

1 Modifications of the TALYS file 'talys.cmb'.

The TALYS source file 'talys.cmb' holds the declaration of all variables used in TALYS. In order to force TALYS to use a specific file provided by the user, the name of that file must be present inside TALYS in the form of a variable. For that purpose, the variable 'yieldfileid' is declared in 'talys.cmb'. The unique file name given by the user as the value of the specifically created keyword 'gefissionfileid' is then stored in the variable 'yieldfileid'. The TALYS source file 'talys.cmb' is included in every subroutine that uses variables, and hence the new variable with the file name is passed internally to all the subroutines that needs to be modified in order for TALYS to use the file provided by the user. The modifications are highlighted in yellow.

```
c
c TALYS.CMB for TALYS-1.962
c
c +-----+
c Author: Arjan Koning, Stephane Hilaire and Stephane Gorielyc Date : March 10, 2023
c Task : File with common blocks and declarations for TALYS
c Edited: Peter Karlsson; Date: May 7 2023
c +-----+
c
c ***** Editing information *****
c Added "subroutine input7" to read unique GEF FY file.
c Declared variable "yieldfileid" as common character variable.
c Unique GEF fission fragment yield file id is used to read specific yield
c file in TALYS library when multi-threading.
c *****
c
c ***** Every local and global variable must be declared *****
c
c      implicit none
c
c ***** All parameters for the array dimensions are set *****
c
c Consult the various subroutines for an explanation of the variables.
c To enable optimal calculations on computers of different size, we
c introduce the variable memorypar.
c
c      integer memorypar,numpar,numiso,numelem,numl,numlines,numpop,
c      + numenin,numZ,numN,numZph,numNph,numbar,nummt,numgam,
c      + numrange,numadj,numenadj,numlev,numisom,numflux,numfile,
c      + numlev2,numrotcc,numgamqrpa,numTqrpa,numomp,numompadj,
c      + numjlm,numrot,nummatchT,numdens,numdensracap,numen,
c      + numang,numangcont,numexc,numconf,numJph,numparx,numfact,
c      + numbins,numex,numJ,numenrec,numangrec,numendisc,numen2,
c      + numenmsd,numJmsd,nummsd,numbinfis,numbeta,numhill,
c      + numtrans,nummold,numgoe,numZastro,numNastro,numT,
c      + numcomp,numZchan,numNchan,numin,numip,numid,numit,numih,
c      + numia,numchantot,nummass,numneu,numA,numnu,numenlow,
c      + numtime,numpfns,numenout,numen6
c
c
c Use memorypar=2 for 64 Mb RAM
c Use memorypar=5 for 256 (or more) Mb RAM
c
c      parameter (
c      + memorypar=6,
cWindows seems to work only with memorypar=5
c + memorypar=5,
c      + numpar=6,
c      + numiso=20,
c      + numelem=124,
c      + numl=60,
c      + numlines=5000,
c      + numpop=1000,
c      + numenin=600,
c      + numZ=2+2*memorypar,
c      + numN=10+4*memorypar,
c      + numZph=4,
```

```

+ numNph=8,
+ numbar=3,
+ nummt=200,
+ numgam=6,
+ numrange=10,
+ numadj=500,
+ numenadj=1000,
c + numlev=3*memorypar*memorypar+2,
+ numlev=40,
+ numisom=10,
+ numflux=100,
+ numfile=100)
parameter (
+ numlev2=200,
+ numrotcc=4,
+ numgamqrpa=300,
+ numTqrpa=11,
+ numomp=500,
+ numompadj=13,
+ numjlm=200,
+ numrot=700,
+ nummatchT=4000,
+ numdens=60,
+ numdensracap=200,
+ numen=260,
+ numang=90,
+ numangcont=36,
+ numconf=72,
+ numexc=12,
+ numJph=30,
+ numparx=numexc/2,
+ numfact=6*numl)
parameter (
+ numbins=20*(memorypar-1),
+ numex=numlev+numbins,
+ numJ=40,
+ numenrec=4*(memorypar-1),
+ numangrec=9,
+ numendisc=400,
+ numen2=numen+numendisc,
+ numenmsd=18,
+ numJmsd=8,
+ nummsd=6,
+ numbinfis=1000,
+ numbeta=200,
+ numhill=20,
+ numtrans=numl*12*(numex+1)+numhill+1,
+ nummold=32,
+ numgoe=50)
parameter (
+ numZastro=4,
+ numNastro=4,
+ numT=30,
+ numcomp=100,
+ numZchan=6,
+ numNchan=10,
+ numin=8,
+ numip=4,
+ numid=2,
+ numit=1,
+ numih=1,
+ numia=3,
+ numchantot=35*(memorypar-1),
+ nummass=414,
+ numneu=nummass-numelem,
+ numA=numZ+numN,
+ numnu=50,
+ numenlow=20,
+ numtime=100,

```

```

+ numpfns=300,
+ numenout=1000,
+ numen6=memorypar*1700)
c
c ***** block data constants0 *****
c
common /constants0c/ parname(-1:numpar),parsym(-1:numpar),
+ nuc(numelem),isochar(-1:numisom),
+ cparity(-1:1)
common /constants0i/ parZ(0:numpar),parN(0:numpar),
+ parA(0:numpar),magic(8)
common /constants0r/ parspin(0:numpar),pi,e2,hbar,clight,kT,emass,
+ avogadro,qelem
common /constants0d/ parmass(0:numpar),excmass(0:numpar),amu
character*1 parsym,isochar,cparity
character*2 nuc
character*8 parname
integer parZ,parN,parA,magic
real parspin,pi,e2,hbar,clight,kT,emass,avogadro,qelem
double precision parmass,excmass,amu
c
c ***** subroutine machine *****
c
common /machinec/ nulldev,path
character*13 nulldev
character*132 path
c
c ***** subroutine constants *****
c
common /constantsl/ flagffruns,flagrpruns
common /constantsc/ natstring(numiso)
common /constantsi/ iso
common /constantser/ twopi,pi2,sqrttwopi,fourpi,deg2rad,rad2deg,
+ onethird,twothird,twopihbar,hbarc,pi2h2c2,
+ pi2h3c2,amupi2h3c2,amu4pi2h2c2,sgn(0:2*numl),
+ pardis,fislim,Emaxtalys
logical flagffruns,flagrpruns
character*4 natstring
integer iso
real twopi,pi2,sqrttwopi,fourpi,deg2rad,rad2deg,onethird,
+ twothird,twopihbar,hbarc,pi2h2c2,pi2h3c2,amupi2h3c2,
+ amu4pi2h2c2,sgn,pardis,fislim,Emaxtalys
c
c ***** subroutine readinput *****
c
common /readinputc/ inline(numlines)
common /readinputi/ nlines0
character*132 inline
integer nlines0
c
c ***** subroutine input1 *****
c
common /input1l/ flaginitpop,flagnatural,flagmicro,flagastro,
+ flagbest,flagbestbr,flagbestend,flagfit
common /input1c/ energyfile,ptype0,Starget
common /input1i/ nlines,Ztarget,k0,Atarget,Ntarget,Zinit,Ninit,
+ Ainit,nin,numinc,npopE,npopJ,npopP,Ztarget0,
+ Atarget0
common /input1r/ eninc(0:numen6+2),enincmin,enincmax,Estop,
+ EdistE(0:numpop),PdistE(0:numpop),
+ PdistJP(0:numpop,0:numJ,-1:1)
logical flaginitpop,flagnatural,flagmicro,flagastro,
+ flagbest,flagbestbr,flagbestend,flagfit
character*1 ptype0
character*2 Starget
character*132 energyfile
integer nlines,Ztarget,k0,Atarget,Ntarget,Zinit,Ninit,Ainit,
+ nin,numinc,npopE,npopJ,npopP,Ztarget0,Atarget0
real eninc,enincmin,enincmax,Estop,EdistE,PdistE,PdistJP

```

```

c
c ***** subroutine input2 *****
c
common /input2l/ flagcol(0:numZ,0:numN),flagcolall,flagomponly,
+ flagequi,flagequispec,flagpopmev,flagracap
common /input2c/ outtype(0:numpar)
common /input2i/ maxZ,maxN,nbins0,segment,nlevmax,nlevmaxres,
+ Ltarget,Lisoinp,core,gammax,nangle,nanglecont,
+ maxenrec,masmodel,disctable,ldmodelall,wmode,
+ preeqmode,mpreeqmode,phmodel,nlev(0:numZ,0:numN),
+ ldmodel(0:numZ,0:numN),nlevbin(0:numpar),
+ ldmodelracap,skipCN(0:numZ,0:numN),maxZrp,
+ maxNrp,ldmodelCN,wfcfactor
common /input2r/ isomer,eninclo,specfacth(0:numZ,0:numN),
+ spectfacexp(0:numZ,0:numN,0:numlev)
logical flagcol,flagcolall,flagomponly,flagequi,
+ flagequispec,flagpopmev,flagracap
character*1 outtype
integer maxZ,maxN,nbins0,segment,nlevmax,nlevmaxres,
+ Ltarget,Lisoinp,core,gammax,nangle,
+ nanglecont,maxenrec,masmodel,disctable,
+ ldmodelall,wmode,mpreeqmode,mpreeqmode,phmodel,
+ nlev,ldmodel,nlevbin,ldmodelracap,skipCN,maxZrp,
+ maxNrp,ldmodelCN,wfcfactor
real isomer,eninclo,specfacth,spectfacexp
c
c ***** subroutine input3 *****
c
common /input3l/ flagciscalc,flaginccalc,flagendfecis,flagrel,
+ flagcomp,flagfullhf,flaggiant0,flagsfglobal,
+ flagpecomp,flagsurface,flag2comp,flagchannels,
+ flagfission,flagparity,flaghbstate,flagclass2,
+ flagbasic,flagciscalc,flagcpang,flagcisdwba,
+ flagonestep,flaglocalomp,flagdisp,flagompall,
+ flagincadj,flagautorot,flagstate,
+ flagsys(0:numpar),flagrot(0:numpar),flagasys,
+ flaggshell,flagmassdis,flagffevap,flagfisfeed,
+ flagffspin,flagendf,flagendfdet,flagrecoil,
+ flaglabddx,flagrecoilav,flagEchannel,
+ flagreaction,flagastrog,flagastroex,flagexpmass,
+ flagjlm,flagriplomp,flagriplrisk,flagngfit,
+ flagnnfit,flagnffit,flagnafit,flaggnorm,
+ flagpnfit,flaggnfit,flagdnfit,flaganfit,
+ flagisofit
common /input3c/ optmod(0:numZph,0:numNph,numpar)
common /input3i/ pespinmodel,maxband,maxrot,strength,strengthM1,
+ fymodel,ffmodel,nonthermlev,riplomp(numpar),
+ pfnsmodel
common /input3r/ ewfc,epreeq,emulpre
logical flagciscalc,flaginccalc,flagendfecis,flagrel,
+ flagcomp,flagfullhf,flaggiant0,flagsfglobal,
+ flagpecomp,
+ flagsurface,flag2comp,flagchannels,flagfission,
+ flagparity,flaghbstate,flagclass2,flagbasic,
+ flagciscalc,flagcpang,flagcisdwba,flagonestep,
+ flaglocalomp,flagdisp,flagompall,flagincadj,
+ flagautorot,flagstate,flagsys,flagrot,flagasys,
+ flaggshell,flagmassdis,flagffevap,flagfisfeed,
+ flagffspin,flagendf,flagendfdet,flagrecoil,
+ flaglabddx,flagrecoilav,flagEchannel,flagreaction,
+ flagastrog,flagastroex,flagexpmass,flagjlm,
+ flagriplomp,flagriplrisk,flagngfit,flagnnfit,
+ flagnafit,flagnffit,flaggnorm,flagpnfit,flaggnfit,
+ flagdnfit,flaganfit,flagisofit
character*132 optmod
integer pespinmodel,maxband,maxrot,strength,strengthM1,
+ fymodel,ffmodel,nonthermlev,riplomp,pfnsmodel
real ewfc,epreeq,emulpre
c

