```fortran
! This file defines various arrays, variables, and parameters.
! The first variables must be entered here by hand
!  whenever to match the case being modeled.
! Note the ordering of these lines is cosmetic and has no functional effect.


C*********************************************************************
c *** These parameters are case-specific and need to be manually entered.
C*********************************************************************
      parameter(npx=165,npy=165,npz=61) ! Grid size array-sizing parameters
     ! Note: nx,ny,nz must not be greater than npx,npy,npz
     parameter(npbglr=npz) ! Number of soil layers (Free Space Only: Set npbglr=0)
     parameter(npexcnpts=99) ! Number of excitation points
        ! npexcnpts is arbitrary if a built-in antenna (i.e. monopole) is used.
     parameter(npout=99) ! Number of output files per output time step


C*********************************************************************
c *** These parameters are semi-permanent and need to occasionally be manually entered.
C*********************************************************************
     parameter(ntmt=34) ! Number of total materials  !nwsltp
     parameter(nbmt=21) ! Number of built-int materials  !nwsltp
     parameter(nemt=ntmt-nbmt) ! Number of externally-inputted materials


C*********************************************************************
c *** These parameters are permanent and do not need to be manually entered.
C*********************************************************************
     parameter(eps0=8.854e-12,xmu0=1.2566306e-6,eta0=376.733341)  ! Constitutive Properties
     parameter(c0=2.99792458e8)  ! Speed of light in free space
     parameter(pi=3.1415926536)



C*********************************************************************
c *** These are variables and do not need to be manually entered.
C*********************************************************************

c *** Space & Time Dimension Variables
     common/space0/ndim  ! dimension configuration
     common/space1/nx,ny,nz,nx1,ny1,nz1  ! grid size
     common/space2/delx,dely,delz  ! grid cell size
     common/time/dt  ! time step size

c *** Material Variables
     common/mat/namt  ! number of available materials
     common/mts/mtx(npx,npy,npz),mty(npx,npy,npz),mtz(npx,npy,npz)  ! material id matrices
     common/perm/epsr(ntmt),eps(ntmt)  ! Permittivity parameters
     common/disp/a1(ntmt),b0(ntmt),b1(ntmt),b2(ntmt)  ! dispersion parameters

c *** Background Medium Variables
     common/geom/ngeom  ! Background medium geometry choice
     common/layer/nzsoil,nbght(npbglr),nbgth(npbglr),ibgtp(npbglr)

c *** Foreground Object Variables
     character*20 matname  ! material input file name
     common/materialfile/matname
```

```fortran
c *** Monopole Antenna Object Variables:
      common/monopole/iantctr,jantctr,kanttop,kshieldlen,kcorelen
     $  ,icorerad,idieth,ishieldth
     $  ,idierad,ishieldrad,kcorebtm,kshieldbtm


C *** Source, Observation, & Object Variables  SelfNote: Probably needs removed
      real*4 xobject,yobject,zobject  ! Object coordinates
      real*4 fll,fww,fhh  ! Object dimensions <sb>
      common/object/xobject,yobject,zobject,fll,fww,fhh


c *** Excitation Variables
      common/excn1/mx(npexcnpts),my(npexcnpts),mz(npexcnpts)  ! Excitation point coordinates
      common/excn2/esx(npexcnpts),esy(npexcnpts),esz(npexcnpts)  ! Directional excitation strengths


c *** Field Component Output Variables
      character*20 loc,tsn,scn,outname  ! material location, time step #, slice coord #, output
      file name
      common/output1/iii,loc,tsn,scn,outname,nout  ! Output time step #, ...
      common/output2/mfld(npout),mdir(npout),mloc(npout)  ! Output file specifiers


c *** Field Component Current Value Matrices
      real*4 exs,eys,ezs,hxs,hys,hzs
      common/xscat/exs(npx,npy,npz),hxs(npx,npy,npz)
      common/yscat/eys(npx,npy,npz),hys(npx,npy,npz)
      common/zscat/ezs(npx,npy,npz),hzs(npx,npy,npz)


c *** Field Component Past Value Matrices
      real*4 exs1(npx,npy,npz),exs2(npx,npy,npz)
      real*4 eys1(npx,npy,npz),eys2(npx,npy,npz)
      real*4 ezs1(npx,npy,npz),ezs2(npx,npy,npz)
      real*4 exs3(npx,npy,npz),hxs1(npx,npy,npz)
      real*4 eys3(npx,npy,npz),hys1(npx,npy,npz)
      real*4 ezs3(npx,npy,npz),hzs1(npx,npy,npz)
      common/esoil1/exs1,eys1,ezs1
      common/esoil2/exs2,eys2,ezs2
      common/esoil3/exs3,eys3,ezs3
      common/hsoil1/hxs1,hys1,hzs1


c *** Boundary Condition Past Value Matrices
      common/radsav1/eysx1(4,npy,npz),ezsx1(4,npy,npz),
     $              ezsy1(npx,4,npz),exsy1(npx,4,npz),
     $              exsz1(npx,npy,4),eysz1(npx,npy,4)
      common/radsav2/eysx2(4,npy,npz),ezsx2(4,npy,npz),
     $              ezsy2(npx,4,npz),exsy2(npx,4,npz),
     $              exsz2(npx,npy,4),eysz2(npx,npy,4)
      common/radsav3/eysx3(4,npy,npz),ezsx3(4,npy,npz),
     $              ezsy3(npx,4,npz),exsy3(npx,4,npz),
     $              exsz3(npx,npy,4),eysz3(npx,npy,4)


c *** Update Equation Constants
      common/hfld/dtxmu,dtymu,dtzmu  ! Magnetic field coefficients
      common/efld/dtxeps(ntmt),dtyeps(ntmt),dtzeps(ntmt)  ! Electric field coefficients
```

```fortran
     $    ,dsp0(ntmt),dsp1(ntmt),dsp2(ntmt),dsp3(ntmt)


c *** Absorbing Boundary Constants
      real*4 xx1X,xx1Y,xx1Z  ! 1st Order
      common/abc1/xx1X,xx1Y,xx1Z
      real*4 xx2,xx3,uu1X,uu1Y,uu1Z,uu2,uu3  ! 2nd Order
      common/abc2/xx2(ntmt),xx3(ntmt)
     $    ,uu1X(ntmt),uu1Y(ntmt),uu1Z(ntmt),uu2(ntmt),uu3
```