9 MM5

9.1 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.

9.2 Basic Equations of MM5

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

Pressure

$$\frac{\partial p'}{\partial t} - \rho_0 g w + \gamma p \nabla \cdot \mathbf{v} = -\mathbf{v} \cdot \nabla p' + \frac{\gamma p}{T}\left(\frac{\dot{Q}}{c_p} + \frac{T_0}{\theta_0} D_\theta\right), \tag{9.1}$$

Momentum (x-component)

$$\frac{\partial u}{\partial t} + \frac{1}{\rho}\left(\frac{\partial p'}{\partial x} - \frac{\sigma}{p^*}\frac{\partial p^*}{\partial x}\frac{\partial p'}{\partial \sigma}\right) = -\mathbf{V} \cdot \nabla u + v\left(f + u\frac{\partial m}{\partial y}\right) - ew - \frac{uw}{r_{earth}} + D_u, \quad (9.2)$$

Momentum (y-component)

$$\frac{\partial v}{\partial t} + \frac{1}{\rho}\left(\frac{\partial p'}{\partial y} - \frac{\sigma}{p^*}\frac{\partial p^*}{\partial y}\frac{\partial p'}{\partial \sigma}\right) = -\mathbf{V} \cdot \nabla v - u\left(f + u\frac{\partial m}{\partial y}\right) - \frac{vw}{r_{earth}} + D_v, \quad (9.3)$$

Momentum (z-component)

$$\frac{\partial w}{\partial t} + \frac{\rho_0}{\rho}\frac{g}{p^*}\frac{\partial p'}{\partial \sigma} + \frac{g}{\gamma}\frac{p'}{p} = -\mathbf{V} \cdot \nabla w + g\frac{p_0}{p}\frac{T'}{T_0} - \frac{gR_d}{c_p}\frac{p'}{p} + eu + \frac{u^2 + v^2}{r_{earth}} + D_w, \quad (9.4)$$

Thermodynamics

$$\frac{\partial T}{\partial t} = -\mathbf{V} \cdot \nabla T + \frac{1}{\rho c_p}\left(\frac{\partial p'}{\partial t} + \mathbf{V} \cdot \nabla p' - \rho_0 g w\right) + \frac{Q}{c_p} + \frac{T_0}{\theta_0}D_\theta. \quad (9.5)$$

Advection terms can be expanded as

$$\mathbf{V} \cdot \nabla A \equiv u\frac{\partial A}{\partial x} + v\frac{\partial A}{\partial y} + \dot{\sigma}\frac{\partial A}{\partial \sigma}, \quad (9.6)$$

where

$$\dot{\sigma} = -\frac{\rho_0 g}{p^*}w - \frac{\sigma}{p^*}\frac{\partial p^*}{\partial x}u - \frac{\sigma}{p^*}\frac{\partial p^*}{\partial y}v. \quad (9.7)$$

Divergence term can be expanded as

$$\nabla \cdot \mathbf{V} = \frac{\partial u}{\partial x} - \frac{\sigma}{p^*}\frac{\partial p^*}{\partial x}\frac{\partial u}{\partial \sigma} + \frac{\partial v}{\partial y} - \frac{\sigma}{p^*}\frac{\partial p^*}{\partial y}\frac{\partial v}{\partial \sigma} - \frac{\rho_0 g}{p^*}\frac{\partial w}{\partial \sigma}. \quad (9.8)$$

Notes about the equations:

- Appendix A shows derivations of Equations 9.1, 9.4, 9.5 and 9.7, and shows the coordinate transformation from z to sigma coordinates.

- In the model, Equation 9.1 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 9.2-9.4 include terms (eu and ew) representing the usually neglected component of the Coriolis force. These equations assume y points north, but in the model these Coriolis terms are rotated according to the local grid direction.

- The $u\frac{\partial m}{\partial y}$ and r_{earth} terms represent curvature effects, and m is map-scale factor.
- Equations 9.2, 9.3 and 9.8 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.

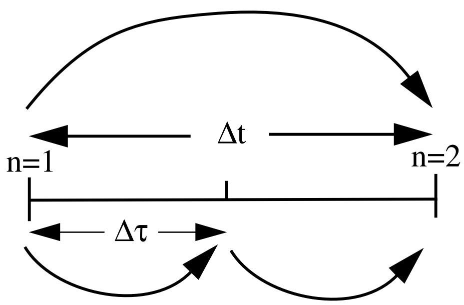
Temporal finite differencing -

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 9.1-9.4 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 9.1-9.4 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.

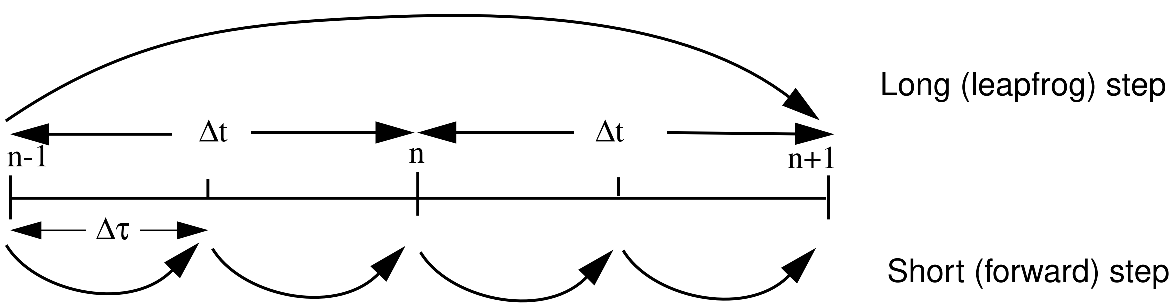
Certain processes are treated implicitly for numerical stability. An implicit time scheme is one in which the tendencies of variables depend not only on the present and past values, but also the future values. These schemes are often numerically stable for all timesteps, but usually require a matrix 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:



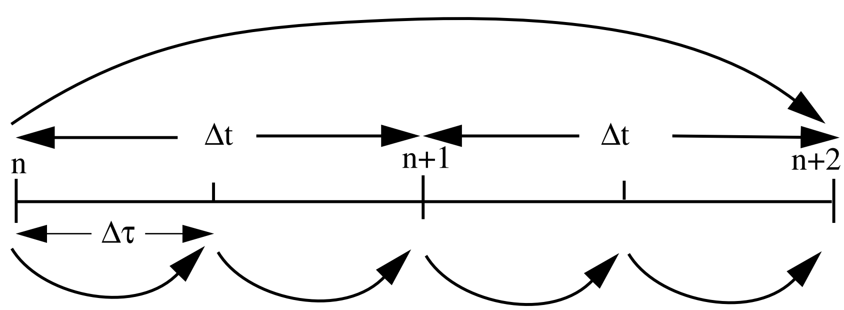
Time step n:

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



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

Time step n+1:



9.3 Physics Options in MM5

9.3.1 Cumulus Parameterizations (ICUPA)

1. None -

Use no cumulus parametrization at grid sizes < 5-10 km.

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 10-30 km, tends to allow a balance between resolved scale rainfall and convective rainfall.

4. Arakawa-Schubert -

multi-cloud scheme that is otherwise likeGrell 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.

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) and Fritsch and Kain (1993) for details.

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.

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.

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.