```

```

c ***** subroutine input4 *****
c
      common /input4l/ flagmain,flagpop,flagcheck,flagoutomp,flagdirect,
+ flaginverse,flaggamma,flaglevels,flagdensity,
+ flagdisc,flagfisout,flagtransen,flagpeout,
+ flagang,flaglegendre,flagspec,flagres,flaggroup,
+ flagddx,flagoutdwba,flaggamdis,flagoutecis,
+ flagecissave,flagexc,flagelectron,flagspher,
+ flagcoulomb,flagupbend,flagcolldamp,flagctmglob,
+ flagpartable,flagurrnjoy,flagprod,flagoutfy,
+ altomp(numpar),flagbinspec,flagrpevap,flagdecay,
+ flagfispartdamp,flagoutkd,flagsoukho,
+ flagsoukhoinp
      common /input4c/ reslib
      common /input4i/ maxchannel,ddxmode,pairmodel,fismodel,
+ fismodelalt,alphaomp,deuteronomp,lurr,gefran
      common /input4r/ eadd,eaddel,cglobal,pglobal,Tres,soswitch,eurr,
+ Rspincutff
      logical flagmain,flagpop,flagcheck,flagoutomp,flagdirect,
+ flaginverse,flaggamma,flaglevels,flagdensity,flagdisc,
+ flagfisout,flagtransen,flagpeout,flagang,flaglegendre,
+ flagspec,flagres,flaggroup,flagddx,flagoutdwba,flaggamdis,
+ flagoutecis,flagecissave,flagexc,flagelectron,flagspher,
+ flagcoulomb,flagupbend,flagcolldamp,flagctmglob,
+ flagpartable,flagurrnjoy,flagprod,flagoutfy,altomp,
+ flagbinspec,flagrpevap,flagdecay,flagfispartdamp,
+ flagoutkd,flagsoukho,flagsoukhoinp
      character*132 reslib
      integer maxchannel,ddxmode,pairmodel,fismodel,fismodelalt,
+ alphaomp,deuteronomp,lurr,gefran
      real eadd,eaddel,cglobal,pglobal,Tres,soswitch,eurr,
+ Rspincutff
c
c ***** subroutine input5 *****
c
      common /input5l/ flagrescue,ompadjustF(numpar),preeqadjust,
+ ompadjustp(numpar),
+ adjustTJ(0:numZ,0:numN,-1:numpar),
+ ldadjust(0:numZ,0:numN),
+ _gamadjust(0:numZ,0:numN),
+ fisadjust(0:numZ,0:numN)
      common /input5c/ levelfile(0:numZ),deformfile(0:numZ),
+ hbtransfile(0:numZ,0:numN),adjustkey(numadj),
+ adjustfile(numadj),clas2file(0:numZ,0:numN),
+ optmodfileN(0:numZ),optmodfileP(0:numZ),
+ Exlfile(0:numZ,0:numN,0:1,numgam),
+ densfile(0:numZ,0:numN),
+ radialfile(0:numZ),ompenenergyfile,yieldfile,
+ rescuefile(nummt,-1:numisom),unitTirrad(5),
+ unitTcool(5),radiounit,yieldunit,masmdir
      common /input5i/ spincutmodel,shellmodel,kvibmodel,
+ Nlow(0:numZ,0:numN,0:numbar),
+ Ntop(0:numZ,0:numN,0:numbar),msdbins,nanglerec,
+ axtype(0:numZ,0:numN,numbar),radialmodel,
+ fismodelx(0:numZ,0:numN),breakupmodel,jlmmodel,
+ ompadjustN(numpar,numompadj),Nadjust,
+ adjustix(numadj,4),nenadjust(numadj),nTmax,
+ Tirrad(5),Tcool(5),nbinsff
      common /input5rA/ Rspincut,alphaId(0:numZ,0:numN),
+ betald(0:numZ,0:numN),Rspincutpreeq,
+ gammashell1(0:numZ,0:numN),gammashell2,
+ pairconstant,Pshiftconstant(0:numZ,0:numN),
+ Ufermi(0:numZ,0:numN,0:numbar),
+ cfermi(0:numZ,0:numN,0:numbar),Kph,M2constant,
+ M2limit,M2shift,Rpipi,Rnunu,Rpinu,Rnupi,Esurf0,
+ Rgamma,elwidth,xscaptherm(-1:numisom),
+ xspttherm(-1:numisom),xsalphatherm(-1:numisom),
+ massnucleus(0:numZ+4,0:numN+4),
+ massexcress(0:numZ+4,0:numN+4),

```

```

+ alev(0:numZ,0:numN),alimit(0:numZ,0:numN),
+ gammald(0:numZ,0:numN),pair(0:numZ,0:numN),
+ deltaW(0:numZ,0:numN,0:numbar),
+ Exmatch(0:numZ,0:numN,0:numbar),
+ T(0:numZ,0:numN,0:numbar),
+ E0(0:numZ,0:numN,0:numbar),
+ s2adjust(0:numZ,0:numN,0:numbar),
+ Krotconstant(0:numZ,0:numN,0:numbar),
+ beta2(0:numZ+4,0:numN+4,0:numbar),
+ ctable(0:numZ,0:numN,0:numbar),
+ ptable(0:numZ,0:numN,0:numbar),
+ ctableadjust(0:numZ,0:numN,0:numbar),
+ ptableadjust(0:numZ,0:numN,0:numbar)
common /input5rB/ g(0:numZ,0:numN),gp(0:numZ,0:numN),
+ gn(0:numZ,0:numN),gamgam(0:numZ,0:numN),
+ D0(0:numZ,0:numN),Risomer(0:numZ,0:numN),
+ etable(0:numZ,0:numN,0:1,numgam),
+ ftable(0:numZ,0:numN,0:1,numgam),
+ wtable(0:numZ,0:numN,0:1,numgam),
+ etableadjust(0:numZ,0:numN,0:1,numgam),
+ ftableadjust(0:numZ,0:numN,0:1,numgam),
+ wtableadjust(0:numZ,0:numN,0:1,numgam),
+ egr(0:numZ,0:numN,0:1,numgam,2),
+ ggr(0:numZ,0:numN,0:1,numgam,2),
+ sgr(0:numZ,0:numN,0:1,numgam,2),
+ epr(0:numZ,0:numN,0:1,numgam,2),
+ gpr(0:numZ,0:numN,0:1,numgam,2),
+ tpr(0:numZ,0:numN,0:1,numgam,2),
+ egradjust(0:numZ,0:numN,0:1,numgam,2),
+ ggradjust(0:numZ,0:numN,0:1,numgam,2),
+ sgradjust(0:numZ,0:numN,0:1,numgam,2),
+ epradjust(0:numZ,0:numN,0:1,numgam,2),
+ gpradjust(0:numZ,0:numN,0:1,numgam,2),
+ tpradjust(0:numZ,0:numN,0:1,numgam,2),
+ upbend(0:numZ,0:numN,0:1,numgam,3),
+ fiso(-1:numpar),fisom(-1:numpar),
+ fisominit(-1:numpar),
+ fbarrier(0:numZ,0:numN,numbar),
+ fwidth(0:numZ,0:numN,numbar),
+ bdamp(0:numZ,0:numN,numbar),
+ fbaradjust(0:numZ,0:numN,numbar),
+ fwidthadjust(0:numZ,0:numN,numbar),
+ bdampadjust(0:numZ,0:numN,numbar),
+ betafiscor(0:numZ,0:numN),
+ betafiscoradjust(0:numZ,0:numN),
+ vfiscor(0:numZ,0:numN),
+ vfiscoradjust(0:numZ,0:numN),
+ Rtransmom(0:numZ,0:numN,numbar),
+ Rclass2mom(0:numZ,0:numN,numbar)
common /input5rC/ widthc2(0:numZ,0:numN,numbar),Cknock(0:numpar),
+ Cstrip(0:numpar),Cbreak(0:numpar),Emsdmin,
+ v1adjust(numpar),v2adjust(numpar),
+ v3adjust(numpar),v4adjust(numpar),
+ rvadjust(numpar),avadjust(numpar),
+ w1adjust(numpar),w2adjust(numpar),
+ w3adjust(numpar),w4adjust(numpar),
+ rwadjust(numpar),awadjust(numpar),
+ rvdadjust(numpar),avdadjust(numpar),
+ d1adjust(numpar),d2adjust(numpar),
+ d3adjust(numpar),rwdadjust(numpar),
+ awdadjust(numpar),rvsoadjust(numpar),
+ avsoadjust(numpar),vso1adjust(numpar),
+ vso2adjust(numpar),rwsoadjust(numpar),
+ awsoadjust(numpar),wso1adjust(numpar),
+ wso2adjust(numpar),rcadjust(numpar),
+ Ejoin(numpar),Vinfadjust(numpar),
+ grescue(nummt,-1:numisom),lvadjust,lwadjust,
+ lv1adjust,Cnubar1,Cnubar2,Tmadjust,Fsadjust
common /input5rD/ lw1adjust,lvsoadjust,lwsoadjust,aradialcor,

