AEAT-1008

NJOY/PERRY-02

Installation of NJOY94.10 at AEA Technology

R J Perry

DRAFT

December 1996

Title	Installation of NJOY94.10 at AEA Technology
Customer	CIRE
Customer reference	96-1064
Confidentiality, copyright and reproduction	Restricted - Commercial This document has been prepared by AEA Technology plc in connection with a contract to supply goods and/or services and is submitted only on the basis of strict confidentiality. The contents must not be disclosed to third parties other than in accordance with the terms of the contract.
File reference	/kanga1/ray/docs/AEAT.1008.fm
Reference number	AEAT- 1008

AEA Technology plc
Winfrith, Dorchester
Dorset DT2 8DH
United Kingdom
Telephone 01305 251888
Facsimile 01305 203030

AEA Technology is the Trading name of
AEA Technology plc

AEA Technology is certified to ISO9001

	Name	Position	Signature	Date
Author	R J Perry	Project Manager		
Checked	C J Dean	Nuclear Data Consultant		
Approved	C A Cooper	Department Manager		

Executive Summary

The NJOY Nuclear Data Processing system is used to process evaluated nuclear data into broad group applications libraries. A new version of the code (NJOY94.10) has been installed and validated at AEA Technology, Winfrith. A number of local amendments have been applied to bring the code to an acceptable condition for use by AEA Technology. These amendments are listed in this paper to allow them to be assessed, by the code author, for possible inclusion in later NJOY releases.

The scheme for interpolating between thermal scattering matrices, used in NJOY94.10, has been replaced by an earlier scheme used in previous versions of NJOY. Validation tests have shown that, for light water and graphite systems, the older scheme gives a better representation of thermal scattering.

This report completes Milestone M1 .

Contents

1 Introduction

Computer codes which model aspects of reactor physics, shielding and criticality require high quality nuclear data. Modern evaluated nuclear data are usually stored in nuclide dependent evaluations in a standard structure called ENDF-6 format. Applications libraries, used by the physics codes, require the evaluated data (e.g.JEF2.2) to be processed into suitable forms, usually group averaged over ranges of the energy of the 'particles' concerned. The NJOY Nuclear Data processing System (1) preforms this function. The NJOY code package is maintained by Dr R MacFarlane of Los Alamos National Laboratory, USA. It is released internationally and feedback from the various user organisations is used to aid the development of the code. These code developments are identified by a system of release and update numbers.The latest version to be installed at AEA Technology, Winfrith is NJOY94.10 signifying that the base version is NJOY94 (assembled in 1994) for which ten further updates have been issued by LANL as a result of testing and used feedback.

This note contains details of the local updates which have been applied to NJOY94.10 to obtain a quality assured version of the program. Previous local updates which had been applied to NJOY-91.91 (2) were studied and any which had not been included by the code author in the NJOY94.10 program were applied.

At Winfrith, NJOY is verified by means of a set of test cases (19 at time of publication) which are all executed after any modification of the code. The first eight are standard test cases and results are supplied with the code release. The remaining test cases were designed to test changes and additions made at Winfrith. These test cases may be modified or new test cases created as is made necessary by modifications to the code. The results from the test cases are compared with those obtained from the previous execution with any differences being explained. A brief description of the current database of test cases is given in Appendix 1.

We attempt to make all improvements to NJOY consistent with the international coding so they can eventually be included in later releases by R MacFarlane. Local updates are labelled with 'idents' of the form upw1, upw2, etc.

The updates which are applied can be distinguished as follows,

- those sent by the code authors to bring the code to their recommended standard e.g. the ten in NJOY94.10

- those suggested by the code authors to make the code operate on particular machine types

- improvements made in the UK but not yet incorporated in the authors coding

- improvements made in the UK to enable the code to operate on local SUN workstations

- extensions made in the UK to link with private codes including additional options to standard modules.

