Supplementary Material 5

```
# ===================== CCTMv5.3.X Run Script ========================
# Usage: run.cctm >&! cctm_Bench_2016_12SE1.log &
#
# To report problems or request help with this script/program:
#           http://www.epa.gov/cmaq (EPA CMAQ Website)
#           http://www.cmascenter.org (CMAS Website)
# ====================================================================

# ====================================================================
#> Runtime Environment Options
# ====================================================================

echo 'Start Model Run At ' `date`

#> Toggle Diagnostic Mode which will print verbose information to
#> standard output
setenv CTM_DIAG_LVL 0

#> Choose compiler and set up CMAQ environment with correct
#> libraries using config.cmaq. Options: intel | gcc | pgi
if ( ! $?compiler ) then
  setenv compiler gcc
endif
if ( ! $?compilerVrsn ) then
  setenv compilerVrsn Empty
endif

#> Source the config.cmaq file to set the build environment
cd ../..
```

```
source ./config_cmaq.csh $compiler
cd CCTM/scripts

#> Set General Parameters for Configuring the Simulation
set VRSN      = v532            #> Code Version
set PROC      = mpi             #> serial or mpi
set MECH      = cb6r3_ae7_aq    #> Mechanism ID
set APPL      = BR_2019  #> Application Name (e.g. Gridname)

#> Define RUNID as any combination of parameters above or others. By default,
#> this information will be collected into this one string, $RUNID, for easy
#> referencing in output binaries and log files as well as in other scripts.
setenv RUNID  ${VRSN}_${compilerString}_${APPL}

#> Set the build directory (this is where the CMAQ executable
#> is located by default).
set BLD       = ${CMAQ_HOME}/CCTM/scripts/BLD_CCTM_${VRSN}_${compilerString}
set EXEC      = CCTM_${VRSN}.exe

#> Output Each line of Runscript to Log File
if ( $CTM_DIAG_LVL != 0 ) set echo

#> Set Working, Input, and Output Directories
setenv WORKDIR ${CMAQ_HOME}/CCTM/scripts         #> Working Directory. Where the runscript is.
setenv OUTDIR  ${CMAQ_DATA}/output_CCTM_${RUNID}  #> Output Directory
setenv INPDIR  ${CMAQ_DATA}/inputs/BR_2019          #> Input Directory
setenv LOGDIR  ${OUTDIR}/LOGS     #> Log Directory Location
setenv NMLpath ${BLD}           #> Location of Namelists. Common places are:
                        #>   ${WORKDIR} | ${CCTM_SRC}/MECHS/${MECH} | ${BLD}

echo ""
```

```
echo "Working Directory is $WORKDIR"
echo "Build Directory is $BLD"
echo "Output Directory is $OUTDIR"
echo "Log Directory is $LOGDIR"
echo "Executable Name is $EXEC"

# ==========================================================================
#> CCTM Configuration Options
# ==========================================================================

#> Set Start and End Days for looping
setenv NEW_START FALSE           #> Set to FALSE for model restart
set START_DATE = "2019-01-21"    #> beginning date (July 1, 2016)
set END_DATE   = "2019-01-22"    #> ending date    (July 1, 2016)

#> Set Timestepping Parameters
set STTIME     = 000000          #> beginning GMT time (HHMMSS)
set NSTEPS     = 240000          #> time duration (HHMMSS) for this run
set TSTEP      = 010000          #> output time step interval (HHMMSS)

#> Horizontal domain decomposition
if ( $PROC == serial ) then
  setenv NPCOL_NPROW "1 1"; set NPROCS   = 1 # single processor setting
else
  @ NPCOL  =  7; @ NPROW =  5
  @ NPROCS = $NPCOL * $NPROW
  setenv NPCOL_NPROW "$NPCOL $NPROW";
endif

#   @ NPROCS =
```

```
#> Define Execution ID: e.g. [CMAQ-Version-Info]_[User]_[Date]_[Time]
setenv EXECUTION_ID "CMAQ_CCTM${VRSN}_`id -u -n`_`date -u +%Y%m%d_%H%M%S_%N`"    #> Inform IO/API of the Execution ID
echo ""
echo "---CMAQ EXECUTION ID: $EXECUTION_ID ---"

#> Keep or Delete Existing Output Files
set CLOBBER_DATA = TRUE

#> Logfile Options
#> Master Log File Name; uncomment to write standard output to a log, otherwise write to screen
#setenv LOGFILE $CMAQ_HOME/$RUNID.log
if (! -e $LOGDIR ) then
  mkdir -p $LOGDIR
endif
setenv PRINT_PROC_TIME Y          #> Print timing for all science subprocesses to Logfile
                          #>   [ default: TRUE or Y ]
setenv STDOUT T                   #> Override I/O-API trying to write information to both the processor
                          #>   logs and STDOUT [ options: T | F ]

setenv GRID_NAME BR_2019        #> check GRIDDESC file for GRID_NAME options
