DKMA
#CFLAGS = -V -O3 -h display_opt -h report=imsvf -DMPI -D$(RUNTIME_SYSTEM)
#ARCH_OBJS = error_dupt3d.o set_to_nan.o milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
#       7k. Cray XD1, Linux Opteron.  Need Portland Group pgf90.
#
# The following information has been added to this file with MM5v3.6.3:
#
# Information on Portland Group compiler:
#
#   http://www.pggroup.com
#
# If using a different Fortran compiler, modify FCFLAGS and LDOPTIONS as
# needed.  The compiler should be capable of doing little- to big-endian
# conversion and it should understand integer (Cray-style) pointers.  It
# is recommended that the same fortran compiler be used to compile
# mpich.  Edit the LINUX_MPIHOME macro, below, to point to the top level mpich
# directory.  See also:
#
#   http://www.mmm.ucar.edu/mm5/mpp/linuxhelp.html (by Steve Webb, NCAR/RAP)
#
# Note for pgf77 on RedHat Linux6: patches available from Portland Group at:
#
#   http://www.pggroup.com/downloads/rh6patches.html
#-----
#RUNTIME_SYSTEM = "linux"
#MPP_TARGET=$(RUNTIME_SYSTEM)
# edit the following definition for your system
#LINUX_MPIHOME = /usr/mpich/mpich-1.2.5
### mpif77, mpicc are not yet installed on XD1
#MFC = $(LINUX_MPIHOME)/bin/mpif77
#MCC = $(LINUX_MPIHOME)/bin/mpicc
#MLD = $(LINUX_MPIHOME)/bin/mpif77
#MFC = pgf90
#MCC = pgcc
#MLD = pgf90
#FCFLAGS = -DDEC_ALPHA -O3 -fastsse -Mnoreentrant -Mcray=pointer -Mnoframe -
byteswapio
#LDOPTIONS = -DDEC_ALPHA -O3 -Mcray=pointer -Mnoframe -byteswapio
#
### need to point to header and libs for mpich explicitly for XD1
#OBJS_PATH = /opt/benchmark/shome/CONTRIB
#LOCAL_OBJS = $(OBJS_PATH)/if.o $(OBJS_PATH)/strdup.o $(OBJS_PATH)/farg.o
#LIB_PATH = -L $(PGI)/linux86-64/5.1/lib -L $(LINUX_MPIHOME)/lib -L /lib64
#LOCAL_LIBRARIES = $(LIB_PATH) -lgcc -lmpich -lfmpich -lrapl -lmpichfsup -
lpthread $(LOCAL_OBJS)
#
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P -traditional
#CPPFLAGS = -DDEC_ALPHA -DMPI -Dlinux -DSYSTEM_CALL_OK
#CFLAGS = -O3 -DDEC_ALPHA -DMPI -I$(LINUX_MPIHOME)/include
#ARCH_OBJS = milliclock.o

```

---

```
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#-----
# Don't touch anything below this line
#-----
.F.i:
$(RM) $@
$(CPP) $(CPPFLAGS) $*.F > $@
mv $*.i $(DEVTOP)/pick/$*.f
cp $*.F $(DEVTOP)/pick

.C.o:
$(RM) $@ && \
$(CC) -c $(CFLAGS) $*.c

.F.o:
$(RM) $@
$(FC) -c $(FCFLAGS) $*.F

.F.f:
$(RM) $@
$(CPP) $(CPPFLAGS) $*.F > $@

.f.o:
$(RM) $@
$(FC) -c $(FCFLAGS) $*.f
```

## 8.18 mm5.deck

**This is a Bourne shell script. Slight variations may exist on different machines.**

(This file is included here for reference only. Use the most up-to-date one from MM5.TAR file.)