9.3.2 PBL Schemes (IBLTYP)

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 for surface temperature and has to use ISOIL=1 because of its long time step. Its cost is between the MRFPBL and HIRPBL schemes. Before SLAB the scheme calculates exchange coefficients using similarity theory, and after SLAB 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 also calls the SLAB routine and should also have ISOIL=1. Vertical diffusion uses an implicit scheme to allow longer time steps.

6. Gayno-Seaman PBL - (available in release 2.13 and Version 3)

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). Its cost is comparable with the Blackadar scheme’s because it uses split time steps.

9.3.3 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) -

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. (1993, 1996) for details. 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.

7. Reisner graupel -

Based on mixed-phase scheme but adding graupel and ice number concentration prediction equations. Also suitable for cloud-resolving models.

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.

9.3.4 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.

3. CCM2 radiation scheme -

Multiple spectral bands in shortwave and longwave, but cloud treated simply based on RH. 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.

9.3.5 Ground Temperature 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. OSU/Eta Land-Surface Model (available in Version 3)

See Version 3 chapter. Not available in MM5 Version 2.

9.4 Four-Dimensional Data Assimilation (FDDA)

9.4.1 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.

9.4.2 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.

9.4.3 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.

9.4.4 Data used in FDDA

Analysis nuding -

When doing three-dimensional analysis nudging, no additional input data files are required. The MM5 job deck simply copies the mminput file to a different fortran unit for use in FDDA. 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.

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
RJO:	x-location - J dot-point location on coarse mesh
RKO:	z-location - K half-σ level
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.

9.5 How to run MM5

There are 2 steps to compiling and running the MM5 (V2) system.

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

9.5.1 Compiling MM5

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

(see 9.5.3 for running batch job on Cray.)

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 4 things.

- Set the RUNTIME_SYSTEM variable. This should indicate what kind of machine you are running on (e.g., CRAY).
- Pick the Fortran compiler options appropriate for your system. Comment out the default option (which is for Cray), and uncomment the ones for your system.
- 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, dynamics and physics option for (selective) compiling purposes.

When finished editing, just type 'make' to compile the code.

If you wish to compile the model on a PC with Linux OS,
- copy the *configure.user.linux* file to *configure.user;*
- type 'make'

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),
- 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 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, and on Web page: *http://www.mmm.ucar.edu/mm5/mm5v2-mpp.html*

9.5.2 Running MM5

- make 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

- run the “mm5.deck” script by typing ‘mm5.deck’.

9.5.3 Running MM5 Batch Job on Cray

- If you want to work in batch mode, whether to compile and/or execute, get a copy of mm5.deck from mesouser directory: *~mesouser/MM5V2/MM5* on ouray/paiute/chipeta, or anonymous ftp under */mesouser/MM5V2/MM5*. Or, you may get the deck once you obtain the mm5v2.tar file on your local machine. To do so, first un-tar the tar file, edit the *configure.user* file to define RUNTIME_SYSTEM=“CRAY”; then type ‘make mm5.deck’. This deck has an appearance of V1 deck, and has the *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 Cray, you can either use the above deck, or use the Cray interactive deck, by setting the RUNTIME_SYSTEM=“CRAY_IA”, and followed by typing ‘make mm5.deck’. The mm5.deck generated this way has an appearance of other workstations decks.

- 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.*

9.6 Input to MM5

Files from INTERP program

- Model input file(s)
- Boundary condition file for coarse mesh

Files from MM5 program, if it is a restart run

- Model restart file(s)

Files from RAWINS, if running gridded 4DDA using surface analysis

- FDDA surface analysis

Files generated by user, if running observational nudging

- FDDA 4D obs file(s)

Namelists: user-specified options

landuse.tbl: user-modifiable landuse characteristics (in ASCII)

Note that the workstation mm5.deck expects all input files to be in the Run directory. See the mm5.deck for details.

9.7 Output from MM5

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

- history files (fortran units 40's), if IFTAPE = 1, and the output frequency is set by TAPFRQ.
- restart files (fortran units 50's), if IFSAVE = .TRUE., and the output frequency is set by SAVFRQ.

Output from each domain will be written to different fortran units. For example, domain 1's history file is written to fortran unit 41, and its restart file to fortran unit 51. Each output file contains data for all output times for that domain. On NCAR's Crays, if a model output file exceeds 6,000 Mb, split.deck from mesouser directory will split it into smaller files. If the restart files exceed 6,000 Mb the bsplit utility will split the binary file (cat can be used to re-join it).

For each time period the model history output includes:

- A header record describing the output
- 3D forecast arrays dimensioned (IX, JX, KX) for that domain (half levels unless specified) - note that the variables are coupled (i.e., multiplied by Pstar)

1	U-wind (kPa m/s)	coupled	dot
2	V-wind (kPa m/s)	coupled	dot
3	Temperature (kPa K)	coupled	cross
4	Water vapor mixing ratio (kPa kg/kg) (if IMPHYS≥2)	coupled	cross
5	Cloud water mixing ratio (kPa kg/kg) (if IMPHYS≥3)	coupled	cross
6	Rain water mixing ratio (kPa kg/kg) (if IMPHYS≥3)	coupled	cross
7	Ice cloud mixing ratio (kPa kg/kg) (if IMPHYS≥5)	coupled	cross
8	Snow mixing ratio (kPa kg/kg) (if IMPHYS≥5)	coupled	cross
9	Graupel (kPa kg/kg) (if IMPHYS 6)	coupled	cross
10	Number concentration of ice (if IMPHYS 6)	coupled	cross
11	Turbulent k.e. (J/kg) (if IBLTYP=3,4,6)	not coupled	cross
12	Atmospheric radiation tendency (K/day) (if FRAD≥2)	not coupled	cross
13	Vertical velocity (kPa m/s) full levels (if INHYD=1)	coupled	cross
14	Perturbation pressure (kPa Pa) (if INHYD=1)	coupled	cross

- 2D forecast arrays dimensioned (IX, JX)

1	Pstar (cb)	cross
2	Ground temperature (K)	cross
3	Accum. convective rainfall (cm)	cross
4	Accum. nonconv. rainfall (cm)	cross
5	PBL height (m)	cross
6	PBL regime (catagory, 1-4)	cross
7	Surface sensible heat flux (W/m^2)	cross
8	Surface latent heat flux (W/m^2)	cross
9	Frictional velocity (m/s)	cross
10	Surface downward shortwave radiation (W/m^2)	cross
11	Surface downward longwave radiation (W/m^2)	cross
12	Soil temperature in layers 1 - 6 (K)	cross

- 2D constant arrays dimensioned (IX, JX)

13	Terrain elevation (m)	cross
14	Map scale factor	cross
15	Map scale factor	dot
16	Coriolis parameter (/s)	dot
17	Substrate temperature (K)	cross
18	Latitude (deg)	cross
19	Longitude (deg)	cross
20	Land-use category	cross
21	Snow cover	cross

The output file size per time period assuming 8 bytes per word for Cray:
3.52 Mb (for the header) + n3d*IX*JX*KX*8 + n2d*IX*JX*8
Where n3d is the number of 3d arrays and n2d is the number of 2d arrays.

(On 32-bit workstation, the header size is 3.36 Mb)

9.8 MM5 Files and Unit Numbers

The data for each domain is written to a separate file. MM5 accesses most files by referring to the fortran unit number. Unit number are assigned as follows:

Table 9.1 Shell names, fortran unit numbers, MSS names and their description for MM5.