setenv GRIDDESC ${CMAQ_HOME}/data/mcip/${APPL}/GRIDDESC    #> grid description file

#> Retrieve the number of columns, rows, and layers in this simulation
set NZ = 35
set NX = `grep -A 1 ${GRID_NAME} ${GRIDDESC} | tail -1 | sed 's/  */ /g' | cut -d' ' -f6`
set NY = `grep -A 1 ${GRID_NAME} ${GRIDDESC} | tail -1 | sed 's/  */ /g' | cut -d' ' -f7`
set NCELLS = `echo "${NX} * ${NY} * ${NZ}" | bc -l`

#> Output Species and Layer Options
   #> CONC file species; comment or set to 'ALL' to write all species to CONC
```

```
  setenv CONC_SPCS "ALL"
  setenv CONC_BLEV_ELEV " 1 1" #> CONC file layer range; comment to write all layers to CONC

  #> ACONC file species; comment or set to 'ALL' to write all species to ACONC
  setenv AVG_CONC_SPCS "O3 NO CO NO2 ASO4I ASO4J NH3 SO2"
  setenv ACONC_BLEV_ELEV " 1 1" #> ACONC file layer range; comment to write all layers to ACONC
  setenv AVG_FILE_ENDTIME N     #> override default beginning ACONC timestamp [ default: N ]

#> Synchronization Time Step and Tolerance Options
setenv CTM_MAXSYNC 300       #> max sync time step (sec) [ default: 720 ]
setenv CTM_MINSYNC  60       #> min sync time step (sec) [ default: 60 ]
setenv SIGMA_SYNC_TOP 0.7    #> top sigma level thru which sync step determined [ default: 0.7 ]
#setenv ADV_HDIV_LIM 0.95    #> maximum horiz. div. limit for adv step adjust [ default: 0.9 ]
setenv CTM_ADV_CFL 0.95      #> max CFL [ default: 0.75]
#setenv RB_ATOL 1.0E-09      #> global ROS3 solver absolute tolerance [ default: 1.0E-07 ]

#> Science Options
setenv CTM_OCEAN_CHEM Y      #> Flag for ocean halgoen chemistry and sea spray aerosol
emissions [ default: Y ]
setenv CTM_WB_DUST N         #> use inline windblown dust emissions [ default: Y ]
setenv CTM_WBDUST_BELD BELD3 #> landuse database for identifying dust source regions
                     #>    [ default: UNKNOWN ]; ignore if CTM_WB_DUST = N
setenv CTM_LTNG_NO N         #> turn on lightning NOx [ default: N ]
setenv KZMIN Y               #> use Min Kz option in edyintb [ default: Y ],
                     #>    otherwise revert to Kz0UT
setenv CTM_MOSAIC N          #> landuse specific deposition velocities [ default: N ]
setenv CTM_FST N             #> mosaic method to get land-use specific stomatal flux
                     #>    [ default: N ]
setenv PX_VERSION Y          #> WRF PX LSM
setenv CLM_VERSION N         #> WRF CLM LSM
setenv NOAH_VERSION N        #> WRF NOAH LSM
```

```
setenv CTM_ABFLUX N         #> ammonia bi-directional flux for in-line deposition
                    #>    velocities [ default: N ]
setenv CTM_BIDI_FERT_NH3 N   #> subtract fertilizer NH3 from emissions because it will be handled
                    #>    by the BiDi calculation [ default: Y ]
setenv CTM_HGBIDI N         #> mercury bi-directional flux for in-line deposition
                    #>    velocities [ default: N ]
setenv CTM_SFC_HONO Y       #> surface HONO interaction [ default: Y ]
setenv CTM_GRAV_SETL Y      #> vdiff aerosol gravitational sedimentation [ default: Y ]
setenv CTM_BIOGEMIS N       #> calculate in-line biogenic emissions [ default: N ]

#> Vertical Extraction Options
setenv VERTEXT N
setenv VERTEXT_COORD_PATH ${WORKDIR}/lonlat.csv

#> I/O Controls
setenv IOAPI_LOG_WRITE F     #> turn on excess WRITE3 logging [ options: T | F ]
setenv FL_ERR_STOP N        #> stop on inconsistent input files
setenv PROMPTFLAG F         #> turn on I/O-API PROMPT*FILE interactive mode [ options: T | F ]
setenv IOAPI_OFFSET_64 YES   #> support large timestep records (>2GB/timestep record) [ options:
YES | NO ]
setenv IOAPI_CHECK_HEADERS N #> check file headers [ options: Y | N ]
setenv CTM_EMISCHK N        #> Abort CMAQ if missing surrogates from emissions Input files
setenv EMISDIAG N           #> Print Emission Rates at the output time step after they have been
                    #>   scaled and modified by the user Rules [options: F | T or 2D | 3D | 2DSUM ]
                    #>   Individual streams can be modified using the variables:
                    #>      GR_EMIS_DIAG_## | STK_EMIS_DIAG_## | BIOG_EMIS_DIAG
                    #>      MG_EMIS_DIAG    | LTNG_EMIS_DIAG   | DUST_EMIS_DIAG
                    #>      SEASPRAY_EMIS_DIAG
                    #>   Note that these diagnostics are different than other emissions diagnostic
                    #>   output because they occur after scaling.