```
#!/bin/sh
#
# Version 3 of mm5 job deck
#
# The mm5 executable (mm5.exe) expects to find the following files
# in the Run/ directory:
#   MMINPUT_DOMAIN1 - |
#   BDYOUT_DOMAIN1   | --> output files from Interpf
#   LOWBDY_DOMAIN1   - |
#   TERRAIN_DOMAIN[2,3..] if running nests --> output from Terrain
#
# If it is a restart run:
#   RESTART_DOMAIN1[,2,3..] --> output from MM5 run: renamed from
#                               SAVE_DOMAIN1[,2,3...]
#
# If it is gridded FDDA run with surface analysis nudging:
#   SFCFDDA_DOMAIN1[2,3,...]
#
# If it is observational nudging run:
#   MM5OBS_DOMAIN1[,2,3..] --> user-created observation files
#
# Output from a MM5 run:
#   If IFTAPE = 1
```



[illegible]



```

;      (only if snow data are generated in REGRID), 2 (simple snow model -
;      only if WEASD is provided by REGRID)
ISFMTHD= 1, ;method for calculation of 2m/10m diagnostics
;0 - old method, 1 - new method for stable conditions
IZOTOPT = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ;Thermal roughness length option
;      - 0 (default), 1 (Garratt), 2 (Zilitinkevich)
ISFFLX = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ;surface fluxes - 0, 1
ITGFLG = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ;surface temperature prediction -
;      1:yes, 3:no
ISFPAR = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ;surface characteristics - 0, 1
ICLOUD = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ;cloud effects on radiation - 0, 1
;      currently for IFRAD = 1,2
IEVAP = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ;evap of cloud/rainwater - <0, 0, >0
;      (currently for IMPHYS=3,4,5 only)
ISMIRD = 0, ;soil moisture initialization by PX LSM:
;      =0, use moisture avail from LANDUSE.TBL;
;      =2, use soil moisture from REGRID;
;
; Default soil layers expected as input for ISOIL 2 & 3
; These values reflect the BOTTOM of the soil layer available
ISTLYR = 10,40,100,200,
ISMLYR = 10,40,100,200,
;ISTLYR = 10,200,0,0,
;ISMLYR = 10,200,0,0,
; Other common layers used by EC models (for instance ERA40)
;ISTLYR = 7,28,100,255,
;ISMLYR = 7,28,100,255,
;
; Next two switches for new version of NOAH LSM (ISOIL=2)
RDMAXALB = .FALSE., ; T: use climatological max snow albedo
RDBRDALB = .FALSE., ; T: use climatological monthly albedo
;      (not landuse table)
EOF
cat > ./Run/nparam << EOF
&NPARAM
;
;      ***** NEST AND MOVING NEST OPTIONS *****
;
LEVIDN = 0,1,2,1,1,1,1,1,1,1,1, ; level of nest for each domain
NUMNC = 1,1,2,1,1,1,1,1,1,1,1, ; ID of mother domain for each nest
NESTIX = 35, 49, 31, 46, 46, 46, 46, 46, 46, 46, 46, ; domain size i
NESTJX = 41, 52, 31, 61, 61, 61, 61, 61, 61, 61, 61, ; domain size j
NESTI = 1, 10, 8, 1, 1, 1, 1, 1, 1, 1, 1, ; start location i
NESTJ = 1, 17, 9, 1, 1, 1, 1, 1, 1, 1, 1, ; start location j
XSTNES = 0., 0.,900., 0., 0., 0., 0., 0., 0., 0., ; domain initiation
XENNES =1440.,1440.,1440.,720.,720.,720.,720.,720.,720.,720.;domain termination
IOVERW = 1, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, ; overwrite nest input
;      0=interpolate from coarse mesh (for nest domains);
;      1=read in domain initial conditions
;      2=read in nest terrain file
IACTIV = 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ;
;      in case of restart: is this domain active?
;
;      ***** MOVING NEST OPTIONS *****
;
IMOVE = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ; move domain 0,1
IMOVCO = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ; 1st move #

```