Shell name	Unit number	Description (MSS name)
mmlif	fort.10	Input, namelist file
ehtran	fort.8	Input, emissivity file
bdyout	fort.9	Input, boundary files created by program INTERP ($InitName/BDYOUT_DOMAIN1)
landuse.tbl	fort.19	Input, physical properties for landuse categories
	fort.11, 12, ... 19	Input, initial files created by program INTERP ($InitName/MMINPUT_DOMAIN1, 2, ... 9); or Terrain output files for fine meshes
	fort.31, 31, ... 39	Input, 3D analysis nudging files (same as initial files)
	fort.71, 72, ... 79 (fort.81, 82, ... 89)	Input, surface analysis nudging files created by program RAWINS ($ExpName/RW4DDA_DOMAIN1, 2, ... 9)
	fort.61,62, ... 69	Input, observation nudging files created by user's own program
	fort.91, 92 ... 99	Input, restart files ($InitName/${CaseName}/SAVE_DOMAIN1, 2, ... 9)
	fort.41,42, ... 49	MM5 model output files ($OutMM/MMOUT_DOMAIN1, 2, ... 9)
	fort.51, 52, ..., 59	Output, restart or save files ($OutMM/SAVE_DOMAIN1, 2, ... 9)
	fort.61, 62, ..., 69	Output, shutdown restart files ($OutMM/SHUTDO_DOMAIN1, 2, ... 9)

9.9 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 precompilation variables. Sections 1, 4 and make rules will be explained in Chapter 10.

RUNTIME_SYSTEM	computer system to run model on.
NHYDRO	=1, for nonhydrostatic run; =2 for hydrostatic run.
FDDAGD	=1, for 4DDA grid analysis nudging; =0, no 4DDA.
FDDAOB	=1, for 4DDA observation nudging; =0, no obs 4DDA.
MAXNES	maximum number of domains in simulation. Note though, there are only 4 default nest levels.
MIX,MJX,MKX	maximum number of grid points in I, J, and K.
IMPHYS	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)
MPHYSTBL	=1, use look-up table version of explicit scheme options 4 and 5; =0, not using look-up table version.
ICUPA	options for cumulus parameterization schemes: =1, none; =2, Anthes-Kuo; =3, Grell; =4, Arakawa-Schubert; =5, Fritsch-Chappell; =6, Kain-Fritsch; =7, Betts-Miller.
IBLTYP	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 (release-2-13 and Version 3)
FRAD	options for atmospheric radiation schemes: =0, none; =1, simple cooling;

	=2, cloud (Dudhia) (require IMPHYS $\geq$ 3); =3, CCM2.
ISOIL	=1, use the multi-layer soil model (require IBLTYP=2 or 5); =0, no soil model.
ISHALLO	=1, use shallow convective scheme; =0, no.

9.10 Script Variables

Script variables are found in mm5 decks.

9.10.1 Variables used in Cray decks only:

NCPUS	number of processors to use (if NCPUS>1, use with -Zu in compiler option).
ExpName	experiment name used in setting MSS pathname.
InName	input MSS pathname.
RetPd	mass store retention period (days).
Host	= username@host.domain:/usr/tmp/username, host computer to rcp user's mods.
compile	=yes, compile the mm5 code; =no, expect an existing executable.
execute	=yes, execute the model; =no, compile the code only.
UseMySource	=yes, use your own source code; =no, use mesouser version of the source code.
CaseName	MSS pathname for this run.
OutMM	MSS name for output

9.10.2 Variables used in both Cray and workstation decks:

STARTsw	= NoReStart: start model run at hour zero (initialize). = ReStart: restart model run.
FDDAsw	= NoFDDA, no FDDA input files, = Anly, gridded FDDA input files, = Obs, obsFDDA input files,

	= Both, gridded and obs FDDA input files.
HYDROsw	= Hydro, hydrostatic input files,
	= NonHydro, nonhydrostatic input files.
InBdy	MSS name of boundary file.
InMM	MSS name(s) of model input files.
InRst	MSS name(s) of model restart files.
In4DSfc	MSS name of surface analysis used for 4DDA.
In4DObs	MSS name of fdda obs files.

9.11 Namelist Variables

A namelist file, called *mmlif*, is created when *mm5.deck* is executed. In MM5 V2, this file is created partially from the *configure.user* file, and partially from *mm5.deck*.

9.11.1 OPARAM

IFREST	=TRUE, for restart, =FALSE, for initialization
IXTIMR	= integer time in minutes for restart
LEVIDN	= level of nest for each domain (0 for domain 1)
NUMNC	= id number of parent domain for each domain (1 for domain 1)
IFSAVE	=TRUE, if saving data for restart, = FALSE, for no restart output
SVLAST	= TRUE, if only saving the last time; = FALSE, save multiple times
SAVFRQ	= frequency of restart output in minutes
IFTAPE	=1, for model output; =0, no model output
TAPFRQ	= frequency of model output in minutes
INCTAP	= multipliers of TAPFRQ for outputting
IFSKIP	=TRUE, skip input files to start the model - DO NOT use this when restart
MDATEST	= MDATE (yymmddhh) of the start file
IFPRT	=1, for printed output fields,
	=0, for no printed output fields
PRTFRQ	= frequency of printed output fields in minutes
MASCHK	= integer frequency in number of time steps for budget/rainfall prints (coarsest mesh)

9.11.2 LPARAM

1) Defined in *mm5.deck*:

RADFRQ = frequency in minutes of radiation calculations (surface and atmospheric)

IMVDIF =1, for moist vertical diffusion in clouds (requires IMPHYS>2, and IBLTYP=2 or 5),

=0, vertical diffusion is dry

IVQADV =0, vertical moisture advection uses log interpolation (old method),

=1, vertical moisture advection uses linear interpolation (affects all moisture variables)

IVTADV =0, vertical temperature advection uses log interpolation (old method),

=1, vertical temperature advection uses linear interpolation

ITHADV =0, temperature advection and adiabatic term use temperature (old method),

=1, temperature advection and adiabatic term use potential temperature

ITPDIF =1, for diffusion using perturbation temperature in NH model;

=0, not using this function (new in V2)

ICOR3D =1, for full 3D Coriolis force (requires INHYD=1),

=0, for traditional approximation.

IFUPR =1, for upper radiative boundary condition (NH run only).

=0, rigid upper boundary in nonhydrostatic runs.

IFDRY =1, for fake dry run with no latent heating release (requires IMPHYS>1, and ICUPA=1)

ICUSTB =1, stability check in Anthes-Kuo convective scheme;

=0, do not use stability check in Anthes-Kuo scheme.

IBOUDY Boundary condition options:

=0, fixed,

=1, relaxation (hydrostatic run only),

=2, time-dependent (recommend all 2-way nests use this option),

=3, relaxation inflow/outflow (NH run) or inflow/outflow (hydrostatic run),

=4, sponge (hydrostatic run only).

IFSNOW =1, snow cover effects (requires input SNOWC field from DATAGRID, and ITGFLG=1

ISFFLX =1, compute surface heat and moisture fluxes; =0, no fluxes.

ITGFLG =1, ground temperature predicted;

=3, constant ground temperature.

ISFPAR =1, use 13 land-use categories;

=0, use only 2 (land/water) categories.