setenv EMISDIAG_SUM F       #> Print Sum of Emission Rates to Gridded Diagnostic File
```

```
#> Diagnostic Output Flags
setenv CTM_CKSUM N          #> checksum report [ default: Y ]
setenv CLD_DIAG N           #> cloud diagnostic file [ default: N ]

setenv CTM_PHOTDIAG N       #> photolysis diagnostic file [ default: N ]
setenv NLAYS_PHOTDIAG "1"    #> Number of layers for PHOTDIAG2 and PHOTDIAG3 from
                    #>     Layer 1 to NLAYS_PHOTDIAG  [ default: all layers ]
#setenv NWAVE_PHOTDIAG "294 303 310 316 333 381 607"  #> Wavelengths written for variables
                                    #>   in PHOTDIAG2 and PHOTDIAG3
                                    #>   [ default: all wavelengths ]

setenv CTM_PMDIAG N         #> Instantaneous Aerosol Diagnostic File [ default: Y ]
setenv CTM_APMDIAG N        #> Hourly-Average Aerosol Diagnostic File [ default: Y ]
setenv APMDIAG_BLEV_ELEV "1 1"  #> layer range for average pmdiag = NLAYS

setenv CTM_SSEMDIAG N       #> sea-spray emissions diagnostic file [ default: N ]
setenv CTM_DUSTEM_DIAG N     #> windblown dust emissions diagnostic file [ default: N ];
                    #>     Ignore if CTM_WB_DUST = N
setenv CTM_DEPV_FILE N      #> deposition velocities diagnostic file [ default: N ]
setenv VDIFF_DIAG_FILE N     #> vdiff & possibly aero grav. sedimentation diagnostic file [ default: N ]
setenv LTNGDIAG N           #> lightning diagnostic file [ default: N ]
setenv B3GTS_DIAG N         #> BEIS mass emissions diagnostic file [ default: N ]
setenv CTM_WVEL N           #> save derived vertical velocity component to conc
                    #>    file [ default: Y ]

# ====================================================================
#> Input Directories and Filenames
# ====================================================================

set ICpath    = ${CMAQ_HOME}/data/icon                  #> initial conditions input directory
```

```
set BCpath    = ${CMAQ_HOME}/data/bcon                    #> boundary conditions input directory
set EMISpath  = $INPDIR   #> gridded emissions input directory
set EMISpath2 = #$INPDIR/emis/gridded_area/rwc       #> gridded surface residential wood combustion emissions directory
set IN_PTpath = #$INPDIR/emis/inln_point           #> point source emissions input directory
set IN_LTpath = #$INPDIR/lightning                 #> lightning NOx input directory
set METpath   = ${CMAQ_HOME}/data/mcip/${APPL}           #> meteorology input directory
#set JVALpath  = $INPDIR/jproc                     #> offline photolysis rate table directory
set OMIpath   = $BLD                           #> ozone column data for the photolysis model
set LUpath    = #$INPDIR/land                      #> BELD landuse data for windblown dust model
set SZpath    = $INPDIR                       #> surf zone file for in-line seaspray emissions

# ======================================================================
#> Begin Loop Through Simulation Days
# ======================================================================
set rtarray = ""

set TODAYG = ${START_DATE}
set TODAYJ = `date -ud "${START_DATE}" +%Y%j` #> Convert YYYY-MM-DD to YYYYJJJ
set START_DAY = ${TODAYJ}
set STOP_DAY = `date -ud "${END_DATE}" +%Y%j` #> Convert YYYY-MM-DD to YYYYJJJ
set NDAYS = 1

while ($TODAYJ <= $STOP_DAY )  #>Compare dates in terms of YYYYJJJ

  set NDAYS = `echo "${NDAYS} + 1" | bc -l`

  #> Retrieve Calendar day Information
  set YYYYMMDD = `date -ud "${TODAYG}" +%Y%m%d` #> Convert YYYY-MM-DD to YYYYMMDD
  set YYYYMM = `date -ud "${TODAYG}" +%Y%m`     #> Convert YYYY-MM-DD to YYYYMM
  set YYMMDD = `date -ud "${TODAYG}" +%y%m%d`   #> Convert YYYY-MM-DD to YYMMDD
```

```
  set YYYYJJJ = $TODAYJ

  #> Calculate Yesterday's Date
  set YESTERDAY = `date -ud "${TODAYG}-1days" +%Y%m%d` #> Convert YYYY-MM-DD to YYYYJJJ

# ========================================================================
#> Set Output String and Propagate Model Configuration Documentation
# ========================================================================
  echo ""
  echo "Set up input and output files for Day ${TODAYG}."

  #> set output file name extensions
  setenv CTM_APPL ${RUNID}_${YYYYMMDD}

  #> Copy Model Configuration To Output Folder
  if ( ! -d "$OUTDIR" ) mkdir -p $OUTDIR
  cp $BLD/CCTM_${VRSN}.cfg $OUTDIR/CCTM_${CTM_APPL}.cfg

# ========================================================================
#> Input Files (Some are Day-Dependent)
# ========================================================================

  #> Initial conditions
  if ($NEW_START == true || $NEW_START == TRUE ) then
    setenv ICFILE ICON_v532_BR_2019_profile_20190101
    setenv INIT_MEDC_1 notused
    setenv INITIAL_RUN Y #related to restart soil information file
  else
    set ICpath = $OUTDIR
    setenv ICFILE CCTM_CGRID_${RUNID}_${YESTERDAY}.nc
    setenv INIT_MEDC_1 $ICpath/CCTM_MEDIA_CONC_${RUNID}_${YESTERDAY}.nc
```

```
  setenv INITIAL_RUN N
endif

#> Boundary conditions
set BCFILE = BCON_v532_BR_2019_profile_20190101

#> Off-line photolysis rates
#set JVALfile  = JTABLE_${YYYYJJJ}

#> Ozone column data
set OMIfile   = OMI_1979_to_2019.dat

#> Optics file
set OPTfile = PHOT_OPTICS.dat

#> MCIP meteorology files
setenv GRID_BDY_2D $METpath/GRIDBDY2D_${APPL}.nc  # GRID files are static, not day-specific
setenv GRID_CRO_2D $METpath/GRIDCRO2D_${APPL}.nc
setenv GRID_CRO_3D $METpath/GRIDCRO3D_${APPL}.nc
setenv GRID_DOT_2D $METpath/GRIDDOT2D_${APPL}.nc
setenv MET_CRO_2D $METpath/METCRO2D_${APPL}.nc
setenv MET_CRO_3D $METpath/METCRO3D_${APPL}.nc
setenv MET_DOT_3D $METpath/METDOT3D_${APPL}.nc
setenv MET_BDY_3D $METpath/METBDY3D_${APPL}.nc
setenv LUFRAC_CRO $METpath/LUFRAC_CRO_${APPL}.nc

#> Emissions Control File
#>
#> IMPORTANT NOTE
#>
```

```
#> The emissions control file defined below is an integral part of controlling the behavior of the model simulation.

#> Among other things, it controls the mapping of species in the emission files to chemical species in the model and

#> several aspects related to the simulation of organic aerosols.

#> Please carefully review the emissions control file to ensure that it is configured to be consistent with the assumptions

#> made when creating the emission files defined below and the desired representation of organic aerosols.