```

```

+ adepthcor,aadjust(0:numZ,0:numN),
+ Tadjust(0:numZ,0:numN,0:numbar),
+ E0adjust(0:numZ,0:numN,0:numbar),
+ Exmatchadjust(0:numZ,0:numN,0:numbar),
+ gndjust(0:numZ,0:numN),gpadjust(0:numZ,0:numN),
+ gadjust(0:numZ,0:numN),
+ gamgamadjust(0:numZ,0:numN),
+ ompadjustE1(numpar,numompadj,numrange),
+ ompadjustE2(numpar,numompadj,numrange),
+ ompadjustD(numpar,numompadj,numrange),
+ ompadjusts(numpar,numompadj,numrange),
+ adjustpar(numadj,4),Eadjust(numadj,numenadj),
+ Dadjust(numadj,numenadj),
+ TJadjust(0:numZ,0:numN,-1:numpar),
+ RprimeU,astroT9,
+ astroE,Ebeam,Eback,Ibeam,Area,rhotarget
logical flagrescue,preeqadjust,ompadjustF,ompadjustp,
+ adjustTJ,ldadjust,gamadjust,fisadjust
character*132 adjustkey,adjustfile
character*132 levelfile,deformfile,hbtransfile,clas2file,
+ optmodfileN,optmodfileP,Exlfile,densfile,radialfile,
+ ompenergyfile,rescuefile,masmdir,yieldfile
character*3 radiounit,yieldunit
character*1 unitTirrad,unitTcool
integer spincutmodel,shellmodel,kvibmodel,Nlow,Ntop,msdbins,
+ nanglerec,axtype,radialmodel,fismodelx,breakupmodel,
+ jlmmode,ompadjustN,Nadjust,adjustix,nenadjust,nTmax,
+ Tirrad,Tcool,nbinsff
real Rspincut,alphaId,betaId,gammashell1,Rspincutpreeq,
+ gammashell2,pairconstant,Pshiftconstant,Ufermi,
+ cfermi,Kph,M2constant,M2limit,
+ M2shift,Rpipi,Rnuu,Rpinu,Rnupi,Esurf0,Rgamma,
+ elwidth,xscaptherm,xsptherm,xsalphatherm,massnucleus,
+ massexcess,alev,alimit,gammald,pair,deltaW,Exmatch,T,
+ E0,s2adjust,Krotconstant,beta2,ctable,ptable,g,gp,gn,
+ ctableadjust,ptableadjust,gamgam,D0,etable,ftable,
+ etableadjust,ftableadjust,egr,ggr,sgr,epr,gpr,tpr,
+ egradjust,ggradjust,sgradjust,epradjust,gpradjust,
+ tpradjust,upbend,fiso,fisom,fbarrier,fwidth,bdamp,
+ fbaradjust,fwidthadjust,bdampadjust,betafiscor,
+ vfiscor,Rtransmom,Rclass2mom,widthc2,Cknoock,Cstrip,
+ Cbreak,Emsdmin,v1adjust,v2adjust,v3adjust,v4adjust,
+ rvadjust,avadjust,w1adjust,w2adjust,w3adjust,
+ w4adjust,rwadjust,awadjust,rvdadjust,avdadjust,
+ d1adjust,d2adjust,d3adjust,rwdadjust,awdadjust,
+ rvsoadjust,avsoadjust,vso1adjust,vso2adjust,
+ rwsoadjust,awsoadjust,wsoladjust,wso2adjust,
+ rcadjust,Ejoin,Vinfadjust,grescue,lvadjust,lvadjust,
+ lv1adjust,lv1adjust,lvsoadjust,lvsoadjust,aradialcor,
+ adepthcor,aadjust,Tadjust,E0adjust,Exmatchadjust,
+ gndjust,gpadjust,gadjust,gamgamadjust,ompadjustE1,
+ ompadjustE2,ompadjustD,ompadjusts,adjustpar,Eadjust,
+ Dadjust,TJadjust,RprimeU,astroT9,astroE,fisominit,
+ Ebeam,Eback,Ibeam,Area,rhotarget,wtble,wtbleadjust,
+ Cnubar1,Cnubar2,Tmadjust,Fsadjust,vfiscoradjust,
+ betafiscoradjust,Risomer
c
c ***** subroutine input6 *****
c
common /input61/ fileelastic,filespectrum(0:numpar),
+ fileangle(0:numlev),filediscrete(0:numlev),
+ filetotal,fileresidual,filechannels,filerecoil,
+ filefission,filegamdis,filedensity,filepsf,
+ flagintegral,flagsacs,flagcompo,flagblock
common /input6c/ xsfluxfile(numflux),fluxname(numflux)
common /input6i/ transpower,ddxcount(0:numpar),
+ ddxaccount(0:numpar),Nflux,integralexp(numflux)
common /input6r/ xseps,popeps,Rfiseps,fileddxe(0:numpar,numfile),
+ fileddxa(0:numpar,numfile)

```

```

common /input6d/ transeps
logical fileelastic,filespectrum,fileangle,filediscrete,filetotal,
+ fileresidual,filechannels,filerecoil,filefission,
+ filegamdis,filedensity,filepsf,flagsacs,flagintegral,
+ flagcompo,flagblock
character*132 xsfluxfile,fluxname
integer transpower,ddxcount,ddxaccount,Nflux
real xseps,popeps,Rfiseps,fileddxe,fileddxa,
+ integralexp
double precision transeps
C
C ***** subroutine input7 *****
C
common /input7c/ yieldfileid
character*132 yieldfileid
C
C ***** subroutine abundance *****
C
common /abundancei/ isotope(numiso),isonum
common /abundancer/ abun(numiso)
integer isotope,isonum
real abun
C
C ***** subroutine particles *****
C
common /particlesl/ parinclude(-1:numpar),parskip(-1:numpar)
logical parinclude,parskip
C
C ***** subroutine nuclides *****
C
common /nuclidesl/ strucexist(0:numZ,0:numN),
+ strucwrite(0:numZ,0:numN),
+ invexist(0:numZ,0:numN),primary
common /nuclidesi/ Zindex(0:numZ,0:numN,0:numpar),
+ Nindex(0:numZ,0:numN,0:numpar),
+ ZZ(0:numZ,0:numN,0:numpar),
+ NN(0:numZ,0:numN,0:numpar),
+ AA(0:numZ,0:numN,0:numpar),targetspin2,targetP
common /nuclidesr/ targetspin,targetE,Q(0:numpar),
+ coulbar(0:numpar)
common /nuclidesd/ tarmass
logical strucexist,strucwrite,invexist,primary
integer Zindex,Nindex,ZZ,NN,AA,targetspin2,targetP
real targetspin,targetE,Q,coulbar
double precision tarmass
C
C ***** subroutine masses *****
C
common /massesi/ gsparity(0:numZ+4,0:numN+4)
common /massesd/ nucmass(0:numZ+4,0:numN+4),
+ expmass(0:numZ+4,0:numN+4),
+ thmass(0:numZ+4,0:numN+4),
+ expmexc(0:numZ+4,0:numN+4),
+ thmexc(0:numZ+4,0:numN+4),
+ dumexc(0:numZ+4,0:numN+4),
+ specmass(0:numZ,0:numN,0:numpar),
+ redumass(0:numZ,0:numN,0:numpar)
common /massesr/ gsspin(0:numZ+4,0:numN+4),
+ beta4(0:numZ+4,0:numN+4)
integer gsparity
double precision nucmass,expmass,thmass,expmexc,thmexc,dumexc,
+ specmass,redumass
real gsspin,beta4
C
C ***** subroutine separation *****
C
common /separationr/ S(0:numZ,0:numN,0:numpar)
real S
C

```



```

c ***** subroutine levels *****
c
    common /levelsc/ jassign(0:numZ,0:numN,0:numlev2),
+ passign(0:numZ,0:numN,0:numlev2),
+ ENSDF(0:numZ,0:numN,0:numlev),
+ bassign(0:numZ,0:numN,0:numlev,0:numlev)
    common /levelsi/ nbranch(0:numZ,0:numN,0:numlev),Ltarget0,
+ branchlevel(0:numZ,0:numN,0:numlev,0:numlev),
+ nlevmax2(0:numZ,0:numN),
+ levnum(0:numZ,0:numN,0:numlev2),
+ parlev(0:numZ,0:numN,0:numlev2),
+ Lisomer(-1:numZ,-1:numN,0:numisom),
+ Nisomer(-1:numZ,-1:numN),Liso
    common /levelsrr/ edis(0:numZ,0:numN,0:numlev2),
+ jdis(0:numZ,0:numN,0:numlev2),
+ tau(0:numZ,0:numN,0:numlev2),
+ branchratio(0:numZ,0:numN,0:numlev,0:numlev),
+ conv(0:numZ,0:numN,0:numlev,0:numlev)
    character*1 jassign,passign,bassign
    character*18 ENSDF
    integer nbranch,Ltarget0,branchlevel,nlevmax2,parlev,Lisomer,
+ Nisomer,Liso,levnum
    real edis,jdis,tau,branchratio,conv
c
c ***** subroutine deformpar *****
c
    common /deformparc/ colltype(0:numZ,0:numN),
+ deftype(0:numZ,0:numN),
+ leveltype(0:numZ,0:numN,0:numlev2)
    common /deformpari/ ndef(0:numZ,0:numN),nrot(0:numZ,0:numN),
+ indexlevel(0:numZ,0:numN,0:numlev2),
+ indexcc(0:numZ,0:numN,0:numlev2),
+ vibband(0:numZ,0:numN,0:numlev2),
+ lband(0:numZ,0:numN,0:numlev2),
+ Kband(0:numZ,0:numN,0:numlev2),
+ iphonon(0:numZ,0:numN,0:numlev2)
    common /deformparr/ rotpar(0:numZ,0:numN,0:numrotcc),
+ deform(0:numZ,0:numN,0:numlev2),
+ defpar(0:numZ,0:numN,0:numlev2),
+ Irigid0(0:numZ,0:numN),
+ Irigid(0:numZ,0:numN,0:numbar)
    character*1 colltype,deftype,leveltype
    integer ndef,nrot,indexlevel,indexcc,vibband,lband,Kband,
+ iphonon
    real rotpar,deform,defpar,Irigid0,Irigid
c
c ***** subroutine resonancepar *****
c
    common /resonancepari/ Nrr(0:numZ,0:numN)
    common /resonanceparr/ dD0(0:numZ,0:numN),dgamgam(0:numZ,0:numN),
+ Eavres
    integer Nrr
    real dD0,dgamgam,Eavres
c
c ***** subroutine gammapar *****
c
    common /gammaparl/ qrpaexist(0:numZ,0:numN,0:1,numgam)
    common /gammapari/ ngr(0:numZ,0:numN,0:1,numgam),nTqrpa
    common /gammaparr/ kgr(numgam),
+ eqrpa(0:numZ,0:numN,0:numgamqrpa,0:1,numgam),
+ Tqrpa(numTqrpa),
+ fqrpa(0:numZ,0:numN,0:numgamqrpa,numTqrpa,0:1,
+ numgam)
    logical qrpaexist
    integer ngr,nTqrpa
    real kgr,eqrpa,Tqrpa,fqrpa
c
c ***** subroutine ompar *****
c

```

```

common /ompparrl/ ompglobal(0:numZ,0:numN,numpar),
+ disp(0:numZ,0:numN,numpar)
common /omppari/ omplines(0:numZ,0:numN,numpar)
common /ompparrA/ ef(0:numZ,0:numN,numpar),
+ rc0(0:numZ,0:numN,numpar),
+ rv0(0:numZ,0:numN,numpar),
+ av0(0:numZ,0:numN,numpar),
+ v1(0:numZ,0:numN,numpar),
+ v2(0:numZ,0:numN,numpar),
+ v3(0:numZ,0:numN,numpar),
+ w1(0:numZ,0:numN,numpar),
+ w2(0:numZ,0:numN,numpar),
+ w3(0:numZ,0:numN,numpar),
+ w4(0:numZ,0:numN,numpar),
+ rvd0(0:numZ,0:numN,numpar),
+ avd0(0:numZ,0:numN,numpar),
+ d1(0:numZ,0:numN,numpar),
+ d2(0:numZ,0:numN,numpar),
+ d3(0:numZ,0:numN,numpar),
+ rvso0(0:numZ,0:numN,numpar),
+ avso0(0:numZ,0:numN,numpar)
common /ompparrB/ vso1(0:numZ,0:numN,numpar),
+ vso2(0:numZ,0:numN,numpar),
+ wso1(0:numZ,0:numN,numpar),
+ wso2(0:numZ,0:numN,numpar),
+ eomp(0:numZph,0:numNph,numpar,0:numomp),
+ vomp(0:numZph,0:numNph,numpar,0:numomp,19),
+ V0(2),Vjoin(2),Wjoin(2),
+ Eompbeg0(numpar,10),Eompbeg1(numpar,10),
+ Eompend0(numpar,10),Eompend1(numpar,10)
logical ompglobal,disp
integer omplines
real ef,rc0,rv0,av0,v1,v2,v3,w1,w2,w3,w4,rvd0,avd0,d1,d2,d3,
+ rvso0,avso0,vso1,vso2,wso1,wso2,eomp,vomp,V0,Vjoin,Wjoin,
+ Eompbeg0,Eompbeg1,Eompend0,Eompend1
c
c ***** subroutine radialtable *****
c
common /radialtablel/ jlmexist(0:numZ,0:numN,numpar)
common /radialtablel/ rhojlmn(0:numZ,0:numN,numjlm,6),
+ rhojlm(0:numZ,0:numN,numjlm,6),
+ radjlm(0:numZ,0:numN,numjlm),
+ potjlm(0:numZ,0:numN,numjlm,6),
+ normjlm(0:numZ,0:numN,6)
logical jlmexist
real rhojlmn,rhojlm,radjlm,potjlm,normjlm
c
c ***** subroutine fissionpar *****
c
common /fissionpari/ nfisbar(0:numZ,0:numN),
+ nclass2(0:numZ,0:numN),nbinswkb,
+ nfistrhb(0:numZ,0:numN,numbar),
+ pfistrhb(0:numZ,0:numN,numbar,0:numlev),
+ nfisc2hb(0:numZ,0:numN,numbar),
+ pfisc2hb(0:numZ,0:numN,numbar,0:numlev)
common /fissionparr/ minertia(0:numZ,0:numN,numbar),
+ fecont(0:numZ,0:numN,numbar),
+ efistrhb(0:numZ,0:numN,numbar,0:numlev),
+ jfistrhb(0:numZ,0:numN,numbar,0:numlev),
+ minerc2(0:numZ,0:numN,numbar),
+ efisc2hb(0:numZ,0:numN,numbar,0:numlev),
+ jfisc2hb(0:numZ,0:numN,numbar,0:numlev)
integer nfisbar,nclass2,nbinswkb,nfistrhb,pfistrhb,nfisc2hb,
+ pfisc2hb
real minertia,fecont,efistrhb,jfistrhb,minerc2,efisc2hb,
+ jfisc2hb
c
c ***** subroutine wkb *****
c