ICLOUD =1, consider cloud effects on surface radiation when FRAD=0,1 ; consider both surface and atmospheric radiation when FRAD=2 (requires ITGFLG=1, ISFFLX=1);

=0, do not consider cloud effect on radiation.

ICDCON =0, drag coefficients vary with stability;

=1, constant drag coefficients (requires IBLTYP=1)

IVMIXM =1, vertical mixing of momentum in mixed layer (requires IBLTYP=2);

=0, no mixing.

HYDPRE =1, consider water loading effects in computing geopotential height in hydrostatic run (requires NHYDRO=0, IMPHYS>1)

IEVAP =1, normal evaporative cooling;

=0, no evaporative effects;

=-1, no precip evaporative cooling, (for IMPHYS=3,4, and 5).

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.

IMOVEJ = increment in J (parent domain grids) of this move for this domain.

IMOVET = 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.

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’

9.11.3 PPARAM

TIMAX	= forecast length in minutes.
TISTEP	= time step in seconds for the coarsest domain (recommend 3*dx(km))
ZZLND	= roughness length over land (m) (if ISFPAR=0)
ZZWTR	= roughness length over water (m) (if ISFPAR=0)
ALBLND	= albedo over land (if ISFPAR=0)
THILND	= thermal inertia of land (cal-1 cm-2 K-1 s-0.5, if ISFPAR=0)
XMAVA	= moisture availability over land (if ISFPAR=0)
CONF	= non-convective precip saturation criterior (fraction $\leq$ 1 for IMPHYS=1)
IFEED	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
IABSOR	=1, use sponge upper boundary condition (for hydrostatic run only); =0, none

9.11.4 FPARAM

FDASTA	(MAXSES); time (min) for initiation of FDDA.

FDAEND	(MAXSES); time (min) for termination of FDDA.
I4D	(MAXSES, 2); will FDDA analysis nudging be employed, (0=no; 1=yes).
DIFTIM	(MAXNES, 2); time (min) between input analyses for analysis nudging.
IWIND	(MAXSES, 2); will the wind field be nudged from analyses, (0=no; 1=yes).
GV	(MAXSES, 2); analysis-nudging coefficient (s^{-1}) for wind.
ITEMP	(MAXSES, 2); will the temperature be nudged from analyses, (0=no; 1=yes).
GT	(MAXSES, 2); analysis-nudging coefficient (s^{-1}) for temperature.
IMOIS	(MAXSES, 2); will the mixing ratio be nudged from analyses, (0=no; 1=yes).
GQ	(MAXSES, 2); analysis-nudging coefficient (s^{-1}) for mixing ratio.
IROT	(MAXSES); will vorticity be nudged from analyses, (0=no; 1=yes).
GR	(MAXSES, 2); analysis-nudging coefficient (m2 s^{-1}) for vorticity.
INONBL	(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).
RINBLW	radius of influence (km) for surface-analysis nudging where the horizontal weighting function depends on surface data density.
NPFG	coarse-grid time-step frequency for select diagnostic print of analysis nudging.
I4DI	(MAXSES); will FDDA observation nudging be employed, (0=no; 1=yes).
ISWIND	(MAXSES); will the wind field be nudged from observations, (0=no; 1=yes).
GIV	(MAXSES); observation-nudging coefficient (s^{-1}) for wind.
ISTEMP	(MAXSES); will the temperature be nudged from observations, (0=no; 1=yes).
GIT	(MAXSES); observation-nudging coefficient (s^{-1}) for temperature.
ISMOIS	(MAXSES); will the mixing ratio be nudged from observations, (0=no; 1=yes).
GIQ	(MAXSES); observation-nudging coefficient (s^{-1}) for mixing ratio.
RINXY	default horizontal radius of influence (km) for distance-weighted nudging corrections (for observation nudging).
RINSIG	vertical radius of influence (on sigma) for distance -weighted nudging corrections (for observationnudging).
TWINDO	(time window)/2 (min) over which an observation will be used for nudging.
NPFI	coarse-grid time-step frequency for select diagnostic print of observation nudging.
IONF	observation-nudging frequency in coarse grid time steps for observation-nudging calculations.
IDYNIN	for dynamic initialization using a ramp-down function to gradually turn off the FDDA before the pure forecast (1=yes, 0=no).
DTRAMP	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.

9.12 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 Cray job, 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:

- "Read past end-of-file": This is usually followed by a fortran unit number. Check this unit number with Table 9.1 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.
- "CPU limit exceeded": Increase the amount of time requested in the QSUB statements when running the job on a Cray.
- "Not enough space": Increase the amount of memory requested in the QSUB statements when running on a Cray.
- "Unrecognized namelist variable": This usually means there are typos in the namelist.
- Unmatched physics option: for instance, the following should appear in the 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
```

  They should match what one had in INTERP/GRIN deck.

9.13 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 how to run MM5
README.MPP	General information on how to compile and run on DM machines
ReleaseNotes/	Release notes and difference file between consecutive releases
Run/	Where MM5 runs
Templates/	MM5 job decks for different machines
Util/	Utility programs for cpp
configure.user	Make rules and model configuration
configure.user.linux	Make rules and model configuration for PC running Linux OS
domain/	
dynamics/	
fdda/	
include/	
memory/	
physics/	
pick/	

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

The model is executed in the directory *Run*.

The bug fixes and changes to the source code and tar file are provided in directory *ReleaseNotes*, and described in file *CHANGES*.

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/*).

9.14 Configure.user

```
# Sections
# 1. System Variables
# 2. User Variables
# 3. Fortran options
#    3a. Cray (YMP, J90)
#        Note: set RUNTIME_SYSTEM="CRAY_IA" for Cray interactive job
#    3b. IRIX.6.X (SGI_R8000,SGI_R10000,Origin200,Origin2000)
#   3b2. IRIX.6.X (use OpenMP parallel directives on SGI Origin)
#    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)
#    3g. HP (UX)
#    3h. HP (SPP-UX) for HP Exemplar S/X-Class Systems
# 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
#    7b. Cray T3E
#    7c. SGI Origin 2000
#    7d. HP Exemplar
#    7e. DEC ALPHA/MPI
#    7f. Fujitsu VPP
#    7g. Network of Linux PCs with MPI
#
#-----------------------------------------------------------------------------
# 1. System Variables
#-----------------------------------------------------------------------------
SHELL = /bin/sh
.SUFFIXES: .F .i .o .f .c
#-----------------------------------------------------------------------------
# 2. User Variables
#-----------------------------------------------------------------------------
# RUNTIME_SYSTEM          - Currently supported systems.
#                         SGI_R4000,SGI_R8000,CRAY,CRAY_IA,
#                         SUN,DEC_ALPHA,IBM,HP,,HP-SPP,HP-SPP_IA
#                         - MPP version:
#                         o2k, t3e, alpha, sp2, hp, vpp, linux
#                         Presently, of the MPP platforms only the "sp2"
#                         is supplied with the "make deck" capability.
RUNTIME_SYSTEM = "CRAY"
#
#-----------------------------------------------------------------------------
# 3. Fortran options
#-----------------------------------------------------------------------------
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 3.1.0.0 and above
#-----------------------------------------------------------------------------
# if you are running on paiute
#FC = cf77
```