#> For further information, please see:

#> + AERO7 Release Notes section on 'Required emission updates':

#>   https://github.com/USEPA/CMAQ/blob/master/DOCS/Release_Notes/aero7_overview.md

#> + CMAQ User's Guide section 6.9.3 on 'Emission Compatability':

#> https://github.com/USEPA/CMAQ/blob/master/DOCS/Users_Guide/CMAQ_UG_ch06_model_configuration_options.md#6.9.3_Emission_Compatability

#> + Emission Control (DESID) Documentation in the CMAQ User's Guide:

#> https://github.com/USEPA/CMAQ/blob/master/DOCS/Users_Guide/Appendix/CMAQ_UG_appendix B_emissions_control.md

#>

setenv EMISSCTRL_NML ${BLD}/EmissCtrl_${MECH}.nml


#> Spatial Masks For Emissions Scaling

# setenv CMAQ_MASKS ${EMISpath}/OCEAN #> horizontal grid-dependent surf zone file


#> Gridded Emissions Files

setenv N_EMIS_GR 4


set EMISfile  = BRAVESout

setenv GR_EMIS_001 ${EMISpath}/${EMISfile}

setenv GR_EMIS_LAB_001 GRIDDED_EMIS

setenv GR_EM_SYM_DATE_001 F # To change default behaviour please see Users Guide for EMIS_SYM_DATE
```

```
 set EMISfile  = FINNout
 setenv GR_EMIS_002 ${EMISpath}/${EMISfile}
 setenv GR_EMIS_LAB_002 GR_FIRES
 setenv GR_EM_SYM_DATE_002 F #

 set EMISfile  = MEGANout
 setenv GR_EMIS_003 ${EMISpath}/${EMISfile}
 setenv GR_EMIS_LAB_003 GRIDDED_BIO
 setenv GR_EM_SYM_DATE_003 F #
 setenv GR_EMIS_DIAG_003 F

 set EMISfile  = INDout
 setenv GR_EMIS_004 ${EMISpath}/${EMISfile}
 setenv GR_EMIS_LAB_004 GRIDDED_IND
 setenv GR_EM_SYM_DATE_004 F #
 setenv GR_EMIS_DIAG_004 F

#  set EMISfile  = emis_mole_rwc_${YYYYMMDD}_12US1_cmaq_cb6_2016ff_16j.nc
#  setenv GR_EMIS_002 ${EMISpath2}/${EMISfile}
#  setenv GR_EMIS_LAB_002 GR_RES_FIRES
#  setenv GR_EM_SYM_DATE_002 F # To change default behaviour please see Users Guide for
EMIS_SYM_DATE

 #> In-line point emissions configuration
 setenv N_EMIS_PT  0        #> Number of elevated source groups

 set STKCASEG = 12US1_2016ff_16j          # Stack Group Version Label
 set STKCASEE = 12US1_cmaq_cb6_2016ff_16j  # Stack Emission Version Label

 # Time-Independent Stack Parameters for Inline Point Sources
```

```
 setenv STK_GRPS_001 $IN_PTpath/stack_groups/stack_groups_ptnonipm_${STKCASEG}.nc

 setenv STK_GRPS_002 $IN_PTpath/stack_groups/stack_groups_ptegu_${STKCASEG}.nc

 setenv STK_GRPS_003 $IN_PTpath/stack_groups/stack_groups_othpt_${STKCASEG}.nc

 setenv STK_GRPS_004
$IN_PTpath/stack_groups/stack_groups_ptagfire_${YYYYMMDD}_${STKCASEG}.nc

 setenv STK_GRPS_005
$IN_PTpath/stack_groups/stack_groups_ptfire_${YYYYMMDD}_${STKCASEG}.nc

 setenv STK_GRPS_006
$IN_PTpath/stack_groups/stack_groups_ptfire_othna_${YYYYMMDD}_${STKCASEG}.nc

 setenv STK_GRPS_007 $IN_PTpath/stack_groups/stack_groups_pt_oilgas_${STKCASEG}.nc

 setenv STK_GRPS_008 $IN_PTpath/stack_groups/stack_groups_cmv_c3_${STKCASEG}.nc


 # Emission Rates for Inline Point Sources

 setenv STK_EMIS_001
$IN_PTpath/ptnonipm/inln_mole_ptnonipm_${YYYYMMDD}_${STKCASEE}.nc

 setenv STK_EMIS_002 $IN_PTpath/ptegu/inln_mole_ptegu_${YYYYMMDD}_${STKCASEE}.nc

 setenv STK_EMIS_003 $IN_PTpath/othpt/inln_mole_othpt_${YYYYMMDD}_${STKCASEE}.nc

 setenv STK_EMIS_004 $IN_PTpath/ptagfire/inln_mole_ptagfire_${YYYYMMDD}_${STKCASEE}.nc

 setenv STK_EMIS_005 $IN_PTpath/ptfire/inln_mole_ptfire_${YYYYMMDD}_${STKCASEE}.nc

 setenv STK_EMIS_006
$IN_PTpath/ptfire_othna/inln_mole_ptfire_othna_${YYYYMMDD}_${STKCASEE}.nc

 setenv STK_EMIS_007 $IN_PTpath/pt_oilgas/inln_mole_pt_oilgas_${YYYYMMDD}_${STKCASEE}.nc

 setenv STK_EMIS_008 $IN_PTpath/cmv_c3/inln_mole_cmv_c3_${YYYYMMDD}_${STKCASEE}.nc


 # Label Each Emissions Stream

 setenv STK_EMIS_LAB_001 PT_NONEGU

 setenv STK_EMIS_LAB_002 PT_EGU