2 Installation of the Standard Version of NJOY94.10

The first step in the installation process was to install NJOY94.10 as supplied by the NEA databank, without any Winfrith enhancements, to ensure that the base code functioned correctly on local machines. The standard test cases were run and the results compared with those supplied by the databank. There were a number of small differences in the results. It was noted, however, that the banner heading on the databank output files indicated that NJOY94.0 had been used rather than NJOY94.10. In an attempt to verify this, NJOY94.0 was installed (i.e. without updates 1-10) and the test cases rerun. The results from the installed version of NJOY94.0 differed slightly from both the NJOY94.10 and the databank results. It is possible that the databank results were produced by some version between 94.0 and 94.10 but the exact version used could not be ascertained. As none of the differences were significant, this was not pursued further and NJOY94.10 was considered to be functioning correctly.

3 Winfrith Updates

The previous version of NJOY in use at Winfrith, NJOY91.91w25, contained local modifications and extensions (2). When the code package for NJOY94.10 was installed, checks were made to ascertain whether any of these modifications had been included in the latest version of NJOY. Any changes which were not included in the latest version and which were still required had to be made consistent with the new source coding.

It was found that all but two of the updates applied to NJOY91.91w25 were still required. The two which had been superseded were:

Update 8

This update corrected an error in ACER in which the first argument of subroutine gety1 was specified as an integer rather than a real value. This has now been corrected in the standard code.

Update 25

This was standard update 91.92 which was added as a Winfrith update after the installation of NJOY91.91 and is thus now included in the standard code.

Part of update 7 added a new weighting spectrum (e - constant - 1/e) to GROUPR. This spectrum was found to give unsatisfactory results and has therefore been removed from the update. Other modifications made by update 7 have been retained.

The remaining Winfrith updates were applied to produce code version NJOY94.10w23. The updates are listed, in numerical order, in Appendix 2. These include an update (upw22) which causes the code revert to an earlier scheme for interpolation between thermal scattering matrices in the GROUPR module. The need for this update is described in Section 4.

The eight NEA and eleven Winfrith test cases were run and the results compared with those from NJOY91.91w25. The only remaining significant differences were in the thermal scattering matrices produced by THERMR. These differences were due to an improvement (update 91.116) in the calculation of thermal angular distributions in NJOY94.

4 Thermal Scattering Interpolation in GROUPR

In NJOY91.91, a new scheme for interpolating between thermal scattering matrices was introduced to the GROUPR module. This scheme interpolated between secondary energy distributions, tabulated at adjacent incident neutron energies, along lines of constant energy transfer to improve the representation of the excitation peaks. There was a problem with the new interpolation scheme which was detailed in Reference 3 together with a suggested fix which removed part of update 91.64 thus reverting to the previous interpolation scheme as used in NJOY89. This update was applied to NJOY91.91w25. It was reported that the problem with the new interpolation scheme had been corrected in later versions of NJOY91 (updates 91.102 and 91.105). A new test case, to generate data for hydrogen in water in the XMAS 172 group scheme, was run using both the old NJOY89 and the new NJOY94 interpolation schemes. These tests showed that the two interpolation schemes produced significant differences in the group to group thermal scattering matrices for hydrogen in water. These differences were most significant for high energy loss events (i.e. at the down-scatter tail of the secondary energy distributions) having a magnitude of up to 200%. In order to gauge the effect of these differences for a practical calculation, hydrogen data from both schemes were added to a WIMS library and a PWR pincell calculation performed. The results from this calculation showed that the NJOY94 data reduced k-infinity by ~40PCM thus moving the result further away from the value from MONK7 (5). Comparison of the group fluxes showed that, while values from the NJOY89 data and MONK7 were in good agreement, the values from the NJOY94 data overestimated the flux in the lowest energy group by ~25%.

The next step was to produce a 'reference' set of data for hydrogen that was independent of interpolation scheme. This was done using modified versions of the GROUPR and THERMR modules. First, GROUPR was modified to print out the integration energy grid used to produce group values, then THERMR was modified to use this grid as the incident energy grid for the generation of secondary energy distributions. Thus, these thermal scattering data produced by the modified THERMR would require no interpolation in GROUPR.

Using point data from the modified THERMR, hydrogen data were generated using both the NJOY89 and NJOY94 interpolation schemes. The thermal scattering matrices from the two schemes were identical, indicating that the effects of the interpolation had been removed. It was now possible to compared the thermal scattering matrices for the reference data with those generated previously from the two interpolation schemes. This comparison revealed two trends; the NJOY89 scheme gave much better results for high energy loss events (e.g. average error for downscatter into group 1 was less than 1% for NJOY89 but ~50% for NJOY94), the NJOY94 scheme was somewhat better for near scatter events. To test which of these effects was causing the observed reduction on k-effective, the reference data were added to the WIMS library and the PWR pincell case rerun. The k-infinity results for all three cases are given in Table 1 below.