```
#FCFLAGS = -D$(RUNTIME_SYSTEM) -I$(LIBINCLUDE) -Zu
# if you are running on ouray
FC = f90
FCFLAGS = -D$(RUNTIME_SYSTEM) -I$(LIBINCLUDE) -O task1 -x omp
CFLAGS =
CPP = /opt/ctl/bin/cpp
CPPFLAGS = -I$(LIBINCLUDE) -C -P
LDOPTIONS =
LOCAL_LIBRARIES = -L /usr/local/lib -l imsl
MAKE = make -i -r
#-----------------------------------------------------------------------------
#    3b. IRIX.6.X (SGI_R8000,SGI_R10000,Origin200,Origin2000)
#        - set RUNTIME_SYSTEM = SGI_R8000
#       - use the second LDOPTIONS if compiling Burk-Thompson PBL scheme
#        - use 7.0 and above compiler
#        - do not use -lfastm for R10000 and Origin series for compiler
#          versions 7.0 and 7.1, unless patches are installed. For more
#          information please see MM5 Web page:
#            http://www.mmm.ucar.edu/mm5/mm5v2-sgi.html
#-----------------------------------------------------------------------------
#FC = f77
#FCFLAGS=-I$(LIBINCLUDE) -O3 -n32 -mips4 -mp -OPT:roundoff=3:IEEE_arithmetic=3
#CFLAGS =
#CPP = /usr/lib/cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS = -n32 -mips4 -mp
##LDOPTIONS = -n32 -mips4 -mp -Wl,-Xlocal,bt1_,-Xlocal,blk1_,-Xlocal,blk2_
#LOCAL_LIBRARIES = -lfastm
##LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----------------------------------------------------------------------------
#   3b2. IRIX.6.X (SGI_Origin)
#        Use OpenMP directives for multi-processor runs.
#        - set RUNTIME_SYSTEM = SGI_Origin
#        - works with 7.2.1 and above compiler
#
#    - For parallel execution of MM5 set the following environment variables:
# setenv OMP_NUM_THREADS <number_of_processors>
# setenv _DSM_PLACEMENT ROUND_ROBIN
#    - For parallel execution on a processor set without contention:
# setenv _DSM_WAIT SPIN
# setenv OMP_DYNAMIC FALSE
# setenv MPC_GANG OFF
#    - For parallel execution on a contented set of processors:
# setenv _DSM_WAIT YEILD
# setenv OMP_DYNAMIC TRUE
# setenv MPC_GANG OFF
#-----------------------------------------------------------------------------
#FC = f77
#ABI = -n32 # 2 GB address space
##ABI = -64 # For 64-bit address space
#PREC = # default 32-bit presicion.
##PREC = -r8 # 64-bit precision.
##Conversion program between different precisions of  mminput and bdyout avail-
able from wesley@sgi.com
#MP = -mp -MP:old_mp=OFF
##MP = -mp -MP:open_mp=OFF # Use SGI multiprocessing directives
#OPT = -O3 -OPT:roundoff=3:IEEE_arithmetic=3
##debugging#OPT = -g -DEBUG:div_check:subscript_check=ON:trap_uninitialized=ON
#FCFLAGS = -I$(LIBINCLUDE) $(ABI) $(PREC) $(MP) $(OPT)
#CFLAGS =
#CPP = /usr/lib/cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS = $(ABI) $(PREC) $(MP) $(OPT)
#LOCAL_LIBRARIES = -lfastm
#MAKE = make -i -r -P
```

```
#-----------------------------------------------------------------------------
#    3c. IRIX.6.X (SGI_R4400/SGI_R4000/SGI_R5000)
#-----------------------------------------------------------------------------
#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)
#-----------------------------------------------------------------------------
#FC = f77
#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)
#-----------------------------------------------------------------------------
#FC = f77
#FCFLAGS=-cpp -D$(RUNTIME_SYSTEM) -I$(LIBINCLUDE) -c -O4 -Olimit 2000 \
#-automatic -fpe0 -align dcommons -align records -warn nounreachable \
#-convert big_endian
#CFLAGS =
#CPP = cpp
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
#LDOPTIONS = -math_library accurate
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#-----------------------------------------------------------------------------
#   3e2. DEC_ALPHA (4100/8400 Series)
#        Use OpenMP directives for multi-processor runs.
#        - set RUNTIME_SYSTEM = DEC_ALPHA
#-----------------------------------------------------------------------------
#FC = f90
#FCFLAGS = -omp -cpp -D$(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)
#        for xlf compiler, use `make little_f' instead of `make' to compile
#-----------------------------------------------------------------------------
#FC = xlf
#FCFLAGS = -I$(LIBINCLUDE) -O -qmaxmem=-1
##FCFLAGS = -I$(LIBINCLUDE) -qmaxmem=-1
#CPP = /usr/lib/cpp
#CFLAGS =
#CPPFLAGS = -I$(LIBINCLUDE) -C -P
##LDOPTIONS = -qmaxmem=-1 -O
#LDOPTIONS = -qmaxmem=-1
#LOCAL_LIBRARIES =
#MAKE = make -i
#-----------------------------------------------------------------------------
#    3g. HP (UX)
```

```
#-----------------------------------------------------------------------------
#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
#-----------------------------------------------------------------------------
#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 +Onodynsel
#
## 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
#-----------------------------------------------------------------------------
# 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
#-----------------------------------------------------------------------------
#
# NHYDRO (integer)                  - "1" -> NonHydrostatic run
#                                   - "0" -> Hydrostatic run
NHYDRO = 1
# FDDAGD (integer)                  - "1" -> FDDA gridded run
#                                   - "0" -> NonFDDA run
FDDAGD = 0
# FDDAOBS (integer)                 - "1" -> FDDA obs run
#                                   - "0" -> NonFDDA 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.
#-----------------------------------------------------------------------------
# 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(gsfc),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)
#
# ICUPA - for cumulus schemes (array,integer)
#                                 - None,Kuo,Grell,AS,FC,KF,BM - 1,2,3,4,5,6,7
ICUPA  = "3,3,1,1,1,1,1,1,1,1"
#
# IBLTYP - for planetary boundary layer (array,integer)
#                                   - PBL type 0=no PBL fluxes,1=bulk,
#                                  2=Blackadar,3=Burk-Thompson,4=Eta M-Y,5=MRF
IBLTYP  = "5,5,2,2,2,2,2,2,2,1"
#
# FRAD - for atmospheric radiation (integer)
#                                   - Radiation cooling of atmosphere
#                                     0=none,1=simple,2=cloud,3=ccm2
FRAD = "2,0,0,0,0"
#
# ISOIL - for multi-layer soil temperature model (integer)
#                                  - 0=no,1=yes (only works with IBLTYP=2,4,5)
ISOIL = 1
#
# ISHALLO (array,integer)           - Shallow Convection Option
#                                 1=shallow convection,0=No shallow convection
ISHALLO  = "0,0,0,0,0,0,0,0,0,0,0"
#-----------------------------------------------------------------------------
# 7. MPP options
#-----------------------------------------------------------------------------
# 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_HOMOGENOUS_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
#------------------------------------------------------------------------------
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = xlf
#MCC = mpcc
#MLD = mpxlf
#FCFLAGS = -O3 -qstrict -qarch=pwrx -qhsflt
#LDOPTIONS =
##LOCAL_LIBRARIES = -lmass
#LOCAL_LIBRARIES = -lessl
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -DMPI -Drs6000
#CFLAGS = -DNOUNDERSCORE -DMPI
#ARCH_OBJS =  milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#------------------------------------------------------------------------------
#    7a.1 IBM SP2 with Silver nodes
#         type `make mpp' for the SP2
#       - You must compile with XLF or MPXLF version 6.1 or greater.
#       - Check with your system admin before linking to lessl or lmass.
#       - Expect unusually long compile time for solve3 (~ 1hr). (3/99)
#       - Note for running on blue.llnl.gov:
#          newmpxlf_r is LLNL specific wrapper around HPF 6.1 w/ HPF off.
#------------------------------------------------------------------------------
#MPP_TARGET=$(RUNTIME_SYSTEM)
## On llnl.blue.gov, (3/99)
##MFC = time newmpxlf_r
##MCC = mpcc_r
##MLD = newmpxlf_r
## On systems with R6.1 or greater of IBM Fortran.
#MFC = time mpxlf
#MCC = mpcc
#MLD = mpxlf
#FCFLAGS = -O3 -qarch=auto -qzerosize -qsmp=noauto,schedule=static -qnosave -
qmaxmem=-1
#LDOPTIONS =  -qsmp=noauto,schedule=static
##LOCAL_LIBRARIES = -lmass
##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
#CFLAGS = -DNOUNDERSCORE -DMPI
#ARCH_OBJS =  milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#------------------------------------------------------------------------------
```