 setenv STK_EMIS_LAB_003 PT_OTHER

 setenv STK_EMIS_LAB_004 PT_AGFIRES

 setenv STK_EMIS_LAB_005 PT_FIRES

 setenv STK_EMIS_LAB_006 PT_OTHFIRES

 setenv STK_EMIS_LAB_007 PT_OILGAS

 setenv STK_EMIS_LAB_008 PT_CMV
```

```
 # Stack emissions diagnostic files
 #setenv STK_EMIS_DIAG_001 2DSUM
 #setenv STK_EMIS_DIAG_002 2DSUM
 #setenv STK_EMIS_DIAG_003 2DSUM
 #setenv STK_EMIS_DIAG_004 2DSUM
 #setenv STK_EMIS_DIAG_005 2DSUM

 # Allow CMAQ to Use Point Source files with dates that do not
 # match the internal model date
 # To change default behaviour please see Users Guide for EMIS_SYM_DATE
 setenv STK_EM_SYM_DATE_001 T
 setenv STK_EM_SYM_DATE_002 T
 setenv STK_EM_SYM_DATE_003 T
 setenv STK_EM_SYM_DATE_004 T
 setenv STK_EM_SYM_DATE_005 T
 setenv STK_EM_SYM_DATE_006 T
 setenv STK_EM_SYM_DATE_007 T
 setenv STK_EM_SYM_DATE_008 T

 #> Lightning NOx configuration
 if ( $CTM_LTNG_NO == 'Y' ) then
   setenv LTNGNO "InLine"    #> set LTNGNO to "Inline" to activate in-line calculation

 #> In-line lightning NOx options
   setenv USE_NLDN  Y       #> use hourly NLDN strike file [ default: Y ]
   if ( $USE_NLDN == Y ) then
     setenv NLDN_STRIKES ${IN_LTpath}/NLDN.12US1.${YYYYMMDD}_bench.nc
   endif
   setenv LTNGPARMS_FILE ${IN_LTpath}/LTNG_AllParms_12US1_bench.nc #> lightning parameter
file
```

```
endif

#> In-line biogenic emissions configuration
if ( $CTM_BIOGEMIS == 'Y' ) then
  set IN_BEISpath = ${INPDIR}/land
  setenv GSPRO      $BLD/gspro_biogenics.txt
  setenv B3GRD      $IN_BEISpath/b3grd_bench.nc
  setenv BIOSW_YN   Y     #> use frost date switch [ default: Y ]
  setenv BIOSEASON  $IN_BEISpath/bioseason.cmaq.2016_12US1_full_bench.ncf
                    #> ignore season switch file if BIOSW_YN = N
  setenv SUMMER_YN  Y     #> Use summer normalized emissions? [ default: Y ]
  setenv PX_VERSION Y     #> MCIP is PX version? [ default: N ]
  setenv SOILINP    $OUTDIR/CCTM_SOILOUT_${RUNID}_${YESTERDAY}.nc
                    #> Biogenic NO soil input file; ignore if INITIAL_RUN = Y
endif

#> Windblown dust emissions configuration
if ( $CTM_WB_DUST == 'Y' ) then
  # Input variables for BELD3 Landuse option
  setenv DUST_LU_1 $LUpath/beld3_12US1_459X299_output_a_bench.nc
  setenv DUST_LU_2 $LUpath/beld4_12US1_459X299_output_tot_bench.nc
endif

#> In-line sea spray emissions configuration
setenv OCEAN_1 ${EMISpath}/OCEAN #> horizontal grid-dependent surf zone file

#> Bidirectional ammonia configuration
if ( $CTM_ABFLUX == 'Y' ) then
  setenv E2C_SOIL ${LUpath}/epic_festc1.4_20180516/2016_US1_soil_bench.nc
  setenv E2C_CHEM ${LUpath}/epic_festc1.4_20180516/2016_US1_time${YYYYMMDD}_bench.nc
```

```
    setenv E2C_CHEM_YEST
${LUpath}/epic_festc1.4_20180516/2016_US1_time${YESTERDAY}_bench.nc
    setenv E2C_LU ${LUpath}/beld4_12kmCONUS_2011nlcd_bench.nc
  endif

#> Inline Process Analysis
  setenv CTM_PROCAN N       #> use process analysis [ default: N]
  if ( $?CTM_PROCAN ) then   # $CTM_PROCAN is defined
    if ( $CTM_PROCAN == 'Y' || $CTM_PROCAN == 'T' ) then
#> process analysis global column, row and layer ranges
#       setenv PA_BCOL_ECOL "10 90"  # default: all columns
#       setenv PA_BROW_EROW "10 80"  # default: all rows
#       setenv PA_BLEV_ELEV "1  4"   # default: all levels
        setenv PACM_INFILE ${NMLpath}/pa_${MECH}.ctl
        setenv PACM_REPORT $OUTDIR/"PA_REPORT".${YYYYMMDD}
    endif
  endif

#> Integrated Source Apportionment Method (ISAM) Options
setenv CTM_ISAM N
if ( $?CTM_ISAM ) then
   if ( $CTM_ISAM == 'Y' || $CTM_ISAM == 'T' ) then
      setenv SA_IOLIST ${WORKDIR}/isam_control.txt