```

```

common /wkbi/ nbeta,iextr(0:2*numbar),nextr
common /wkbr/ betafis(numbeta),vfis(numbeta),Vpos(2*numbar),
+ Vheight(2*numbar),Vwidth(2*numbar),
+ Uwkb(0:numZ,0:numN,0:numbins),
+ Twkb(0:numZ,0:numN,0:numbins,numbar),
+ Twkbdir(0:numZ,0:numN,0:numbins,numbar),
+ Twkbtrans(0:numZ,0:numN,0:numbins,numbar),
+ Twkbphase(0:numZ,0:numN,0:numbins,numbar)
integer nbeta,iextr,nextr
real betafis,vfis,Vpos,Vheight,Vwidth,Uwkb,Twkb,Twkbdir,
+ Twkbtrans,Twkbphase
c
c ***** block data fisdata *****
c
common /fisdatar/ barcof(7,7),l80cof(5,4),l20cof(5,4),lmxc(6,4),
+ egsc(5,6,4),x1b(6,11),x2b(6,11),x3b(10,20),
+ x1h(6,11),x2h(6,11),x3h(10,20)
real barcof,l80cof,l20cof,lmxc,egsc,x1b,x2b,x3b,x1h,x2h,x3h
c
c ***** subroutine rotband *****
c
common /rotbandi/ nfistrrot(0:numZ,0:numN,numbar),
+ pfistrrot(0:numZ,0:numN,numbar,0:numrot)
common /rotbandr/ efistrrot(0:numZ,0:numN,numbar,0:numrot),
+ jfistrrot(0:numZ,0:numN,numbar,0:numrot)
integer nfistrrot,pfistrrot
real efistrrot,jfistrrot
c
c ***** subroutine rotclass2 *****
c
common /rotclass2i/ nfisc2rot(0:numZ,0:numN,numbar),
+ pfisc2rot(0:numZ,0:numN,numbar,0:numrot)
common /rotclass2r/ Emaxclass2(0:numZ,0:numN,numbar),
+ efisc2rot(0:numZ,0:numN,numbar,0:numrot),
+ jfisc2rot(0:numZ,0:numN,numbar,0:numrot)
integer nfisc2rot,pfisc2rot
real Emaxclass2,efisc2rot,jfisc2rot
c
c ***** subroutine densitypar *****
c
common /densityparl/ ldparexist(0:numZ,0:numN)
common /densitypari/ Nlast(0:numZ,0:numN,0:numbar)
common /densityparr/ delta0(0:numZ,0:numN),
+ delta(0:numZ,0:numN,0:numbar),
+ Pshift(0:numZ,0:numN,0:numbar),
+ Pshiftadjust(0:numZ,0:numN,0:numbar),
+ Tcrit(0:numZ,0:numN),
+ aldrcrit(0:numZ,0:numN,0:numbar),
+ Econd(0:numZ,0:numN,0:numbar),
+ Ucrit(0:numZ,0:numN,0:numbar),
+ Scrit(0:numZ,0:numN,0:numbar),
+ Dcrit(0:numZ,0:numN,0:numbar)
logical ldparexist
integer Nlast
real delta0,delta,Pshift,Pshiftadjust,Tcrit,aldrcrit,Econd,
+ Ucrit,Scrit,Dcrit
c
c ***** subroutine densitymatch *****
c
common /densitymatchi/ NP,NLo
common /densitymatchr/ EL,EP,Ediscrete(0:numZ,0:numN,0:numbar),
+ scutoffdisc(0:numZ,0:numN,0:numbar),
+ logrho(nummatchT),temprho(nummatchT),Ttemp,
+ Exmemp,E0save
integer NP,NLo
real EL,EP,Ediscrete,scutoffdisc,logrho,temprho,Ttemp,Exmemp,
+ E0save
c
c ***** subroutine densitytable *****

```

```

c
common /densitytable/ ldexist(0:numZ,0:numN,0:numbar)
common /densitytablei/ nendens(0:numZ,0:numN)
common /densitytable/ edens(0:numdens),Edensmax(0:numZ,0:numN)
common /densitytable/
+ ldttable(0:numZ,0:numN,0:numdens,0:numJ,-1:1,0:numbar),
+ ldtottableP(0:numZ,0:numN,0:numdens,-1:1,0:numbar),
+ ldtottable(0:numZ,0:numN,0:numdens,0:numbar)
logical ldexist
integer nendens
real edens,Edensmax
double precision ldttable,ldtottableP,ldtottable

c
c ***** subroutine dtheory *****
c
common /d0theoryr/ D0theo(0:numZ,0:numN),Dlj(0:numl,0:numJ),
+ D1(0:numl),Ditheo(0:numZ,0:numN)
real D0theo,Dlj,D1,Ditheo

c
c ***** subroutine radwidththeory *****
c
common /radwidththeoryr/ gamgamth(0:numZ,0:numN,0:numl),
+ swaveth(0:numZ,0:numN)
real gamgamth,swaveth

c
c ***** subroutine phdensitytable *****
c
common /phdensitytablei/ phexist2(0:numZ,0:numN,0:numexc,
+ 0:numexc,0:numexc,0:numexc),
+ phexist1(0:numZ,0:numN,0:numexc,
+ 0:numexc)
common /phdensitytablei/ nenphdens,Nphconf1,Nphconf2,
+ ppitable(numconf),hpitable(numconf),
+ pntable(numconf),hnutable(numconf),
+ pptable(numconf),hhtable(numconf)
common /phdensitytable/ Ephdensmax,phtable2(0:numZph,0:numNph,
+ 0:numexc,0:numexc,0:numexc,0:numexc,0:numdens),
+ phtable1(0:numZph,0:numNph,0:numexc,
+ 0:numexc,0:numdens)
logical phexist2,phexist1
integer nenphdens,Nphconf2,Nphconf1,ppitable,hpitable,
+ pntable,hnutable,pptable,hhtable
real Ephdensmax,phtable2,phtable1

c
c ***** subroutine weakcoupling *****
c
common /weakcouplingi/ pcore(0:numZ,0:numN,0:numlev2)
common /weakcouplingr/ jcore(0:numZ,0:numN,0:numlev2)
integer pcore
real jcore

c
c ***** subroutine sumrules *****
c
common /sumrulesr/ Egrcoll(0:3,2),Ggrcoll(0:3,2),betagr(0:3,2)
real Egrcoll,Ggrcoll,betagr

c
c ***** subroutine decaydata *****
c
common /decaydatai/ Td(-1:numZ,0:numN,-1:numisom,5),
+ rtyp(-1:numZ,0:numN,-1:numisom)
common /decaydatar/ minutesec,hoursec,daysec,yearsec,
+ Thalf(-1:numZ,0:numN,-1:numisom),
+ lambda(-1:numZ,0:numN,-1:numisom)
integer Td,rtyp
real minutesec,hoursec,daysec,yearsec,Thalf,lambda

c
c ***** subroutine kalbachsep *****
c
common /kalbachsepr/ Smyers(numpar)

```

```

      real Smyers
c
c ***** subroutine grid *****
c
      common /gridc/ ecisstatus
      common /gridi/ maxen,ebegin(0:numpar),eendmax(0:numpar),numincrow,
+ Nrescue(nummt,-1:numisom)
      common /gridr/ egrid(0:numen),deltaE(0:numen),Etop(0:numen),
+ Ebottom(0:numen),coullimit(0:numpar),Einc,E1v,
+ translimit,angle(0:numang),cosangmin(0:numang),
+ cosangmax(0:numang),sinangmin(0:numang),
+ sinangmax(0:numang),dcosang(0:numang),
+ anglecont(0:numangcont),angcontmin(0:numangcont),
+ angcontmax(0:numangcont),
+ cosangcontmin(0:2*numangcont+1),
+ cosangcontmax(0:2*numangcont+1),
+ sinangcontmin(0:2*numangcont+1),
+ sinangcontmax(0:2*numangcont+1),
+ dcosangcont(0:2*numangcont+1),T9(numT),
+ Erescue(nummt,-1:numisom,numen6),
+ frescue(nummt,-1:numisom,numen6)
      character*7 ecisstatus
      integer maxen,ebegin,eendmax,numincrow,Nrescue
      real egrid,deltaE,Etop,Ebottom,coullimit,Einc,E1v,
+ translimit,angle,cosangmin,cosangmax,sinangmin,
+ sinangmax,dcosang,anglecont,angcontmin,angcontmax,
+ cosangcontmin,cosangcontmax,sinangcontmin,
+ sinangcontmax,dcosangcont,T9,Erescue,frescue
c
c ***** subroutine energies *****
c
      common /energiesl/ flagwidth,flagurr,flagcompang,flagpreeq,
+ flaggiant,flagmulpre,flagadd,flagaddel,
+ mulpreZN(0:numZ,0:numN)
      common /energiesi/ eendhigh,eend(0:numpar),nendisc(0:numpar),
+ nbins,nin0
      common /energiesr/ Einc0,enincm,wavenum,Ettotal,speceps,
+ eoutdis(0:numpar,0:numlev2)
      logical flagwidth,flagurr,flagcompang,flagpreeq,flaggiant,
+ flagmulpre,flagadd,flagaddel,mulpreZN
      integer eendhigh,eend,nendisc,nbins,nin0
      real Einc0,enincm,wavenum,Ettotal,speceps,eoutdis
c
c ***** subroutine inverse *****
c
      common /inversec/ transfile,csfile
      character*13 transfile,csfile
c
c ***** subroutine inverseecis *****
c
      common /inverseecisl/ legendre,flaginvecis,flagecisinp
      common /inverseecisc/ ecis1,ecis2,title,tarparity,Plevel(numlev2)
      common /inverseecisi/ Nband,ncoll,npp,iterm,idvib(numlev2),nrad,
+ iph(numlev2),Kmag(numlev2),iband(numlev2),
+ Jband(numlev2),Nrotbeta,iqm,iqmax,njmax
      common /inverseecisr/ hint,rmatch,anginc,angend,angbeg,
+ Elevel(numlev2),tarspin,Jlevel(numlev2),
+ vibbeta(numlev2),rotbeta(numrot),efer,
+ w2disp,d3disp,d2disp,spin,prodZ
      common /inverseecisd/ projmass,resmass
      logical legendre,flaginvecis,flagecisinp
      character*1 tarparity,Plevel
      character*50 ecis1,ecis2
      character*72 title
      integer Nband,ncoll,npp,iterm,idvib,nrad,iph,Kmag,iband,
+ Jband,Nrotbeta,iqm,iqmax,njmax
      real hint,rmatch,anginc,angend,angbeg,Elevel,tarspin,
+ Jlevel,vibbeta,rotbeta,efer,w2disp,d3disp,d2disp,
+ spin,prodZ