```
#     7b. T3E
#-----------------------------------------------------------------------------
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = f90
#MCC = cc
#MLD = $(MFC)
##FCFLAGS = -g
#FCFLAGS = -O2
#LDOPTIONS =
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /opt/ctl/bin/cpp -C -P
#CPPFLAGS = -DMPI -DT3E
#CFLAGS = -DNOUNDERSCORE -Dt3e -DT3E -DMPI
#ARCH_OBJS = error_dupt3d.o t3etraps.o set_to_nan.o milliclock.o
#IWORDSIZE = 8
#RWORDSIZE = 8
#LWORDSIZE = 8
#ASSUME_HOMOGENEOUS_ENVIRONMENT = 1
#-----------------------------------------------------------------------------
#     7c. Origin 2000
#-----------------------------------------------------------------------------
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = f77 -n32 -mips4 -w
#MCC = cc -n32 -mips4 -w
#MLD = f77 -n32 -mips4
##FCFLAGS = -g
#FCFLAGS = -O3 -OPT:roundoff=3:IEEE_arithmetic=3 -OPT:fold_arith_limit=2001
#LDOPTIONS =
#LOCAL_LIBRARIES = -lfastm /usr/lib32/libmpi.so
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -DMPI -DO2K
#CFLAGS = -DO2K -DMPI
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#ASSUME_HOMOGENEOUS_ENVIRONMENT = 1
#-----------------------------------------------------------------------------
#     7d. HP Exemplar
#-----------------------------------------------------------------------------
#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
#CFLAGS = -DNOUNDERSCORE -DMPI
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#ASSUME_HOMOGENEOUS_ENVIRONMENT = 1
#-----------------------------------------------------------------------------
#    7e. DEC 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.
#-----------------------------------------------------------------------------
#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 = $(OMP)
###### If using DEC MPI (e.g. on fir.mmm.ucar.edu) ###
##MFC = f90
##MCC = cc
##MLD = f90
##LDOPTIONS = -lmpi $(SPECIAL_OMP)
######
#FCFLAGS = -O4 -Olimit 2000 -fpe0 -align dcommons -align records \
#          -convert big_endian $(SPECIAL_OMP)
#LOCAL_LIBRARIES =
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = cpp -C -P
#CPPFLAGS = -DMPI -DDEC_ALPHA
#CFLAGS = -DMPI -DDEC_ALPHA
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#ASSUME_HOMOGENEOUS_ENVIRONMENT = 1
#-----------------------------------------------------------------------------
#    7f. Fujitsu VPP
#-----------------------------------------------------------------------------
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = frt
#MCC = cc
#MLD = frt
##FCFLAGS =  -Sw -Wl,-P -lmpi -lmp -Of
#FCFLAGS =  -Sw -Wv,-Of,-te,-ilfunc,-noalias,-m3,-P255 -Oe,-P -Kfast -Pdos -
lmpi -lmp
#LDOPTIONS = -Wl,-P -L$(MPILIBS) -lmpi -J -lmp
#LOCAL_LIBRARIES =
#MAKE = make -i -r
```

```
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = $(CAT)
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -DMPI -Dvpp -I$(MPIINCDIR)
#CFLAGS = -DMPI -Dvpp -I$(MPIINCDIR)
#ARCH_OBJS = milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#ASSUME_HOMOGENEOUS_ENVIRONMENT = 1
#FLIC_MACROS = LMvpp.m4
#-----------------------------------------------------------------------------
#    7g. Linux PCs.  Need Portland Group pgf77 and MPICH.
#                    You may need to edit value for LOCAL_LIBRARIES for MPICH
#                    on your machine.
#                    Set RUNTIME_SYSTEM to linux, (Section 2, above)
#         type `make mpp'
#-----------------------------------------------------------------------------
#MPP_TARGET=$(RUNTIME_SYSTEM)
#MFC = pgf77
#MCC = gcc
#MLD = pgf77
#FCFLAGS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio
#LDOPTIONS = -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswapio
#LOCAL_LIBRARIES = -L/usr/local/mpi/lib/LINUX/ch_p4 -lfmpi -lmpi
#MAKE = make -i -r
#AWK = awk
#SED = sed
#CAT = cat
#CUT = cut
#EXPAND = expand
#M4 = m4
#CPP = /lib/cpp -C -P
#CPPFLAGS = -DMPI -Dlinux
#CFLAGS = -DMPI -I/usr/local/mpi/include
#ARCH_OBJS =  milliclock.o
#IWORDSIZE = 4
#RWORDSIZE = 4
#LWORDSIZE = 4
#ASSUME_HOMOGENEOUS_ENVIRONMENT = 1
#-----------------------------------------------------------------------------
# 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
```

9.15 Configure.user for PC

```
# Sections
# This configure.user.linux file is used to compile on PC running linux only.
#    For options to compile on Unix systems, please use configure.user
# 1. System Variables
# 2. User Variables
# 3. Fortran options
#    3i. PC_PGF77 (Linux)
# 4. General commands
# 5. Options for making "./include/parame.incl"
# 6. Physics Options (memory related)
#
#-------------------------------------------------------------------------------
# 1. System Variables
#-------------------------------------------------------------------------------
SHELL = /bin/sh
.SUFFIXES: .F .i .o .f
#-------------------------------------------------------------------------------
# 2. User Variables
#-------------------------------------------------------------------------------
# RUNTIME_SYSTEM- Currently supported systems.
#     PC_PGF77
RUNTIME_SYSTEM = "PC_PGF77"
#
#-------------------------------------------------------------------------------
# 3. Fortran options
#-------------------------------------------------------------------------------
LIBINCLUDE = $(DEVTOP)/include
#-------------------------------------------------------------------------------
#    3i. PC running Linux and using pgf77 compiler
#-------------------------------------------------------------------------------
#    3i1. PC_PGF77 (LINUX/Portland Group Inc.)
#         pgf77 version 1.6 and above
#-------------------------------------------------------------------------------
 FC = pgf77
 FCFLAGS = -I$(LIBINCLUDE) -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswa-
pio
#FCFLAGS = -I$(LIBINCLUDE) -O2 -Mcray=pointer -tp p6 -pc 32 -Mnoframe -byteswa-
pio -mp
 CPP = /lib/cpp
 CFLAGS = -O
 CPPFLAGS = -I$(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
#-------------------------------------------------------------------------------
# 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
#-------------------------------------------------------------------------------
#
# NHYDRO (integer)                    - "1" -> NonHydrostatic run
```