      setenv ISAM_BLEV_ELEV " 1 1"
      setenv AISAM_BLEV_ELEV " 1 1"

      #> Set Up ISAM Initial Condition Flags
      if ($NEW_START == true || $NEW_START == TRUE ) then
        setenv ISAM_NEW_START Y
        setenv ISAM_PREVDAY
      else
```

```
      setenv ISAM_NEW_START N
      setenv ISAM_PREVDAY "$OUTDIR/CCTM_SA_CGRID_${RUNID}_${YESTERDAY}.nc"
    endif

    #> Set Up ISAM Output Filenames
    setenv SA_ACONC_1     "$OUTDIR/CCTM_SA_ACONC_${CTM_APPL}.nc -v"
    setenv SA_CONC_1      "$OUTDIR/CCTM_SA_CONC_${CTM_APPL}.nc -v"
    setenv SA_DD_1        "$OUTDIR/CCTM_SA_DRYDEP_${CTM_APPL}.nc -v"
    setenv SA_WD_1        "$OUTDIR/CCTM_SA_WETDEP_${CTM_APPL}.nc -v"
    setenv SA_CGRID_1     "$OUTDIR/CCTM_SA_CGRID_${CTM_APPL}.nc -v"

    #> Set optional ISAM regions files
    #setenv ISAM_REGIONS $INPDIR/GRIDMASK_STATES_12SE1.nc


  endif
endif


#> Sulfur Tracking Model (STM)
setenv STM_SO4TRACK N       #> sulfur tracking [ default: N ]
if ( $?STM_SO4TRACK ) then
  if ( $STM_SO4TRACK == 'Y' || $STM_SO4TRACK == 'T' ) then

    #> option to normalize sulfate tracers [ default: Y ]
    setenv STM_ADJSO4 Y

  endif
endif

#> CMAQ-DDM-3D
```

```
setenv CTM_DDM3D N
set NPMAX    = 1
setenv SEN_INPUT ${WORKDIR}/sensinput.dat

setenv DDM3D_HIGH N     # allow higher-order sensitivity parameters [ T | Y | F | N ] (default is N/F)

if ($NEW_START == true || $NEW_START == TRUE ) then
   setenv DDM3D_RST N   # begins from sensitivities from a restart file [ T | Y | F | N ] (default is Y/T)
   set S_ICpath =
   set S_ICfile =
else
   setenv DDM3D_RST Y
   set S_ICpath = $OUTDIR
   set S_ICfile = CCTM_SENGRID_${RUNID}_${YESTERDAY}.nc
endif

setenv DDM3D_BCS F      # use sensitivity bc file for nested runs [ T | Y | F | N ] (default is N/F)
set S_BCpath =
set S_BCfile =

setenv CTM_NPMAX      $NPMAX
setenv CTM_SENS_1     "$OUTDIR/CCTM_SENGRID_${CTM_APPL}.nc -v"
setenv A_SENS_1       "$OUTDIR/CCTM_ASENS_${CTM_APPL}.nc -v"
setenv CTM_SWETDEP_1   "$OUTDIR/CCTM_SENWDEP_${CTM_APPL}.nc -v"
setenv CTM_SDRYDEP_1   "$OUTDIR/CCTM_SENDDEP_${CTM_APPL}.nc -v"
setenv CTM_NPMAX      $NPMAX
   if ( $?CTM_DDM3D ) then
     if ( $CTM_DDM3D == 'Y' || $CTM_DDM3D == 'T' ) then
setenv INIT_SENS_1     $S_ICpath/$S_ICfile
setenv BNDY_SENS_1     $S_BCpath/$S_BCfile
      endif
```

```
  endif

# ====================================================================

#> Output Files

# ====================================================================


  #> set output file names

  setenv S_CGRID        "$OUTDIR/CCTM_CGRID_${CTM_APPL}.nc"        #> 3D Inst. Concentrations

  setenv CTM_CONC_1     "$OUTDIR/CCTM_CONC_${CTM_APPL}.nc -v"      #> On-Hour
Concentrations

  setenv A_CONC_1       "$OUTDIR/CCTM_ACONC_${CTM_APPL}.nc -v"     #> Hourly Avg.
Concentrations

  setenv MEDIA_CONC     "$OUTDIR/CCTM_MEDIA_CONC_${CTM_APPL}.nc -v"  #> NH3 Conc. in
Media

  setenv CTM_DRY_DEP_1  "$OUTDIR/CCTM_DRYDEP_${CTM_APPL}.nc -v"    #> Hourly Dry
Deposition

  setenv CTM_DEPV_DIAG  "$OUTDIR/CCTM_DEPV_${CTM_APPL}.nc -v"      #> Dry Deposition
Velocities

  setenv B3GTS_S        "$OUTDIR/CCTM_B3GTS_S_${CTM_APPL}.nc -v"   #> Biogenic Emissions

  setenv SOILOUT        "$OUTDIR/CCTM_SOILOUT_${CTM_APPL}.nc"      #> Soil Emissions

  setenv CTM_WET_DEP_1  "$OUTDIR/CCTM_WETDEP1_${CTM_APPL}.nc -v"   #> Wet Dep From All
Clouds

  setenv CTM_WET_DEP_2  "$OUTDIR/CCTM_WETDEP2_${CTM_APPL}.nc -v"   #> Wet Dep From
SubGrid Clouds

  setenv CTM_PMDIAG_1   "$OUTDIR/CCTM_PMDIAG_${CTM_APPL}.nc -v"    #> On-Hour Particle
Diagnostics

  setenv CTM_APMDIAG_1  "$OUTDIR/CCTM_APMDIAG_${CTM_APPL}.nc -v"   #> Hourly Avg.