```

;          imovei(j,k)=L, ; I-INCREMENT MOVE (DOMAIN J, MOVE NUMBER K) IS L
IMOVEI =   0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; I move #1
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; I move #2
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; I move #3
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; I move #4
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; I move #5
IMOVEJ =   0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; J move #1
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; J move #2
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; J move #3
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; J move #4
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; J move #5
IMOVET =   0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; time of move #1
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; time of move #2
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; time of move #3
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; time of move #4
           0,   0,   0,   0,   0,   0,   0,   0,   0,   0, ; time of move #5
IFEED  =   3,          ; no feedback; 9-pt weighted average; 1-pt feedback w/o smoothing /
                    ; light smoothing / heavy smoothing - 0,1,2,3, and 4
&END
EOF
cat > ./Run/pparam << EOF
&PPARAM
;
;          ***** MISCELLANEOUS OPTIONS *****
;
; The values for the following 5 variables are only used if ISFPAR = 0
; (i.e. only land/water surface catagories)
;
ZZLND  = 0.1,          ; roughness length over land in meters
ZZWTR  = 0.0001,       ; roughness length over water in meters
ALBLND = 0.15,         ; albedo
THINLD = 0.04,         ; surface thermal inertia
XMAVA  = 0.3,          ; moisture availability over land as a decimal fraction of one
;
CONF   = 1.0,          ; non-convective precipitation saturation threshold (=1: 100%)
&END
EOF
cat > ./Run/fparam << EOF
&FPARAM
;
;          ***** 4DDA OPTIONS *****
;
; THE FIRST DIMENSION (COLUMN) IS THE DOMAIN IDENTIFIER:
; COLUMN 1 = DOMAIN #1, COLUMN 2 = DOMAIN #2, ETC.
;
; START TIME FOR FDDA (ANALYSIS OR OBS) FOR EACH DOMAIN
; (IN MINUTES RELATIVE TO MODEL INITIAL TIME)
FDASTA=0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.
; ENDING TIME FOR FDDA (ANALYSIS OR OBS) FOR EACH DOMAIN
; (IN MINUTES RELATIVE TO MODEL INITIAL TIME)
FDAEND=780.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.
;
;          ***** ANALYSIS NUDGING *****
;
; THE FIRST DIMENSION (COLUMN) OF THE ARRAYS DENOTES THE

```

```

;      DOMAIN IDENTIFIER:
;      COLUMN 1 = DOMAIN #1, COLUMN 2 = DOMAIN #2, ETC.
;      THE SECOND DIMENSION (ROW OR LINE) EITHER REFERS TO THE 3D VS
;      SFC ANALYSIS OR WHICH VARIABLE IS ACCESSED:
;      LINE 1 = 3D, LINE 2 = SFC    OR
;      LINE 1 = U, LINE 2 = V, LINE 3 = T, LINE 4 = Q
;
;      IS THIS A GRID 4DDA RUN? 0 = NO; 1 = YES
I4D= 0,0,0,0,0,0,0,0,0,0,
      0,0,0,0,0,0,0,0,0,0,
;
;      SPECIFY THE TIME IN MINUTES BETWEEN THE INPUT (USUALLY
;      FROM INTERP) USED FOR GRID FDDA
DIFTIM=720.,720.,0.,0.,0.,0.,0.,0.,0.,0.,      ; 3D ANALYSIS NUDGING
      180.,180.,0.,0.,0.,0.,0.,0.,0.,0.,      ; SFC ANALYSIS NUDGING
;
;      GRID NUDGE THE WIND FIELD? 0 = NO; 1 = YES
IWIND=1,1,0,0,0,0,0,0,0,0,      ; 3D ANALYSIS NUDGING
      1,1,0,0,0,0,0,0,0,0,      ; SFC ANALYSIS NUDGING
;
;      NUDGING COEFFICIENT FOR WINDS ANALYSES
GV=2.5E-4,1.0E-4,0.,0.,0.,0.,0.,0.,0.,0.,      ; 3D ANALYSIS NUDGING
      2.5E-4,1.0E-4,0.,0.,0.,0.,0.,0.,0.,0.,      ; SFC ANALYSIS NUDGING
;
;      GRID NUDGE THE TEMPERATURE FIELD? 0 = NO; 1 = YES
ITEMP=1,1,0,0,0,0,0,0,0,0,      ; 3D ANALYSIS NUDGING
      1,1,0,0,0,0,0,0,0,0,      ; SFC ANALYSIS NUDGING
;
;      NUDGING COEFFICIENT FOR TEMPERATURE ANALYSES
GT=2.5E-4,1.0E-4,0.,0.,0.,0.,0.,0.,0.,0.,      ; 3D ANALYSIS NUDGING
      2.5E-4,1.0E-4,0.,0.,0.,0.,0.,0.,0.,0.,      ; SFC ANALYSIS NUDGING
;
;      GRID NUDGE THE ROTATIONAL WIND FIELD? 0 = NO; 1 = YES
IROT=0,0,0,0,0,0,0,0,0,0,      ; 3D ANALYSIS NUDGING
;
;      NUDGING COEFFICIENT FOR THE MIXING RATIO ANALYSES
GQ=1.E-5,1.E-5,0.,0.,0.,0.,0.,0.,0.,0.,      ; 3D ANALYSIS NUDGING
      1.E-5,1.E-5,0.,0.,0.,0.,0.,0.,0.,0.,      ; SFC ANALYSIS NUDGING
;
;      GRID NUDGE THE ROTATIONAL WIND FIELD? 0 = NO; 1 = YES
IROT=0,0,0,0,0,0,0,0,0,0,      ; 3D ANALYSIS NUDGING
;
;      NUDGING COEFFICIENT FOR THE ROTATIONAL COMPONENT OF THE WINDS
GR=5.E6,5.E6,0.,0.,0.,0.,0.,0.,0.,0.,      ; 3D ANALYSIS NUDGING
;
;      IF GRID NUDGING (I4D(1,1)=1) AND YOU WISH TO EXCLUDE THE
;      BOUNDARY LAYER FROM FDDA OF COARSE GRID THREE DIMENSIONAL
;      DATA (USUALLY FROM INTERP),
;      0 = NO,    INCLUDE BOUNDARY LAYER NUDGING
;      1 = YES,   EXCLUDE BOUNDARY LAYER NUDGING
INONBL =0,0,0,0,0,0,0,0,0,0,      ; U WIND
      0,0,0,0,0,0,0,0,0,0,      ; V WIND
      1,1,1,1,1,1,1,1,1,1,      ; TEMPERATURE
      1,1,1,1,1,1,1,1,1,1,      ; MIXING RATIO
;
;      RADIUS OF INFLUENCE FOR SURFACE ANALYSIS (KM).
;      IF I4D(2,1)=1 OR I4D(2,2)=1, ETC, DEFINE RINBLW (KM) USED

```