```
#                                    - "0" -> Hydrostatic run
NHYDRO = 1
# FDDAGD (integer)                   - "1" -> FDDA gridded run
#                                    - "0" -> NonFDDA run
FDDAGD = 0
# FDDAOBS (integer)                  - "1" -> FDDA obs run
#                                    - "0" -> NonFDDA run
FDDAOBS = 0
# MAXNES (integer)                   - Max Number of Domains in simulation
MAXNES = 1
# MIX,MJX,MKX (integer)              - Maximum Dimensions of any Domain
MIX = 49
MJX = 52
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.
#-----------------------------------------------------------------------------
# IMPHYS - for explicit moisture schemes (array,integer)
IMPHYS = "4,4,1,1,1,1,1,1,1,1"
#                                    - Dry,stable precip.,warm rain,simple ice
#                                    - 1  ,2            ,3        ,4
#                                   - mix phase,graupel(gsfc),graupel(reisner2)
#                                    - 5        ,6            ,7
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)
#
# ICUPA - for cumulus schemes (array,integer)
#                                  - None,Kuo,Grell,AS,FC,KF,BM - 1,2,3,4,5,6,7
ICUPA  = "3,3,1,1,1,1,1,1,1,1"
#
# IBLTYP - for planetary boundary layer (array,integer)
#                                    - PBL type 0=no PBL fluxes,1=bulk,
#                                    2=Blackadar,3=Burk-Thompson,5=MRF(ISOIL=1)
IBLTYP  = "5,5,1,1,1,1,1,1,1,1"
#
# FRAD - for atmospheric radiation (integer)
#                                    - Radiation cooling of atmosphere
#                                      0=none,1=simple,2=cloud,3=ccm2
FRAD = "2,0,0,0,0"
#
# ISOIL - for multi-layer soil temperature model (integer)
#                                    - 0=no,1=yes (only works with IBLTYP=2,5)
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"
#-----------------------------------------------------------------------------
# Don't touch anything below this line
#-----------------------------------------------------------------------------
.c.o:
    $(RM) $@ && \
    $(CC) -c $(CFLAGS) $*.c
.F.o:
    $(RM) $@
```

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

9.16 mm5.deck

This is a Bourne shell script. Slight variations exist for different machines.

```
#!/bin/sh
#
#  Version 2 of mm5 job deck
#
# temp files should be accessible
umask 022

# Sections
# 1. Options for namelist ("mmlif")
# 2. Options for I/O
# 3. Running...
#
#-----------------------------------------------------------------------------
# 1. Options for namelist ("mmlif")
#-----------------------------------------------------------------------------
#
#           The first dimension (column) of the arrays denotes the domain
#           identifier.
#           Col 1 = Domain #1, Col 2 = Dom #2, etc.
#
cat > ./Run/oparam << EOF
 &OPARAM
 ;
 ;      ************** OUTPUT/RESTART OPTIONS ***************
 ;
 IFREST = .FALSE.,          ; whether this is a restart
    IXTIMR  = 720,          ; restart time in minutes
 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
 IFSAVE = .TRUE.,           ; save data for restart
    SVLAST = .FALSE.,       ; T: only save the last file for restart
                            ; F: save multiple files
    SAVFRQ =360.,           ; how frequently to save data (in minutes)
 IFTAPE = 1,                ; model output: 0,1
    TAPFRQ =60.,         ; how frequently to output model results (in minutes)
     INCTAP =1,1,1,1,1,1,1,1,1,1, ; multipliers of TAPFRQ for outputting
 IFSKIP = .FALSE.,       ; whether to skip input files - DO NOT use this for restart
    MDATEST = 00000000,     ; the MDATE for the start file
 IFPRT = 0,                 ; sample print out: =1, a lot of print
 PRTFRQ = 720.,             ; Print frequency for sample output (in minutes)
 MASCHK = 60,            ; mass conservation check (KTAU or no. of time steps)
 &END
EOF
cat > ./Run/lparam << EOF
 &LPARAM
 ;
 ;       1. user-chosen options I
 ;
 RADFRQ     = 30.,   ;atmospheric radiation calculation frequency (in minutes)
 IMVDIF     = 1,     ;moist vertical diffusion in clouds - 0, 1 (IBLTYP=2,5 only)
 IVQADV     = 1,      ;vertical moisture advection uses log interpolation - 0,
                      ;linear - 1
 IVTADV     = 1,     ;vertical temperature advection uses theta interpolation - 0,
                      ;linear - 1
```