Particle Diagnostics

  setenv CTM_RJ_1       "$OUTDIR/CCTM_PHOTDIAG1_${CTM_APPL}.nc -v"  #> 2D Surface Summary
from Inline Photolysis

  setenv CTM_RJ_2       "$OUTDIR/CCTM_PHOTDIAG2_${CTM_APPL}.nc -v"  #> 3D Photolysis Rates

  setenv CTM_RJ_3       "$OUTDIR/CCTM_PHOTDIAG3_${CTM_APPL}.nc -v"  #> 3D Optical and
Radiative Results from Photolysis

  setenv CTM_SSEMIS_1   "$OUTDIR/CCTM_SSEMIS_${CTM_APPL}.nc -v"    #> Sea Spray Emissions

  setenv CTM_DUST_EMIS_1 "$OUTDIR/CCTM_DUSTEMIS_${CTM_APPL}.nc -v"  #> Dust Emissions
```

```
 setenv CTM_IPR_1      "$OUTDIR/CCTM_PA_1_${CTM_APPL}.nc -v"       #> Process Analysis
 setenv CTM_IPR_2      "$OUTDIR/CCTM_PA_2_${CTM_APPL}.nc -v"       #> Process Analysis
 setenv CTM_IPR_3      "$OUTDIR/CCTM_PA_3_${CTM_APPL}.nc -v"       #> Process Analysis
 setenv CTM_IRR_1      "$OUTDIR/CCTM_IRR_1_${CTM_APPL}.nc -v"      #> Chem Process Analysis
 setenv CTM_IRR_2      "$OUTDIR/CCTM_IRR_2_${CTM_APPL}.nc -v"      #> Chem Process Analysis
 setenv CTM_IRR_3      "$OUTDIR/CCTM_IRR_3_${CTM_APPL}.nc -v"      #> Chem Process Analysis
 setenv CTM_DRY_DEP_MOS "$OUTDIR/CCTM_DDMOS_${CTM_APPL}.nc -v"      #> Dry Dep
 setenv CTM_DRY_DEP_FST "$OUTDIR/CCTM_DDFST_${CTM_APPL}.nc -v"      #> Dry Dep
 setenv CTM_DEPV_MOS    "$OUTDIR/CCTM_DEPVMOS_${CTM_APPL}.nc -v"    #> Dry Dep Velocity
 setenv CTM_DEPV_FST    "$OUTDIR/CCTM_DEPVFST_${CTM_APPL}.nc -v"    #> Dry Dep Velocity
 setenv CTM_VDIFF_DIAG  "$OUTDIR/CCTM_VDIFF_DIAG_${CTM_APPL}.nc -v"  #> Vertical
Dispersion Diagnostic
 setenv CTM_VSED_DIAG   "$OUTDIR/CCTM_VSED_DIAG_${CTM_APPL}.nc -v"   #> Particle Grav.
Settling Velocity
 setenv CTM_LTNGDIAG_1  "$OUTDIR/CCTM_LTNGHRLY_${CTM_APPL}.nc -v"    #> Hourly Avg
Lightning NO
 setenv CTM_LTNGDIAG_2  "$OUTDIR/CCTM_LTNGCOL_${CTM_APPL}.nc -v"     #> Column Total
Lightning NO
 setenv CTM_VEXT_1      "$OUTDIR/CCTM_VEXT_${CTM_APPL}.nc -v"       #> On-Hour 3D Concs at
select sites


 #> set floor file (neg concs)
 setenv FLOOR_FILE ${OUTDIR}/FLOOR_${CTM_APPL}.txt


 #> look for existing log files and output files
 ( ls CTM_LOG_???.${CTM_APPL} > buff.txt ) >& /dev/null
 ( ls ${LOGDIR}/CTM_LOG_???.${CTM_APPL} >> buff.txt ) >& /dev/null
 set log_test = `cat buff.txt`; rm -f buff.txt


 set OUT_FILES = (${FLOOR_FILE} ${S_CGRID} ${CTM_CONC_1} ${A_CONC_1} ${MEDIA_CONC}
${CTM_DRY_DEP_1} $CTM_DEPV_DIAG $B3GTS_S $SOILOUT $CTM_WET_DEP_1 $CTM_WET_DEP_2
$CTM_PMDIAG_1 $CTM_APMDIAG_1 $CTM_RJ_1 $CTM_RJ_2 $CTM_RJ_3 $CTM_SSEMIS_1
$CTM_DUST_EMIS_1 $CTM_IPR_1 $CTM_IPR_2 $CTM_IPR_3 $CTM_IRR_1 $CTM_IRR_2
```

```
$CTM_IRR_3 $CTM_DRY_DEP_MOS $CTM_DRY_DEP_FST $CTM_DEPV_MOS $CTM_DEPV_FST
$CTM_VDIFF_DIAG $CTM_VSED_DIAG $CTM_LTNGDIAG_1 $CTM_LTNGDIAG_2 $CTM_VEXT_1 )

 if ( $?CTM_ISAM ) then

   if ( $CTM_ISAM == 'Y' || $CTM_ISAM == 'T' ) then

     set OUT_FILES = (${OUT_FILES} ${SA_ACONC_1} ${SA_CONC_1} ${SA_DD_1} ${SA_WD_1}
${SA_CGRID_1} )

   endif

 endif

 if ( $?CTM_DDM3D ) then

   if ( $CTM_DDM3D == 'Y' || $CTM_DDM3D == 'T' ) then

     set OUT_FILES = (${OUT_FILES} ${CTM_SENS_1} ${A_SENS_1} ${CTM_SWETDEP_1}
${CTM_SDRYDEP_1} )

   endif

 endif

 set OUT_FILES = `echo $OUT_FILES | sed "s; -v;;g" | sed "s;MPI:;;g" `

 ( ls $OUT_FILES > buff.txt ) >& /dev/null

 set out_test = `cat buff.txt`; rm -f buff.txt


 #> delete previous output if requested

 if ( $CLOBBER_DATA == true || $CLOBBER_DATA == TRUE  ) then

   echo

   echo "Existing Logs and Output Files for Day ${TODAYG} Will Be Deleted"


   #> remove previous log files

   foreach file ( ${log_test} )

     #echo "Deleting log file: $file"

     /bin/rm -f $file

   end


   #> remove previous output files

   foreach file ( ${out_test} )

     #echo "Deleting output file: $file"
```

```
      /bin/rm -f $file
    end
    /bin/rm -f ${OUTDIR}/CCTM_EMDIAG*${RUNID}_${YYYYMMDD}.nc

  else
    #> error if previous log files exist
    if ( "$log_test" != "" ) then
      echo "*** Logs exist - run ABORTED ***"
      echo "*** To overide, set CLOBBER_DATA = TRUE in run_cctm.csh ***"
      echo "*** and these files will be automatically deleted. ***"
      exit 1
    endif

    #> error if previous output files exist
    if ( "$out_test" != "" ) then
      echo "*** Output Files Exist - run will be ABORTED ***"
      foreach file ( $out_test )
        echo " cannot delete $file"
      end
      echo "*** To overide, set CLOBBER_DATA = TRUE in run_cctm.csh ***"
      echo "*** and these files will be automatically deleted. ***"
      exit 1
    endif
  endif

  #> for the run control ...