Table 1 : **K-infinity for PWR Pincell**

Reference	NJOY89 Interpolation	PCM	NJOY94 Interpolation	PCM
1.38480	1.38483	3	1.38441	-39

Thus it can be seen that the overestimation of high energy loss scatter events by the NJOY94 interpolation scheme is responsible for the reduction in k-effective in this case. A comparison of the flux in the lowest 10 energy groups is shown in Table 2 below.

Table 2 : **Differences in Group Flux for PWR Pincell**

Group	Upper Energy (eV)	Reference	% Diff.	
			NJOY89 Interpolation	NJOY94 Interpolation
163	4.20E-02	2.39158E-01	1.28	-1.82
164	3.50E-02	1.71682E-01	1.00	-1.90
165	3.00E-02	1.67301E-01	0.89	-1.83
166	2.50E-02	1.57219E-01	0.76	-1.35
167	2.00E-02	1.39914E-01	0.69	-0.27
168	1.50E-02	1.13721E-01	0.32	1.77
169	1.00E-02	5.26793E-02	-0.09	5.03
170	6.90E-03	2.40120E-02	-0.32	8.37
171	5.00E-03	1.75749E-02	-0.54	12.35
172	3.00E-03	1.00740E-02	-0.65	21.26

These results show that, in the lowest groups, the data from the NJOY89 interpolation scheme model the reference results very closely while the NJOY94 data significantly overestimate the flux in these groups.

The same procedure was followed for carbon in graphite and a similar pattern of results was obtained. The NJOY94 interpolation scheme again resulted in an overestimation of the scattering source into the lower energy groups. An AGR testcase was used to assess the effect on k-infinity, the results of which are given in Table 3.

Table 3 : K-infinity for Hinkley Point AGR

Reference	NJOY89 Interpolation	PCM	NJOY94 Interpolation	PCM
1.24241	1.24235	-6	1.24183	-58

The fluxes also followed similar pattern to the PWR pincell case with the NJOY94 scheme data leading to a significant overestimation of the flux in the lowest groups. A comparison of the flux in the lowest 10 energy groups is shown in Table 4 below.

Table 4 : Differences in Group Flux for Hinkley Point AGR

Group	Upper Energy (eV)	Reference Flux	% Diff.	
			NJOY89 Interpolation	NJOY94 Interpolation
163	4.20E-02	1.02534E-02	-0.31	-1.17
164	3.50E-02	7.56257E-03	0.19	-1.44
165	3.00E-02	7.50188E-03	0.88	-1.66
166	2.50E-02	7.29903E-03	-0.40	-1.79
167	2.00E-02	6.66614E-03	-2.54	-1.69
168	1.50E-02	5.34241E-03	-2.27	-1.06
169	1.00E-02	2.61209E-03	-0.61	1.39
170	6.90E-03	1.23591E-03	1.77	5.18
171	5.00E-03	9.17690E-04	3.86	10.83
172	3.00E-03	6.07194E-04	4.74	26.12

It was thus concluded that, for both light water and graphite systems, the older NJOY89 interpolation scheme's overall representation of thermal scattering was superior to that from the NJOY94 scheme. Accordingly, the NJOY89 interpolation scheme was retained in the Winfrith installation of NJOY94.10 through the application of upw22.

5 Conclusions

NJOY version 94.10 has been installed at Winfrith.

A number of local amendments have been applied to bring the code to an acceptable condition for use by AEA Technology. The installed code has the designation NJOY94.10w23. The amendments are listed in Appendix 2 to allow them to be assessed, by the code author, for possible inclusion in later NJOY releases.

The latest scheme for interpolating in thermal scattering matrices in GROUPR has been replaced by an earlier scheme used in NJOY89. Tests have shown that, for light water and graphite systems, the older scheme gives a better representation of thermal scattering.