```

```

double precision projmass,resmass
C
C ***** subroutine ompadjust *****
C
common /ompadjustr/ Fv1,Fv2,Fv3,Fv4,Frv,Fav,Fw1,Fw2,Fw3,Fw4,Frw,
+ Faw,Frvd,Favd,Fd1,Fd2,Fd3,Frwd,Fawd,Fvso1,
+ Fvso2,Frvso,Favso,Fwso1,Fwso2,Frwso,Fawso,Frc
real Fv1,Fv2,Fv3,Fv4,Frv,Fav,Fw1,Fw2,Fw3,Fw4,Frw,Faw,Frwd,Favd,
+ Fd1,Fd2,Fd3,Frwd,Fawd,Fvso1,Fvso2,Frvso,Favso,Fwso1,Fwso2,
+ Frwso,Fawso,Frc
C
C ***** subroutine opticaln *****
C
common /opticalnr/ v,rv,av,vd,rvd,avd,w,rw,aw,wd,rwd,awd,vso,rvso,
+ avso,wso,rwso,awso,rc
real v,rv,av,vd,rvd,avd,w,rw,aw,wd,rwd,awd,vso,rvso,avso,wso,rwso,
+ awso,rc
C
C ***** subroutine inverseread *****
C
common /inversereadi/ lmax(0:numpar,0:numen)
common /inversereadr/ xstot(0:numpar,0:numen),
+ xsreac(0:numpar,0:numen),
+ xsopt(0:numpar,0:numen),
+ xselas(0:numpar,0:numen),
+ Tj1(0:numpar,0:numen,-1:1,0:numl),
+ Tl(0:numpar,0:numen,0:numl)
integer lmax
real xstot,xsreac,xsopt,xselas,Tj1,Tl
C
C ***** subroutine inversenorm *****
C
common /inversenormr/ threshnorm(0:numpar)
real threshnorm
C
C ***** subroutine preeqinit *****
C
common /preeqiniti/ maxexc,maxpar,maxJph
common /preeqinitr/ Efermi,nfac(0:numexc),
+ ncomb(0:numexc,0:numexc),RnJsum(0:numexc),
+ RnJ(0:numexc,0:numJ),
+ Apauli(-1:numparx+1,-1:numparx+1),
+ Apauli2(-1:numparx+1,-1:numparx+1,-1:numparx+1,-1:numparx+1),
+ Rblann(2,2,numparx)
integer maxexc,maxpar,maxJph
real Efermi,nfac,ncomb,RnJsum,RnJ,Apauli,Apauli2,Rblann
C
C ***** subroutine bonetti *****
C
common /bonettir/ wvol(2,-200:10*numen)
real wvol
C
C ***** subroutine excitoninit *****
C
common /excitoninitr/ wfac(0:numpar),Qfactor(0:numpar,0:numparx)
real wfac,Qfactor
C
C ***** subroutine compoundinit *****
C
common /compoundiniti/ spin2(0:numpar),wpower,nmold,ngoep,ngoes,
+ ngoet
common /compoundinitr/ xmold(nummold),wmold(nummold),
+ xgoep(numgoe),wgoep(numgoe),xgoes(numgoe),
+ wgoes(numgoe),xgoet(numgoe),wgoet(numgoe),
+ logfact(numfact)
integer spin2,wpower,nmold,ngoep,ngoes,ngoet
real xmold,wmold,xgoep,wgoep,xgoes,wgoes,xgoet,wgoet,logfact
C
C ***** subroutine astroinit *****

```

```

c
common /astroiniti/ maxZastro,maxNastro
common /astroinitd/ xsastro(0:numZastro,0:numNastro,0:numenin),
+ xsastroex(0:numZastro,0:numNastro,0:numenin,0:numlev),
+ rateastro(0:numZastro,0:numNastro,numT),
+ rateastroex(0:numZastro,0:numNastro,numT,0:numlev),
+ rateastroracap(numT),
+ macsastro(0:numZastro,0:numNastro,numT),
+ macsastroex(0:numZastro,0:numNastro,numT,0:numlev),
+ macsastroracap(numT),
+ rateastrofis(numT),macsastrofis(numT),
+ xsastrofis(0:numenin),partf(numT)
integer maxZastro,maxNastro
double precision xsastro,rateastro,rateastroracap,macsastro,
+ xsastroex,rateastroex,macsastroex,
+ macsastroracap,rateastrofis,macsastrofis,
+ xsastrofis,partf

c
c ***** subroutine incidentread *****
c
common /incidentreadc/ dorigin(0:numpar,0:numlev2)
common /incidentreadi/ lmaxinc
common /incidentreadr/ xstotinc,xsreacinc,xsoptinc,xselasinc,
+ Tjline(-1:1,0:numl),Tline(0:numl),
+ dleg(0:numpar,0:numlev2,0:3*numl),
+ directad(0:numpar,0:numlev2,0:numang),
+ ruth(0:numang),elasni(0:numang),xscoupled,
+ xsdirdisc(0:numpar,0:numlev2),
+ xsdirdiscot(0:numpar),xsdirdiscsum
character*6 dorigin
integer lmaxinc
real xstotinc,xsreacinc,xsoptinc,xselasinc,Tjline,Tline,
+ dleg,directad,ruth,xscoupled,xsdirdisc,xsdirdiscot,
+ xsdirdiscsum,elasni

c
c ***** subroutine spr *****
c
common /spr/ Sstrength(0:numl),Rprime
real Sstrength,Rprime

c
c ***** subroutine exgrid *****
c
common /exgridi/ maxex(0:numZ,0:numN),nexmax(0:numpar),
+ maxJ(0:numZ,0:numN,0:numex)
common /exgridr/ Exmax0(0:numZ,0:numN),Exmax(0:numZ,0:numN),
+ deltaEx(0:numZ,0:numN,0:numex),
+ Ex(0:numZ,0:numN,0:numex+1)
common /exgridd/ Ethresh(0:numZ,0:numN,0:numlev),
+ Qres(0:numZ,0:numN,0:numlev),
+ rhogrid(0:numZ,0:numN,0:numex,0:numJ,-1:1)
integer maxex,nexmax,maxJ
real Exmax0,Exmax,deltaEx,Ex
double precision Ethresh,Qres,rhogrid

c
c ***** subroutine recoilinit *****
c
common /recoiliniti/ iejlab(0:numpar),irecinit
common /recoilinitr/
+ ddxrec(0:numZ,0:numN,0:numex,0:numenrec,0:numangrec),
+ ddxrectot(0:numZ,0:numN,0:numex),
+ specrecoil(0:numZ,0:numN,0:numex),
+ recoilint(0:numZ,0:numN),
+ ddxejlab(0:numpar,0:numen2,0:numangcont),
+ xsejlab(0:numpar,0:numen2),xsejlabint(0:numpar),
+ angrecmin(0:numangrec),
+ angrecmax(0:numangrec),cosrecmin(0:2*numangrec+1),
+ cosrecmax(0:2*numangrec+1),dcosangrec(0:2*numangrec+1),
+ Eejlab(0:numpar,0:numen2),
+ Eejlabmin(0:numpar,0:numen2),

```

```

+ Eejlabmax(0:numpar,0:numen2),
+ dEejlab(0:numpar,0:numen2),
+ areaejlab(0:numpar,0:numen2,0:2*numangcont+1),
+ Erecinit,Erec(0:numZ,0:numN,0:numenrec)
common /recoilinitrB/
+ Erecmin(0:numZ,0:numN,0:numenrec),
+ Erecmax(0:numZ,0:numN,0:numenrec),
+ areareclab(0:numZ,0:numN,0:numenrec,0:2*numangrec+1),
+ angcm
integer iejlab,irecinit
real ddxrec,ddxrectot,specrecoil,recoilint,ddxejlab,xsejlab,
+ xsejlabint,angrecmin,angrecmax,cosrecmin,
+ cosrecmax,dcosangrec,Eejlab,Eejlabmin,Eejlabmax,dEejlab,
+ areaejlab,Erecinit,Erec,Erecmin,Erecmax,areareclab,angcm
c
c ***** subroutine directread *****
c
common /directreadr/ xsgrcoll(0:numpar,0:3,2),
+ grcollad(0:numpar,0:3,2,0:numangcont),
+ xscollconttot(0:numpar)
real xsgrcoll,grcollad,xscollconttot
c
c ***** subroutine giant *****
c
common /giantr/ eoutgr(0:numpar,0:3,2),
+ xsgrstate(0:numpar,0:3,2,0:numen),
+ xsgr(0:numpar,0:numen),
+ xsgrad(0:numpar,0:numen,0:numangcont),
+ xsgrtot(0:numpar),xsgrsum,
+ xscollcont(0:numpar,0:numen),
+ xscollcontJP(0:numpar,0:numJph,-1:1,0:numen),
+ collcontad(0:numpar,0:numen,0:numangcont)
real eoutgr,xsgrstate,xsgr,xsgrad,xsgrtot,xsgrsum,xscollcont,
+ collcontad,xscollcontJP
c
c ***** subroutine racap *****
c
common /racapi/ racopt,nlevracap(0:numZ,0:numN),inspect,
+ nlevexpracap
common /racapd/
cAK + chglpos(numdensracap),chglneg(numdensracap),
+ chglposj(numdensracap,0:numJ),
+ chglnegj(numdensracap,0:numJ),
+ phdenstot(0:numZ,0:numN,0:numdens),
+ phdensjp(0:numZ,0:numN,0:numdens,0:numJph,-1:1)
common /racapr/ edensphjp(0:numZ,0:numN,0:numdens),
+ jlmracap2(numjlm),vncap2,rvncap2,avncap2,xsracape,
+ xsracap(numenin),xsracapEM(numenin,0:1,numgam),
+ xsracappopex(0:numex),
+ xsracappop(0:numex,0:numJ,-1:1),
+ xsracapedisc,xsracapecont,
+ spectfac(0:numZ,0:numN,0:numex)
integer racopt,nlevracap,inspect,nlevexpracap
cAK double precision chglpos,chglneg,chglposj,chglnegj
double precision chglposj,chglnegj
double precision phdenstot,phdensjp
real edensphjp,
+ jlmracap2,vncap2,rvncap2,avncap2,xsracape,
+ xsracap,xsracapEM,xsracappopex,xsracappop,
+ xsracapedisc,xsracapecont,spectfac
c
c ***** subroutine preeq *****
c
common /preeqi/ p0,h0,ppi0,hpi0,pnu0,hnu0
common /preeqr/ Esurf,xsflux
integer p0,h0,ppi0,hpi0,pnu0,hnu0
real Esurf,xsflux
c
c ***** subroutine exciton *****