```
ITHADV    = 1,      ;advection of temperature uses potential temperature - 1,
                    ;standard - 0
ITPDIF    = 1,    ;diffusion using perturbation temperature - 0,1 (NH run only)
ICOR3D    = 1,      ;3D Coriolis force (for NH run only) - 0, 1
IFUPR     = 1,      ;upper radiative boundary condition (NH) - 0, 1
;
;       2. keep the following two variables as they are
;          unless doing sensitivity runs
;
IFDRY     = 0,      ;fake-dry run (no latent heating) - 0, 1
;                  for IMPHYS = 2,3,4,5,6,7 (requires ICUPA = 1)
ICUSTB    = 1,      ;stability check for Anthes-Kuo CPS only - 0, 1
;
;       3. user-chosen options II
;
IBOUDY = 3, 2, 2, 2, 2, 2, 2, 2, 2, 2, ;boundary conditions
;                   (fixed, relaxation, time-dependent,
;                   time and inflow/outflow dependent (HY)/relaxation (NH),
;                   SPONGE (HY only) - 0, 1, 2, 3, 4)
IFSNOW = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ;SNOW COVER EFFECTS - 0, 1
;                   (only if snow data are generated in DATAGRID)
;
;       4. keep the following 9 variables as they are
;          unless doing sensitivity runs
;
ISFFLX = 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, ;surface temperature prediction - 1, 3
ISFPAR = 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, ;cloud effects on radiation - 0, 1
;                                       currently for IFRAD = 1,2
ICDCON = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ;constant drag coefficients - 0, 1
;                                       (IBLTYP=1 only)
IVMIXM = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ;vertical mixing of momentum - 0, 1
;                                       (IBLTYP=2 only)
HYDPRE = 1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,;HYDRO EFFECTS OF LIQ WATER - 0., 1.
;                                       (HY run only)
IEVAP  = 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ;EVAP OF CLOUD/RAINWATER - <0, 0, >0
;                                       (currently for IMPHYS=3,4,5 only)
;
;
;      ************* NESTING OPTIONS ****************
;
NESTIX =  25,  34,  31,  46,  46,  46,  46,  46,  46,  46,  ; domain size i
NESTJX =  28,  37,  31,  61,  61,  61,  61,  61,  61,  61,  ; domain size j
NESTI  =   1,   8,   8,   1,   1,   1,   1,   1,   1,   1,   ; start location i
NESTJ  =   1,   9,   9,   1,   1,   1,   1,   1,   1,   1,   ; start location i
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,    1,    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=interpolate from coarse mesh, but read in high resolution
 ;         terrain and landuse
IACTIV =   1,    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, ; move domain 0,1
IMOVCO =   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
EOF
cat > ./Run/pparam << EOF
 &PPARAM
 ;
 ;      ************* FORECAST TIME AND TIME STEP ******************
 ;
 TIMAX  = 720.,         ; forecast length in minutes
 TISTEP = 270.,         ; coarse domain DT in model, use 3*DX
 ;
 ;      ************* 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%)
 IFEED  = 3,            ;no feedback, 9-pt weighted average, 1-pt feedback
                        ;with no smoothing / light smoothing / heavy smoothing
                        ; - 0, 1,2,3, and 4
 IABSOR = 0,            ; sponge upper boundary (HYD only) - 0,1
 &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.
 ;       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.,
 ;
 ;      **************** 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
;
IMOIS=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 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 gradu-
ally
;            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
#
make mmlif
cd ./Run
sed -f ../Util/no_comment.sed mmlif | grep [A-Z,a-z] > mmlif.tmp
mv mmlif.tmp mmlif
#
#-----------------------------------------------------------------------------
# 2. Options for I/O
#    Names of input/output files in ./Run directory:
#
#    Input files:
#    1) hydrostatic:    bdyout_hy_domain1
#                       mminput_hy_domain1 [mminput_hy_domain2, ....]
#    2) nonhydrostatic: bdyout_nh_domain1
#                       mminput_nh_domain1 [mminput_nh_domain2, ....]
#    3) restart files:  save_domain1 [save_domain2, ....]
#    4) fdda sfc files: rw4dda_domain1 [rw4dda_domain2, ....]
#    5) fdda obs files: mm5obs_domain1 [mm5obs_domain2, ....]
#
#    Output files
#    6) model output:           fort.41 [fort.42, ....]
#    7) restart files:          fort.51 [fort.52, ....]
#    8) shutdown/restart files: fort.61 [fort.62, ....]
#-----------------------------------------------------------------------------
#
#
STARTsw=NoReStart                # start model run at hour 0
#STARTsw=ReStart                 # restart model run
echo "STARTsw   = $STARTsw"
#
FDDAsw=NoFDDA                    # no FDDA input files
#FDDAsw=Anly                     # gridded FDDA input files
#FDDAsw=Obs                      # obs FDDA input files
#FDDAsw=Both                     # gridded and obs FDDA input files
echo "FDDAsw    = $FDDAsw"
#
#HYDROsw=Hydro                   # hydrostatic input files
HYDROsw=NonHydro                 # nonhydrostatic input files
echo "HYDROsw   = $HYDROsw"
#
#
#      boundary condition
#      initial conditions
#
if [ $HYDROsw = Hydro ]; then
      InBdy=bdyout_hy_domain1
      InMM="mminput_hy_domain1 mminput_hy_domain2"
else
      if [ $HYDROsw = NonHydro ]; then
            InBdy=bdyout_nh_domain1
            InMM="mminput_nh_domain1 mminput_nh_domain2"
      fi
fi
echo "InBdy     = $InBdy"
echo "InMM      = $InMM"
#
#
#      the input restart file
#
if [ $STARTsw = ReStart ]; then
            InRst="save_domain1 save_domain2"
            echo "InRst   = $InRst"
fi
```

```
#
#       4dda surface analyses
#
if [ $FDDAsw = Anly ]; then
            In4DSfc="rw4dda_domain1 rw4dda_domain2"
else
  if [ $FDDAsw = Both ]; then
            In4DSfc="rw4dda_domain1 rw4dda_domain2"
  fi
fi
echo "In4DSfc  = $In4DSfc"
#
#       4dda observations
#
if [ $FDDAsw = Obs ]; then
            In4DObs="mm5obs_domain1 mm5obs_domain2"
else
  if [ $FDDAsw = Both ]; then
            In4DObs="mm5obs_domain1 mm5obs_domain2"
  fi
fi
echo "In4DObs  = $In4DObs"
#
ForUnit=fort.
#
#-----------------------------------------------------------------------------
# 3. Running...
#    This section should not have to be modified by the user!!
#-----------------------------------------------------------------------------
#
rm -rf ${ForUnit}*
rm fparam lparam oparam pparam
#
# namelist file
ln -s mmlif ${ForUnit}10
# transmisivity file "ehtran"
ln -s ehtran ${ForUnit}8
ln -s landuse.tbl ${ForUnit}19
# boundary file
ln -s $InBdy ${ForUnit}9
#
#
#        input initial conditions
#
  NUMd=0
  for i in $InMM
  do
        NUMd=`expr $NUMd + 1`
          MMd=$i
          echo "InMM[$NUMd] = $MMd"
          if [ ! -r ${ForUnit}1$NUMd ]; then
            ln -s $MMd ${ForUnit}1$NUMd
          fi
          echo "ln -s $MMd ${ForUnit}1$NUMd"
          if [ $FDDAsw = Anly -o $FDDAsw = Both ]; then
            cp ${ForUnit}1$NUMd ${ForUnit}3$NUMd
            echo "cp ${ForUnit}1$NUMd ${ForUnit}3$NUMd"
          fi
  done
#
#        input restart conditions
#
if [ $STARTsw = ReStart ]; then
  NUMd=0
  for i in $InRst
  do
```

```
            NUMd=`expr $NUMd + 1`
              Rst=$i
              echo "InRst[$NUMd] = $Rst"
            if [ ! -r ${ForUnit}9$NUMd ]; then
                    ln -s $Rst ${ForUnit}9$NUMd
                echo "ln -s $Rst ${ForUnit}9$NUMd"
            fi
  done
fi
#
#       get analyses for nudging
#
if [ $FDDAsw = Anly -o $FDDAsw = Both ]; then
  for i in $In4DSfc
  do
       NUMd=`echo $i | grep [0-9]$ | sed 's/.*\(.\)//'`
       if [ "$NUMd" != "" ]; then
          echo "Current domain is $NUMd"
          if [ ! -r ${ForUnit}7$NUMd ]; then
             ln -s $i ${ForUnit}7$NUMd
             echo "ln -s $i ${ForUnit}7$NUMd"
          fi
       fi
       cp ${ForUnit}7$NUMd ${ForUnit}8$NUMd
       echo "cp ${ForUnit}7$NUMd ${ForUnit}8$NUMd"
  done
fi
#
#       observations if OBS nudging
#
if [ $FDDAsw = Obs -o $FDDAsw = Both ]; then
  for i in $In4DObs
  do
       NUMd=`echo $i | grep [0-9]$ | sed 's/.*\(.\)//'`
       if [ "$NUMd" != "" ]; then
          echo "Current domain is $NUMd"
          if [ ! -r ${ForUnit}6$NUMd ]; then
             ln -s $i ${ForUnit}6$NUMd
             echo "ln -s $i ${ForUnit}6$NUMd"
          fi
       fi
  done
fi
#
#       run MM5
#
date
echo "time mm5.exe >! mm5.print.out "
time mm5.exe > mm5.print.out 2>&1
```

 # List of MM5 Fortran files