6 References

1 R E MacFarlane, D W Muir
The NJOY Nuclear Data Processing System - Version 91
LA-12740-M
October 1994

2 C R Eaton
Modifications to NJOY-91.91 Made at Winfrith
NJOY/EATON-02
May 1994

3 R C L van der Stad
A Problem in the Treatment of the Thermal Scattering Matrix in NJOY91.91
NJOY/VAN-DER-STAD-01
8th June 1994.

4 E B Webster
A Numerical Problem with NJOY89/BROADR
NJOY/WEBSTER-01

5 J Rowlands et al
An intercomparison of Calculations Made for Thermal Reactor Pin Cell benchmarks and the Small Los Alamos Fast Critical Assemblies using the JEF2.2 Library in Different Codes.
Proceedings of the International Conference on the Physics of Reactors PHYSOR 96, 61-20 September 1996
Volume 2, C-102

Appendices

Contents

APPENDIX 1 : Description of Test Cases Run at Winfrith

The standard test cases provided with the NJOY91 package are as follows :-

Case 1	C-nat ENDF/B-V Modules used - MODER RECONR BROADR HEATR THERMR THERMR GROUPR MODER
Case 2	Pu-238 from ENDF/B-IV Modules used - MODER RECONR BROADR UNRESR GROUPR CCCCR
Case 3	Photon interaction cross sections from dlc7e (ENDF-4) Modules used - RECONR GAMINR DTFR MATXSR
Case 4	U-235 10% PENDF for ERRORR test problem from t511 (ENDF-5) Modules used - MODER RECONR ERRORR GROUPR ERRORR
Case 5	C-nat ENDF/B-V Modules used - MODER ERRORR COVR
Case 6	C-nat ENDF/B-V Modules used - PLOTR
Case 7	U-235 from ENDF/B-V tape 511 Modules used - MODER RECONR BROADR HEATR GROUPR ACER
Case 8	Ni61 from ENDF/B-VI tape eni61 Modules used - MODER RECONR BROADR HEATR GROUPR ACER

The additional test cases which are run at Winfrith to test local extensions and modifications are:-

Case 15 Pu-238 from ENDF/B-IV.

An adaptation of standard test case 2 with the number of background cross sections and the number of legendre orders changed in GROUPR to ensure that a correction made at Winfrith for NJOY89 has been properly applied.

Modules used -

MODER RECONR BROADR UNRESR GROUPR CCCCR

Case 16 Fe-54 from JEF2.2.

Tests the use of ndigits > 7. It also tests the modification to UNRESR to increase the maximum number of temperatures allowed to 10.

Modules used -

MODER RECONR BROADR UNRESR THERMR GROUPR

Case 17 U-238 from JEF2.2.

Tests the use of the WIMSR module the modification to UNRESR to increase the maximum number of sigma zero values allowed to 10 and the modifications to the flux options in GROUPR.

Also tests the flux generator modifications.

Modules used -

MODER UNRESR THERMR GROUPR WIMSR

Case 18 H1 from JEF2.2.

Tests the new WIMSR option to apply user input current weighting to the calculation of the column sum correction to the transport cross-section.

Modules used -

MODER GROUPR WIMSR

Case 19 Fe56 from ENDF/B-VI)

Tests the treatment of spin dependent channel radii (naps=1) Reich-Moore formalism.

Modules used -

MODER RECONR BROADR UNRESR GROUPR

Case 20	U-238 from JEF2.2 Tests the treatment of tabular energy distributions in GROUPR (delayed fission spectra). Modules used - MODER GROUPR
Case 21	Cm245 from JEF2.2. Tests the processing of unresolved resonance parameters for Cm245. Modules used - MODER RECONR BROADR UNRESR GROUPR
Case 22	Test case for E Webster's work (4). Test new weighting function option (iwt=13) and 12 temperatures. NAG version of ERF in BROADR. Modules used - MODER RECONR BROADR HEATR UNRESR THERMR GROUPR
Case 23	H1 from JEF2.2. WIMSR 69 groups. Modules used - WIMSR
Case 24	U-238 from JEF2.2. WIMSR 69 groups. Modules used - WIMSR
Case 25	Al27 from JEF2.2. Test processing of (n,alpha) reaction. Modules used - RECONR
Case 26	Fe57 from ENDF/B-VI COVR was failing when outputting a condensed BCD covariance library. Modules used - RECONR ERRORR COVR