```
;      IN SUBROUTINE BLW TO DETERMINE THE HORIZONTAL VARIABILITY
;      OF THE SURFACE-ANALYSIS NUDGING AS A FUNCTION OF SURFACE
;      DATA DENSITY.  OVER LAND, THE STRENGTH OF THE SURFACE-
;      ANALYSIS NUDGING IS LINEARLY DECREASED BY 80 PERCENT AT
;      THOSE GRID POINTS GREATER THAN RINBLW FROM AN OBSERVATION
;      TO ACCOUNT FOR DECREASED CONFIDENCE IN THE ANALYSIS
;      IN REGIONS NOT NEAR ANY OBSERVATIONS.
RINBLW=250.,
;
;      SET THE NUDGING PRINT FREQUENCY FOR SELECTED DIAGNOSTIC
;      PRINTS IN THE GRID (ANALYSIS) NUDGING CODE (IN CGM
;      TIMESTEPS)
NPFG=50,
;
;      ***** OBSERVATION NUDGING *****
;
;
;      INDIVIDUAL OBSERVATION NUDGING.  VARIABLES THAT ARE ARRAYS
;      USE THE FIRST DIMENSION (COLUMN) AS THE DOMAIN IDENTIFIER:
;      COLUMN 1 = DOMAIN #1, COLUMN 2 = DOMAIN #2, ETC.
;
;      IS THIS INDIVIDUAL OBSERVATION NUDGING? 0 = NO; 1 = YES
I4DI  =0,0,0,0,0,0,0,0,0,0,
;
;      OBS NUDGE THE WIND FIELD FROM STATION DATA? 0 = NO; 1 = YES
ISWIND =1,0,0,0,0,0,0,0,0,0,
;
;      NUDGING COEFFICIENT FOR WINDS FROM STATION DATA
GIV  =4.E-4,4.E-4,0.,0.,0.,0.,0.,0.,0.,0.,
;
;      OBS NUDGE THE TEMPERATURE FIELD FROM STATION DATA? 0 = NO; 1 = YES
ISTEMP=1,0,0,0,0,0,0,0,0,0,
;
;      NUDGING COEFFICIENT FOR TEMPERATURES FROM STATION DATA
GIT  =4.E-4,4.E-4,0.,0.,0.,0.,0.,0.,0.,0.,
;
;      OBS NUDGE THE MIXING RATIO FIELD FROM STATION DATA? 0 = NO; 1 = YES
ISMOIS=1,0,0,0,0,0,0,0,0,0,
;
;      NUDGING COEFFICIENT FOR THE MIXING RATIO FROM STATION DATA
GIQ  =4.E-4,4.E-4,0.,0.,0.,0.,0.,0.,0.,0.,
;
;      THE OBS NUDGING RADIUS OF INFLUENCE IN THE
;      HORIZONTAL IN KM FOR CRESSMAN-TYPE DISTANCE-WEIGHTED
;      FUNCTIONS WHICH SPREAD THE OBS-NUDGING CORRECTION
;      IN THE HORIZONTAL.
RINXY=240.,
;
;      THE OBS NUDGING RADIUS OF INFLUENCE IN THE
;      VERTICAL IN SIGMA UNITS FOR CRESSMAN-TYPE DISTANCE-
;      WEIGHTED FUNCTIONS WHICH SPREAD THE OBS-NUDGING
;      CORRECTION IN THE VERTICAL.
RINSIG=0.001,
;
;      THE HALF-PERIOD OF THE TIME WINDOW, IN MINUTES, OVER
```

---