```



```

c
common /excitonr/ Ecomp,xsstep(0:numpar,0:numparx,0:numen),
+ xspreeq(0:numpar,0:numen),
+ xspreeqJP(0:numpar,0:numen,0:numJph,-1:1)
real Ecomp,xsstep,xspreeq,xspreeqJP
c
c ***** subroutine emissionrate *****
c
common /emissionrater/
+ wemistot(0:numparx,0:numparx),
+ wemispart(0:numpar,0:numparx,0:numparx),
+ wemission(0:numpar,0:numparx,0:numparx,0:numen)
real wemistot,wemispart,wemission
c
c ***** subroutine lifetime *****
c
common /lifetimer/ depletion(0:numparx,0:numparx),
+ tauexc(0:numparx,0:numparx)
real depletion,tauexc
c
c ***** subroutine matrix *****
c
common /matrixr/ M2,M2pipi,M2nunu,M2pinu,M2nupi,Wompfac(0:2)
real M2,M2pipi,M2nunu,M2pinu,M2nupi,Wompfac
c
c ***** subroutine exciton2 *****
c
common /exciton2r/ xsstep2(0:numpar,0:numparx,0:numparx,0:numen)
real xsstep2
c
c ***** subroutine exchange2 *****
c
common /exchange2r/
+ tauexc2(0:numparx,0:numparx,0:numparx,0:numparx),
+ Lexc(0:numparx,0:numparx,0:numparx,0:numparx),
+ Gpiplus(0:numparx,0:numparx,0:numparx,0:numparx),
+ Gnuplus(0:numparx,0:numparx,0:numparx,0:numparx),
+ Gpinu(0:numparx,0:numparx,0:numparx,0:numparx),
+ Gnupi(0:numparx,0:numparx,0:numparx,0:numparx)
real tauexc2,Lexc,Gpiplus,Gnuplus,Gpinu,Gnupi
c
c ***** subroutine emissionrate2 *****
c
common /emissionrate2r/
+ wemistot2(0:numparx,0:numparx,0:numparx,0:numparx),
+ wemispart2(0:numpar,0:numparx,0:numparx,0:numparx,0:numparx),
+ wemission2(0:numpar,0:numparx,0:numparx,0:numparx,0:numparx,
+ 0:numen)
real wemistot2,wemispart2,wemission2
c
c ***** subroutine lifetime2 *****
c
common /lifetime2r/ PP2(0:numparx,0:numparx,0:numparx,0:numparx),
+ Spre(0:numparx,0:numparx,0:numparx,0:numparx)
real PP2,Spre
c
c ***** subroutine msdinit *****
c
common /msdiniti/ maxmsd,maxJmsd,msdbins2
common /msdinitr/ dEmsd,Emsd(0:numenmsd)
integer maxmsd,maxJmsd,msdbins2
real dEmsd,Emsd
c
c ***** subroutine interangle *****
c
common /interanglei/
+ nangleint(0:numangcont,0:numangcont,0:numangcont)
integer nangleint
c

```

```

c ***** subroutine dwbaecis *****
c
c      common /dwbaecisr/ betamsd,Emsdin,Emsdout,Exmsd
c      real betamsd,Emsdin,Emsdout,Exmsd
c
c ***** subroutine dwbaread *****
c
c      common /dwbareadr/ xsdwin(0:numenmsd,0:numenmsd,0:numJmsd,0:2),
c      + xsdw(0:numenmsd,0:numenmsd,0:numJmsd,0:numangcont,0:2)
c      real xsdwin,xsdw
c
c ***** subroutine onecontinuumA *****
c
c      common /onecontinuumAr/
c      + xscont1(0:numpar,0:numpar,0:numenmsd,0:numenmsd),
c      + xscontad1(0:numpar,0:numpar,0:numenmsd,0:numenmsd,0:numangcont)
c      real xscont1,xscontad1
c
c ***** subroutine onestepA *****
c
c      common /onestepAr/ msdstep1(0:numpar,0:numen),
c      + msdstepad1(0:numpar,0:numen,0:numangcont)
c      real msdstep1,msdstepad1
c
c ***** subroutine onestepB *****
c
c      common /onestepBr/ msdstep(0:numpar,nummsd,0:numen),
c      + msdstepad(0:numpar,nummsd,0:numen,0:numangcont)
c      real msdstep,msdstepad
c
c ***** subroutine onecontinuumB *****
c
c      common /onecontinuumBr/
c      + xscont(0:numpar,0:numpar,0:numenmsd,0:numenmsd),
c      + xscontad(0:numpar,0:numpar,0:numenmsd,0:numenmsd,0:numangcont),
c      + msdstep0(0:numpar,nummsd,0:numenmsd),
c      + msdstepad0(0:numpar,nummsd,0:numenmsd,0:numangcont)
c      real xscont,xscontad,msdstep0,msdstepad0
c
c ***** subroutine msdtotal *****
c
c      common /msdtotalr/ msdall,msdsum(0:numpar),
c      + msdtot(0:numpar,0:numen),
c      + msdstepint(0:numpar,nummsd),
c      + msdtotintad(0:numpar,0:numangcont),
c      + msdtotad(0:numpar,0:numen,0:numangcont),
c      + msdstepintad(0:numpar,nummsd,0:numangcont)
c      real msdall,msdsum,msdtot,msdstepint,msdtotintad,msdtotad,
c      + msdstepintad
c
c ***** subroutine msdplumsc *****
c
c      common /msdplumscr/ xspreeqad(0:numpar,0:numen2,0:numangcont)
c      real xspreeqad
c
c ***** subroutine preeqcomplex *****
c
c      common /preeqcomplexr/ xspreeqps(0:numpar,0:numen),
c      + xspreeqki(0:numpar,0:numen),
c      + xspreeqbu(0:numpar,0:numen)
c      real xspreeqps,xspreeqki,xspreeqbu
c
c ***** subroutine breakup *****
c
c      common /breakupr/ Sab,Deff,Ecent,Ca
c      real Sab,Deff,Ecent,Ca
c
c ***** subroutine breakupAVR *****
c

```

```

common /breakupAVRl/ breakupexist
common /breakupAVRr/ xsEB(0:numpar),xsBF(0:numpar),
+ xsBUnuc(0:numpar),
+ xsBFnuc(0:numZ,0:numN),
+ xspopnucT(0:numZ,0:numN),xsmassprodT(0:numA),
+ xsisoBU(0:numZ,0:numN,0:numex),
+ fxisoBU(numenlow,0:numZ,0:numN,0:numlev)
logical breakupexist
real xsEB,xsBF,xsBUnuc,xsBFnuc,xspopnucT,xsmassprodT,xsisoBU,
+ fxisoBU
c
c ***** subroutine NPxsratios *****
c
common /BUsratios/ ebubin,
+ ENHratio(0:numpar,0:numZ,0:numN,0:numenout)
real ebubin,ENHratio
c
c ***** subroutine preeqcorrect *****
c
common /preeqcorrectr/ xspreeqdisc(0:numpar,0:numlev2),
+ xspreeqdiscTot(0:numpar),xspreeqdiscsum
real xspreeqdisc,xspreeqdiscTot,xspreeqdiscsum
c
c ***** subroutine preeqttotal *****
c
common /preeqttotalr/ xssteptot(0:numpar,0:numparx),
+ xspreeqtot(0:numpar),xspreeqtotps(0:numpar),
+ xspreeqtotki(0:numpar),
+ xspreeqtotbu(0:numpar),xspreeqsum,preeqnorm
real xssteptot,xspreeqtot,xspreeqtotps,xspreeqtotki,xspreeqtotbu,
+ xspreeqsum,preeqnorm
c
c ***** subroutine population *****
c
common /populationr/ preeqpopex(0:numZ,0:numN,0:numex),
+ preeqpop(0:numZ,0:numN,0:numex,0:numJ,-1:1),
+ xspopph(0:numZph,0:numNph,0:numex,0:numparx,0:numparx),
+ xspopph2(0:numZph,0:numNph,0:numex,0:numparx,0:numparx,0:numparx,
+ 0:numparx)
real preeqpopex,preeqpop,xspopph,xspopph2
c
c ***** subroutine compnorm *****
c
common /compnormi/ J2beg,J2end,pardif
common /compnormr/ CNfactor,Crescue(nummt,-1:numisom),
+ CNterm(-1:1,0:numJ)
integer J2beg,J2end,pardif
real CNfactor,Crescue,CNterm
c
c ***** subroutine comptarget *****
c
common /comptargeti/ tnumi,tnumo,nulj(-1:numpar,0:numl,0:numJ),
+ lminU,lmaxU,JminU(0:numl),JmaxU(0:numl),
+ Purrlj(0:numl,0:numJ)
common /comptargetr/ Exinc,Wab,dExinc,xsbinary(-1:numpar),
+ Turrlj(-1:numpar,0:numl,0:numJ),
+ Turrljinc(0:numl,0:numJ),
+ xsbinarylj(-1:numpar,0:numl,0:numJ),
+ cleg(0:numpar,0:numlev,0:3*numl),
+ xscompcont(0:numpar),Fnorm(-1:numpar)
common /comptargetd/ xspop(0:numZ,0:numN,0:numex,0:numJ,-1:1),
+ popdecay(-1:numpar,0:numex,0:numJ,-1:1),
+ partdecay(-1:numpar,-1:1),
+ partdecaytot(-1:numpar),
+ xspopex(0:numZ,0:numN,0:numex),
+ xspopnuc(0:numZ,0:numN),
+ xspopexP(0:numZ,0:numN,0:numex,-1:1),
+ xspopnucP(0:numZ,0:numN,-1:1)
integer tnumi,tnumo,nulj,lminU,lmaxU,JminU,JmaxU,Purrlj