  setenv CTM_STDATE     $YYYYJJJ
  setenv CTM_STTIME     $STTIME
  setenv CTM_RUNLEN     $NSTEPS
  setenv CTM_TSTEP      $TSTEP
  setenv INIT_CONC_1 $ICpath/$ICFILE
```

```
  setenv BNDY_CONC_1 $BCpath/$BCFILE
  setenv OMI $OMIpath/$OMIfile
  setenv OPTICS_DATA $OMIpath/$OPTfile
#setenv XJ_DATA $JVALpath/$JVALfile

  #> species defn & photolysis
  setenv gc_matrix_nml ${NMLpath}/GC_$MECH.nml
  setenv ae_matrix_nml ${NMLpath}/AE_$MECH.nml
  setenv nr_matrix_nml ${NMLpath}/NR_$MECH.nml
  setenv tr_matrix_nml ${NMLpath}/Species_Table_TR_0.nml

  #> check for photolysis input data
  setenv CSQY_DATA ${NMLpath}/CSQY_DATA_$MECH

  if (! (-e $CSQY_DATA ) ) then
    echo " $CSQY_DATA  not found "
    exit 1
  endif
  if (! (-e $OPTICS_DATA ) ) then
    echo " $OPTICS_DATA  not found "
    exit 1
  endif

# ========================================================================
#> Execution Portion
# ========================================================================

  #> Print attributes of the executable
  if ( $CTM_DIAG_LVL != 0 ) then
    ls -l $BLD/$EXEC
    size $BLD/$EXEC
```

```
  unlimit
  limit
endif

#> Print Startup Dialogue Information to Standard Out
echo
echo "CMAQ Processing of Day $YYYYMMDD Began at `date`"
echo

#> Executable call for single PE, uncomment to invoke
#( /usr/bin/time -p $BLD/$EXEC ) |& tee buff_${EXECUTION_ID}.txt

#> Executable call for multi PE, configure for your system
# set MPI = /usr/local/intel/impi/3.2.2.006/bin64
# set MPIRUN = $MPI/mpirun
( /usr/bin/time -p mpirun -np $NPROCS $BLD/$EXEC ) |& tee buff_${EXECUTION_ID}.txt

#> Harvest Timing Output so that it may be reported below
set rtarray = "${rtarray} `tail -3 buff_${EXECUTION_ID}.txt | grep -Eo '[+-]?[0-9]+([.][0-9]+)?' | head -1` "
rm -rf buff_${EXECUTION_ID}.txt

#> Abort script if abnormal termination
if ( ! -e $OUTDIR/CCTM_CGRID_${CTM_APPL}.nc ) then
  echo
  echo "**************************************************************"
  echo "** Runscript Detected an Error: CGRID file was not written. **"
  echo "**   This indicates that CMAQ was interrupted or an issue   **"
  echo "**   exists with writing output. The runscript will now     **"
  echo "**   abort rather than proceeding to subsequent days.       **"
  echo "**************************************************************"
```

```
   break
  endif

  #> Print Concluding Text
  echo
  echo "CMAQ Processing of Day $YYYYMMDD Finished at `date`"
  echo
  echo "\\\=====\\\=====\\\=====\\\=====/////=====/////=====/////=====/////"
  echo

# ======================================================================
#> Finalize Run for This Day and Loop to Next Day
# ======================================================================

  #> Save Log Files and Move on to Next Simulation Day
  mv CTM_LOG_???.${CTM_APPL} $LOGDIR
  if ( $CTM_DIAG_LVL != 0 ) then
    mv CTM_DIAG_???.${CTM_APPL} $LOGDIR
  endif

  #> The next simulation day will, by definition, be a restart
  setenv NEW_START false

  #> Increment both Gregorian and Julian Days
  set TODAYG = `date -ud "${TODAYG}+1days" +%Y-%m-%d` #> Add a day for tomorrow
  set TODAYJ = `date -ud "${TODAYG}" +%Y%j` #> Convert YYYY-MM-DD to YYYYJJJ

end  #Loop to the next Simulation Day

# ======================================================================
#> Generate Timing Report
```

```
# ==================================================================
set RTMTOT = 0
foreach it ( `seq ${NDAYS}` )
   set rt = `echo ${rtarray} | cut -d' ' -f${it}`
   set RTMTOT = `echo "${RTMTOT} + ${rt}" | bc -l`
end

set RTMAVG = `echo "scale=2; ${RTMTOT} / ${NDAYS}" | bc -l`
set RTMTOT = `echo "scale=2; ${RTMTOT} / 1" | bc -l`

echo
echo "================================="
echo "  ***** CMAQ TIMING REPORT *****"
echo "================================="
echo "Start Day: ${START_DATE}"
echo "End Day:   ${END_DATE}"
echo "Number of Simulation Days: ${NDAYS}"
echo "Domain Name:               ${GRID_NAME}"
echo "Number of Grid Cells:      ${NCELLS}  (ROW x COL x LAY)"
echo "Number of Layers:          ${NZ}"
echo "Number of Processes:       ${NPROCS}"
echo "   All times are in seconds."
echo
echo "Num  Day        Wall Time"
set d = 0
set day = ${START_DATE}
foreach it ( `seq ${NDAYS}` )
   # Set the right day and format it
   set d = `echo "${d} + 1"  | bc -l`
   set n = `printf "%02d" ${d}`
```

```
    # Choose the correct time variables
    set rt = `echo ${rtarray} | cut -d' ' -f${it}`

    # Write out row of timing data
    echo "${n}   ${day}   ${rt}"

    # Increment day for next loop
    set day = `date -ud "${day}+1days" +%Y-%m-%d`
end
echo "     Total Time = ${RTMTOT}"
echo "      Avg. Time = ${RTMAVG}"

exit
```