```

;      WHICH AN OBSERVATION WILL AFFECT THE FORECAST VIA OBS
;      NUDGING. THAT IS, THE OBS WILL INFLUENCE THE FORECAST
;      FROM TIMEOBS-TWINDO TO TIMEOBS+TWINDO.  THE TEMPORAL
;      WEIGHTING FUNCTION IS DEFINED SUCH THAT THE OBSERVATION
;      IS APPLIED WITH FULL STRENGTH WITHIN TWINDO/2. MINUTES
;      BEFORE OR AFTER THE OBSERVATION TIME, AND THEN LINEARLY
;      DECREASES TO ZERO TWINDO MINUTES BEFORE OR AFTER THE
;      OBSERVATION TIME.
TWINDO=40.0,
;
;      THE NUDGING PRINT FREQUENCY FOR SELECTED DIAGNOSTIC PRINT
;      IN THE OBS NUDGING CODE (IN CGM TIMESTEPS)
NPFI=20,
;
;      FREQUENCY (IN CGM TIMESTEPS) TO COMPUTE OBS NUDGING WEIGHTS
IONF=2,
IDYNIN=0, ;for dynamic initialization using a ramp-down function to gradually
;      turn off the FDDA before the pure forecast, set idynin=1 [y=1, n=0]
DTRAMP=60.,;the time period in minutes over which the
;      nudging (obs nudging and analysis nudging) is ramped down
;      from one to zero.  Set dtramp negative if FDDA is to be ramped
;      down BEFORE the end-of-data time (DATEND), and positive if the
;      FDDA ramp-down period extends beyond the end-of-data time.
&END
EOF
#
#-----
#
#      create namelist: mmlif, and remove comments from namelist:
#
make mmlif
cd ./Run
sed -f ../Util/no_comment.sed mmlif | grep [A-Z,a-z] > mmlif.tmp
mv mmlif.tmp mmlif
rm fparam lparam nparam oparam pparam
#
#      copy gridded FDDA files
#
if [ $FDDAsw = yes ]; then
    echo "Copy grid fdda file"
    for i in MMINPUT_DOMAIN[1-9]
    do
        Num=`echo $i | grep [1-9]$ | sed 's/.*\(.\)\/\1/'`
        cp $i MMINPUT2_DOMAIN$Num
        echo "cp $i MMINPUT2_DOMAIN$Num"
    done
fi
#
#-----
#
#      run MM5
#
date
echo "timex mm5.exe >! mm5.print.out "
timex ./mm5.exe > mm5.print.out 2>&1

```

---

## MM5 FORTRAN FILES