```

```

      real Exinc,Wab,dExinc,xsbinary,Turrlj,Turrljinc,
+ xsbinarylj,cleg,xscompcont,Fnorm
      double precision xspop,popdecay,partdecay,partdecaytot,xspopex,
+ xspopnuc,xspopexP,xspopnucP
c
c ***** subroutine densprepare *****
c
      common /denspreparei/ lmaxhf(0:numpar,0:numex),nbintfis(numbar)
      common /denspreparer/ Tgam(0:numex,0:numgam,0:1),
+ Tjlnex(0:numpar,0:numex,-1:1,0:numl),
+ Tlnex(0:numpar,0:numex,0:numl),
+ eintfis(numbinfis,numbar)
      common /densprepared/ rho0(0:numpar,0:numex,0:numJ,-1:1),
+ rhofis(numbinfis,0:numJ,-1:1,numbar)
      integer lmaxhf,nbintfis
      real Tgam,Tjlnex,Tlnex,eintfis
      double precision rho0,rhofis
c
c ***** subroutine tfission *****
c
      common /tfissiond/ tfisdown(0:numJ,-1:1),tfis(0:numJ,-1:1),
+ gamfis(0:numJ,-1:1),taufis(0:numJ,-1:1),
+ denfis(0:numJ,-1:1),
+ tfisA(0:numJ,-1:1,0:numhill),
+ rhofisA(0:numJ,-1:1,0:numhill),
+ tfisup(0:numJ,-1:1)
      double precision tfisdown,tfis,gamfis,taufis,denfis,tfisA,
+ rhofisA,tfisup
c
c ***** subroutine compprepare *****
c
      common /comppreparei/ tnum,tnuminc
      common /compprepared/ denomhf,feed,transjl(0:5,numtrans),fiswidth,
+ enumhf(0:numpar,0:numex,0:numJ,-1:1)
      integer tnum,tnuminc
      double precision denomhf,feed,transjl,fiswidth,enumhf
c
c ***** subroutine widthprepare *****
c
      common /widthpreparer/ freedom(numtrans),prodwidth(nummold),
+ sumhrtw,vhrtw(numtrans),whrtw(numtrans),
+ agoe1(numgoe,numgoe,numgoe),
+ agoe2(numgoe,numgoe,numgoe),
+ agoe3(numgoe,numgoe,numgoe),
+ agoe4(numgoe,numgoe,numgoe),
+ agoe5(numgoe,numgoe,numgoe),
+ agoe6(numgoe,numgoe,numgoe),
+ agoe7(numgoe,numgoe,numgoe),
+ agoe8(numgoe,numgoe,numgoe),sgoe1,sgoe2,
+ sgoe3,sgoe4,sgoe5
      common /widthprepared/ tjlav(numtrans)
      real freedom,prodwidth,sumhrtw,vhrtw,whrtw,agoe1,
+ agoe2,agoe3,agoe4,agoe5,agoe6,agoe7,agoe8,sgoe1,
+ sgoe2,sgoe3,sgoe4,sgoe5
      double precision tjlav
c
c ***** subroutine astroprepare *****
c
      common /astroprepared/ Tastrotot,rhoastrotot,
+ Tastroinc(0:1,0:numex,0:numJ,-1:1),
+ Tastroout(0:1,0:numpar,0:numex,0:numJ,-1:1)
      double precision Tastrotot,rhoastrotot,Tastroinc,Tastroout
c
c ***** subroutine raynalcomp *****
c
      common /raynalcomp/ pcomp(numcomp)
      common /raynalcomp/ typecomp(0:numcomp),nsp1,nsp2,ncont
      common /raynalcomp/ bz1,elevelcomp(0:numcomp),jcomp(numcomp),
+ spincomp(numcomp),prodZcomp(numcomp),

```

```

+ aldcomp(0:numpar),Umcomp(0:numpar),
+ tempcomp(0:numpar),E0comp(0:numpar),tgo,
+ Excomp(0:numpar)
common /raynalcompd/ ejeccomp(numcomp),masscomp(numcomp)
character*1 pcomp
integer typecomp,nsp1,nsp2,ncont
real bz1,elevelcomp,jcomp,spincomp,prodZcomp,aldcomp,
+ Umcomp,tempcomp,E0comp,tgo,Excomp
double precision ejeccomp,masscomp
c
c ***** subroutine urr *****
c
common /urrl/ flagurrendf,urrexist(-1:11,0:numl)
common /urrr/ Rprime0,urwidth(-1:numpar,0:numl,0:numJ),
+ spot(0:numl),strengthlj(0:numl,0:numJ),
+ strengthl(0:numl),sigurrs(0:numl,0:numJ),
+ sigurrf(0:numl,0:numJ),sigurrc(0:numl,0:numJ),
+ xsurrN(4),xsurrT(4)
logical flagurrendf,urrexist
real Rprime0,urwidth,spot,strengthlj,strengthl,sigurrs,
+ sigurrf,sigurrc,xsurrN,xsurrT
c
c ***** subroutine binary *****
c
common /binaryr/ xspopex0(0:numpar,0:numlev),
+ sfactor(0:numZ,0:numN,0:numex,0:numJ,-1:1),
+ xspopdir(0:numZ,0:numN),xscompdiscatot(0:numpar),
+ Eaveragebin(0:numpar),xsdisc(0:numpar,0:numlev),
+ xscompdisc(0:numpar,0:numlev),
+ xsdiscatot(0:numpar),xsdircont(0:numpar),
+ xsdirect(0:numpar),xsconttot(0:numpar),
+ xscompound(0:numpar),xscompel,xscompel6(numen6),
+ xsnonel6(numen6),xselaotot,xsnonel,xscomponel,
+ feedbinary(0:numpar,0:numex),binemisum(0:numpar)
real xspopex0,sfactor,xspopdir,xscompdiscatot,Eaveragebin,xsdisc,
+ xscompdisc,xsdiscatot,xsdircont,xsdirect,xsconttot,xscompound,
+ xscompel,xscompel6,xsnonel6,xselaotot,xsnonel,xscomponel,
+ feedbinary,binemisum
c
c ***** subroutine binaryspectra *****
c
common /binaryspectrar/ xscomp(0:numpar,0:numen),
+ xsbinemis(0:numpar,0:numen),
+ xscompad(0:numpar,0:numen,0:numangcont),
+ xsbinemisad(0:numpar,0:numen,0:numangcont)
real xscomp,xsbinemis,xscompad,xsbinemisad
c
c ***** subroutine binemission *****
c
common /binemissionr/ xsemis(0:numpar,0:numen),binnorm(0:numpar),
+ binemis(0:numpar,0:numex,0:numen),
+ contrib(0:numpar,0:numex)
real xsemis,binnorm,binemis,contrib
c
c ***** subroutine angdis *****
c
common /angdisl/ legexist(0:numpar,0:numpar,0:numlev),
+ angexist(0:numpar,0:numpar,0:numlev)
common /angdisr/ compad(0:numpar,0:numlev,0:numang),
+ discad(0:numpar,0:numlev,0:numang),
+ tleg(0:numpar,0:numlev,0:3*numl),
+ tlegnor(0:numpar,0:numlev,0:3*numl),
+ cleg0(0:numpar,0:numlev,0:3*numl)
logical legexist,angexist
real compad,discad,tleg,tlegnor,cleg0
c
c ***** subroutine multiple *****
c
common /multipler/ xsmpeemis(0:numpar,0:numen),

```

```

+ xspartial(0:numpar,0:numex+1),
+ xsbinspec(0:numpar,0:numex+1,0:numen),
+ xsmpe(0:numpar,0:numex+1),
+ mcontrib(0:numpar,0:numex+1,0:numex+1),
+ mpecontrib(0:numpar,0:numex+1,0:numex+1),
+ xsmptot(0:numpar),
+ popexcl(0:numZ,0:numN,0:numex+1),
+ Dmulti(0:numex),
+ feedexcl(0:numZchan,0:numNchan,0:numpar,
+ 0:numex+1,0:numex+1),
+ xspoppreeq(0:numZ,0:numN),
+ xspopcomp(0:numZ,0:numN),Fdir(0:numZ,0:numN),
+ Fpreeq(0:numZ,0:numN),Fcomp(0:numZ,0:numN),
+ xsmppreeq(0:numpar,0:numen),
+ xsmppreeqad(0:numpar,0:numen,0:numangcont),
+ fisfeedex(0:numZ,0:numN,0:numex+1),
+ fisfeedJP(0:numZ,0:numN,0:numex+1,0:numJ,-1:1),
+ xsfeed(0:numZ-2,0:numN-2,-1:numpar),
+ Eaveragemul(0:numZ-2,0:numN-2,-1:numpar),
+ xsngn(-1:numpar),xsngnsum,
+ xsngnspec(0:numpar,0:numen)
  real xsmpeemis,xspartial,xsbinspec,xsmpe,mcontrib,mpecontrib,
+ xsmptot,Dmulti,popexcl,feedexcl,xspoppreeq,xspopcomp,Fdir,
+ Fpreeq,Fcomp,xsmppreeq,xsmppreeqad,fisfeedex,fisfeedJP,xsfeed,
+ Eaveragemul,xsngn,xsngnsum,xsngnspec
c
c ***** subroutine excitation *****
c
c   common /excitationr/ xsinitpop
c   real xsinitpop
c
c ***** subroutine cascade *****
c
c   common /cascader/ xsgamdis(0:numZ,0:numN,0:numlev,0:numlev),
+ xsgamdistot(0:numZ,0:numN)
c   real xsgamdis,xsgamdistot
c
c ***** subroutine compemission *****
c
c   common /compemissionr/ compspect(numen),preeqspect(numen)
c   real compspect,preeqspect
c
c ***** subroutine cm2lab *****
c
c   common /cm2labr/ Eejlab11,Eejlab12,Eejlab21,Eejlab22,
+ cosejlab11,cosejlab12,cosejlab21,cosejlab22,
+ sinejlab11,sinejlab12,sinejlab21,sinejlab22,
+ Eejcm1,Eejcm2,vcm,vejcm1,vejcm2,
+ cosejcm1,cosejcm2,sinejcm1,sinejcm2,
+ Ereclab11,Ereclab12,Ereclab21,Ereclab22,
+ cosreclab11,cosreclab12,cosreclab21,cosreclab22,
+ sinreclab11,sinreclab12,sinreclab21,sinreclab22,
+ vrecm1,vrecm2,ejectmass,recoilmass
c   real Eejlab11,Eejlab12,Eejlab21,Eejlab22,
+ cosejlab11,cosejlab12,cosejlab21,cosejlab22,
+ sinejlab11,sinejlab12,sinejlab21,sinejlab22,
+ Eejcm1,Eejcm2,vcm,vejcm1,vejcm2,
+ cosejcm1,cosejcm2,sinejcm1,sinejcm2,
+ Ereclab11,Ereclab12,Ereclab21,Ereclab22,
+ cosreclab11,cosreclab12,cosreclab21,cosreclab22,
+ sinreclab11,sinreclab12,sinreclab21,sinreclab22,
+ vrecm1,vrecm2,ejectmass,recoilmass
c
c ***** subroutine channels *****
c
c   common /channelsl/ chanopen(0:numin,0:numip,0:numid,0:numit,
+ 0:numih,0:numia),idnumfull
c   common /channelsc/ reacstring(0:numchantot),
+ fisstring(0:numchantot)

```

```

common /channelsi/ idnum,idchannel(0:numchantot),opennum
common /channelsr/ xsparcheck(0:numpar),
+ xsspeccheck(0:numpar,0:numen),channelsum,xsabs,
+ xsexcl(0:numchantot,0:numex+1),
+ gamexcl(0:numchantot,0:numex+1),
+ xschannel(0:numchantot),
+ yieldchannel(0:numchantot),
+ xsgamchannel(0:numchantot),
+ xsfishchannel(0:numchantot),
+ xschancecheck(0:numchantot),
+ xsfishchancecheck(0:numchantot),
+ xsratio(0:numchantot),
+ xschaniso(0:numchantot,0:numlev),
+ exclbranch(0:numchantot,0:numlev),
+ xsgamdischan(0:numchantot,0:numlev,0:numlev),
+ Eavchannel(0:numchantot,0:numpar),
+ xschannelssp(0:numchantot,0:numpar,0:numen),
+ xsfishchannelssp(0:numchantot,0:numpar,0:numen),
+ specemis(0:numen),Especsum(0:numchantot),
+ gmult(0:numchantot)
common /channelsd/ Ethrexcl(0:numchantot,0:numlev),
+ Qexcl(0:numchantot,0:numlev)
logical chanopen,idnumfull
character*18 reacstring,fisstring
integer idnum,idchannel,opennum
real xsparcheck,xsspeccheck,channelsum,xsabs,xsexcl,
+ gamexcl,xschannel,xsgamchannel,xsfishchannel,
+ xschancecheck,xsfishchancecheck,xsratio,xschaniso,
+ exclbranch,xsgamdischan,Eavchannel,xschannelssp,
+ xsfishchannelssp,specemis,Especsum,gmult,
+ yieldchannel
double precision Ethrexcl,Qexcl
c
c ***** subroutine totalxs *****
c
common /totalxsr/ xsexclusive(0:numpar),xsexclcont(0:numpar),
+ xsparticle(0:numpar),multiplicity(0:numpar),
+ Eaverage(0:numpar),xsfristot,xsfristot0
real xsexclusive,xsexclcont,xsparticle,multiplicity,Eaverage,
+ xsfristot,xsfristot0
c
c ***** subroutine spectra *****
c
common /spectral/ spexist1(0:numpar),spexist2(0:numpar),
+ ddxexist1(0:numpar),ddxexist2(0:numpar),
+ ddxexist3(0:numpar),ddxexist4(0:numpar)
common /spectrai/ eendout(0:numpar)
common /spectrar/ espec(0:numpar,0:numen2),
+ xsdiscout(0:numpar,0:numen2),
+ xspreeqout(0:numpar,0:numen2),
+ xspreeqpsout(0:numpar,0:numen2),
+ xspreeqkiout(0:numpar,0:numen2),
+ xspreeqbuout(0:numpar,0:numen2),
+ xsmpreeqout(0:numpar,0:numen2),
+ xscompout(0:numpar,0:numen2),
+ xsdiscoutad(0:numpar,0:numen2,0:numangcont),
+ xspreeqoutad(0:numpar,0:numen2,0:numangcont),
+ xsmpreeqoutad(0:numpar,0:numen2,0:numangcont),
+ xscompoutad(0:numpar,0:numen2,0:numangcont),
+ xssumout(0:numpar,0:numen2),
+ preeqratio(0:numpar,0:numen2),
+ buratio(0:numpar,0:numen2),
+ xssumoutad(0:numpar,0:numen2,0:numangcont)
logical spexist1,spexist2,ddxexist1,ddxexist2,ddxexist3,ddxexist4
integer eendout
real espec,xsdiscout,xspreeqout,xspreeqpsout,xspreeqkiout,
+ xspreeqbuout,xsmpreeqout,xscompout,xsdiscoutad,
+ xspreeqoutad,xsmpreeqoutad,xscompoutad,xssumout,
+ preeqratio,buratio,xssumoutad

```

```

C
C ***** subroutine massdis *****
C
    common /massdisl/ fpexist(numelem,numneu),fpaexist(nummass),
+   nubarexist(0:numpar)
    common /massdisi/ Aff,Zff,NEpfn
    common /massdisr/ excfis,xsZApri(numelem,numneu),
+   xsZApri(numelem,numneu),
+   xsApri(nummass),xsApri(nummass),
+   yieldApri(nummass),yieldApri(nummass),
+   nuA(0:numpar,nummass),
+   nuZA(0:numpar,numelem,numneu),
+   EaverageA(0:numpar,nummass),
+   EaverageZA(0:numpar,numelem,numneu),
+   yieldZApri(numelem,numneu),
+   yieldZApri(numelem,numneu),
+   Excff(numelem,numneu),dExcff(numelem,numneu),
+   TKE(numelem,numneu),
+   yieldfpex(numelem,numneu,0:1),
+   fpratio(numelem,numneu,0:1),fpeps,
+   xsfpex(numelem,numneu,0:1),
+   Pdisnu(0:numpar,0:numnu),
+   pfns(0:numpar,0:numpfns),
+   pfnscl(0:numpar,0:numpfns),
+   Eavpfns(0:numpar),maxpfns(0:numpar,0:numpfns),
+   nubar(0:numpar),disa(nummass),disacor(nummass),
+   disaz(nummass,numelem),
+   Epfn(numpfns),dEpfn(numpfns),
+   disazcor(nummass,numelem),xstotpost,
+   xstotpre,yieldtotpost,yieldtotpre,
+   Pdisnuav(0:numpar),Epfnaverage(0:numpar)
    logical fpexist,fpaexist,nubarexist
    integer Aff,Zff,NEpfn
    real excfis,xsZApri,xsZApri,xsApri,xsApri,yieldApri,
+   yieldApri,nuA,nuZA,EaverageA,EaverageZA,yieldZApri,TKE,
+   yieldZApri,Excff,dExcff,yieldfpex,fpratio,fpeps,xsfpex,
+   Pdisnu,pfns,pfnscl,Eavpfns,maxpfns,nubar,disa,disacor,
+   disaz,Epfn,disazcor,xstotpost,xstotpre,yieldtotpost,
+   yieldtotpre,Pdisnuav,Epfnaverage,dEpfn
C
C ***** subroutine brosfy *****
C
    common /brosfyr/ bfsplin(9),hwsplin(9),bf(9),hw(9)
    common /brosfyi/ numtemp
    integer numtemp
    real bfsplin,hwsplin,bf,hw
C
C ***** subroutine neck*****
C
    common /neckr/ cur,c0,totl,di,rest,r1,r2,r3,z1,z2,z3,vtot,rt,rp,
+   rpt,amm,zee,ess,aaa
    real cur,c0,totl,di,rest,r1,r2,r3,z1,z2,z3,vtot,rt,rp,rpt,amm,zee,
+   ess,aaa
C
C ***** subroutine rpevap *****
C
    common /rpevapi/ Arp,Zrp,xspopnuc0(numelem,nummass)
    integer Arp,Zrp
    real xspopnuc0
C
C ***** subroutine residual *****
C
    common /residuali/ maxA
    common /residualr/ xsresprod,xsmassprod(0:numA),
+   xsbranch(0:numZ,0:numN,0:numlev)
    integer maxA
    real xsresprod,xsmassprod,xsbranch
C
C ***** subroutine normalization *****

```



```

c      common /normalizationr/ xstotadjust(numen6),xseladjust(numen6),
+      xsnonadjust(numen6)
+      real xstotadjust,xseladjust,xsnonadjust
c
c ***** subroutine thermal *****
c
c      common /thermalrA/ fxschannel(numenlow,0:numchantot),
+      fxsgamchannel(numenlow,0:numchantot),
+      fxsgamdischan(numenlow,0:numchantot,0:numlev,0:numlev),
+      fxsratio(numenlow,0:numchantot),
+      fxschaniso(numenlow,0:numchantot,0:numlev),
+      fexclbranch(numenlow,0:numchantot,0:numlev),
+      fxsbinary(numenlow,0:numpar),
+      fxspopnuc(numenlow,0:numZ,0:numN),
+      fxspopex(numenlow,0:numZ,0:numN,0:numlev),
+      fxbranch(numenlow,0:numZ,0:numN,0:numlev),
+      fxsexclusive(numenlow,0:numpar),
+      fxsdiscotot(numenlow,0:numpar),
+      fxsexclcont(numenlow,0:numpar),
+      fxsgn(numenlow,-1:numpar),
+      fxsdisc(numenlow,0:numpar,0:numlev),
+      fxsdirdisc(numenlow,0:numpar,0:numlev),
+      fxscmpdisc(numenlow,0:numpar,0:numlev),
+      fxsnonel(numenlow),fxselastot(numenlow),
+      fxstotinc(numenlow),fxscompel(numenlow),
+      fxselasinc(numenlow),fxsreacinc(numenlow)
+      common /thermalrB/ fxscmpnonel(numenlow),fxsdirdiscsum(numenlow),
+      fxsracape(numenlow),
+      fxspreeqsum(numenlow),fnubar(numenlow,0:numpar)
+      real fxschannel,fxsgamchannel,fxsgamdischan,fxsratio,fxschaniso,
+      fexclbranch,fxsbinary,fxspopnuc,fxspopex,fxbranch,
+      fxsexclusive,fxsdiscotot,fxsexclcont,fxsgn,fxsdisc,
+      fxsdirdisc,fxscmpdisc,fxsnonel,fxselastot,fxstotinc,
+      fxscompel,fxselasinc,fxsreacinc,fxscmpnonel,fxsdirdiscsum,
+      fxspreeqsum,fxsracape,fnubar
c
c ***** subroutine residualout *****
c
c      common /residualoutl/ rpexist(0:numZ,0:numN),
+      recexist(0:numZ,0:numN),
+      rpisoexist(0:numZ,0:numN,0:numlev)
+      logical rpexist,rpisoexist,recexist
c
c ***** subroutine fissionout *****
c
c      common /fissionoutl/ fisexist(0:numZ,0:numN)
+      logical fisexist
c
c ***** subroutine channelsout *****
c
c      common /channelsoutl/
+      chanexist(0:numin,0:numip,0:numid,0:numit,0:numih,0:numia),
+      spchanexist(0:numin,0:numip,0:numid,0:numit,0:numih,0:numia),
+      recchanexist(0:numin,0:numip,0:numid,0:numit,0:numih,0:numia),
+      spfischanexist(0:numin,0:numip,0:numid,0:numit,0:numih,0:numia),
+      gamchanexist(0:numin,0:numip,0:numid,0:numit,0:numih,0:numia),
+      chanisoexist(0:numin,0:numip,0:numid,0:numit,0:numih,0:numia,
+      0:numlev),
+      chanfisexist(0:numin,0:numip,0:numid,0:numit,0:numih,0:numia)
+      logical chanexist,spchanexist,recchanexist,spfischanexist,
+      gamchanexist,chanisoexist,chanfisexist
c
c ***** subroutine gamdisout *****
c
c      common /gamdisoutl/ gamexist(0:numZ,0:numN,0:numlev,0:numlev)
+      logical gamexist
c
c ***** subroutine prodres *****

```

```

c
common /prodresl/ prodexist(-1:numZ,-1:numN,-1:numisom)
common /prodresi/ Nenrp(-1:numZ,-1:numN,-1:numisom)
common /prodresr/ Erp(-1:numZ,-1:numN,-1:numisom,numenin),
+ xsrp(-1:numZ,-1:numN,-1:numisom,numenin)
logical prodexist
integer Nenrp
real Erp,xsrp

c
c ***** subroutine prodrates *****
c
common /ratesr/ targetdx,Vtar,Mtar,projnum,heat,
+ prate(-1:numZ,-1:numN,-1:numisom)
real targetdx,Vtar,Mtar,projnum,heat,prate

c
c ***** subroutine prodyield *****
c
common /yieldcalcl/ Yexist(0:numZ,0:numN,-1:numisom)
common /yieldcalci/ Ntime,Tp(0:numZ,0:numN,-1:numisom,5),
+ Tmaxactivity(0:numZ,0:numN,-1:numisom)
common /yieldcalcr/ Ntar0,Tgrid(0:numtime),Tir,Tco,
+ Niso(0:numZ,0:numN,-1:numisom,0:numtime),
+ activity(0:numZ,0:numN,-1:numisom,0:numtime),
+ yield(0:numZ,0:numN,-1:numisom,0:numtime),
+ Nisorel(0:numZ,0:numN,-1:numisom,0:numtime),
+ Nisotot(0:numZ,0:numtime),
+ Tmax(0:numZ,0:numN,-1:numisom)
logical Yexist
integer Ntime,Tp,Tmaxactivity
real Ntar0,Tgrid,Tir,Tco,Niso,activity,yield,
+ Nisorel,Nisotot,Tmax

c
c ***** subroutine endfenergies *****
c
common /endfenergiesi/ nen6
common /endfenergiesd/ e6(numen6)
integer nen6
double precision e6

c
c ***** subroutine endfread *****
c
common /endfreadr/ xstot6(numen6),xsreac6(numen6),xsopt6(numen6),
+ xselash6(numen6),xselas6(numen6),
+ xsnon6(numen6)
real xstot6,xsreac6,xsopt6,xselash6,xselas6,xsnon6

```

Copyright (C) 2023 A.J. Koning, S. Hilaire and S. Goriely