APPENDIX 2 : Updates Applied to NJOY94.10 at Winfrith

```
*/
*/
*/ Machine dependant updates required for running NJOY94 at Winfrith
*/
*ident upw1
*/
*/ NJOY -- 5 Feb 93 -- Remove call to open NJOY output file.
*/                     Set arguments of sun timing routines to
*/                     single precision.
*/                     Due to Winfrith automatic doubling of source
*d sun.9,11
*d sun.26
      real*4 tarray(2)
*d sun.28
      call etime(tarray)
      time = tarray(1)+tarray(2)
*d sun.31
      integer*4 iarray(3)
*/
*ident upw2
*/
*/ NJOY - 11 May 1993  - put in command to empty buffers after each
*/                         module. If job is looping and has to be
*/                         interrupted we know which module it is in.
*/
*b njoy.278
c empty print buffers when module completed.
c if job has to be stopped due to looping etc we then know the module.
      call flush(6)
*/
*ident upw3
*/
*/ DTFR  - 11 May 1993 - modifications for single word length machine
*/                     - character arrays set to real*8
*/
*i dtfr.86
      real iednam,hisnam,hz
*i dtfr.499
      real iednam,kmtid,iblank,iword,nabs,nusf,ntotl,iz
*i dtfr.724
      real iednam,isnam,mti,hdat
*i dtfr.879
      real iednam,isnam
*i dtfr.1186
      real isnam
*i dtfr.1316
      real isnam
*/
*ident upw4
*/
*/ MATXSR - 11 May 1993 - modifications for single word length machine
*/                      - character arrays set to real*8
*/
*d matxsr.389
      implicit real*8 (h)
*i matxsr.467
      huse(1)=hz(2)
      huse(2)=hz(2+mult)
*d matxsr.471,472
*d matxsr.521
      implicit real*8 (h)
*d matxsr.760
      implicit real*8 (h)
```

```
*d matxsr.926
      implicit real*8 (h)
*d matxsr.1317
      implicit real*8 (h)
*d matxsr.1490
      implicit real*8 (h)
*d matxsr.1706
      implicit real*8 (h)
*d matxsr.1799
      implicit real*8 (h)
*d matxsr.1819
      implicit real*8 (h)
*/
****************************************************************************
```

upw5 and upw6 not shown.

```
****************************************************************************
*/
*ident upw7
*/
*/ GROUPR - 25th May 1993 - Add new group structure options (ign=30,31).
*/                          Add new weight function option (iwt=13).
*/                         IWT=13 is similar to IWT = 12 but the Maxwellian
*/                          join is set to 2.5KT to account for high
*/                          temperatures.
*/                        - NJOY89 ident mar18w and aug17w
*/
*i groupr.64
c modification by C Eaton, AEA Technology, Winfrith
c Add new group structure options (ign=30,31)
c        30      mural structure
c        31      dice structure
c     *                                                                    *
c     *          if ign=30, read card6c                                    *
c     * card6c                                                             *
c     *    ezero   top energy boundary (ev)                                *
c     *    width   lethargy width                                          *
c     *    ng      no. of groups                                           *
c     *          if ign=31  read card6d,etc                                *
c     * card6d                                                             *
c     *    nrange  no. of energy group ranges                              *
c     *    npri    print option  (=0,boundaries not printed)               *
c     * card6e,....    nrange cards,given in decreasing energy order       *
c     *    etop    top energy boundary of range                            *
c     *    nopt    option number (see width below)                         *
c     *    ngr     number of groups in range                               *
c     *    width   meaning controlled by nopt                              *
c     *            nopt=1,width is energy width                            *
c     *            nopt=2,width is a group width                           *
*i groupr.129
c     *     30           mural group structure                             *
c     *     31           dice group structure                              *
*i groupr.158
c     *     13           vit-e with t-dep thermal part (2.5KT join)        *
*d groupr.218
      common/groupn/ign,ngn,egn(100000)
*d groupr.708
      common/groupn/ign,ngn,egn(100000)
*i groupr.1175
c modification by C Eaton, AEA Technology, Winfrith
c Add new group structure options (ign=30,31)
c        30      mural structure
c        31      dice structure
*d groupr.1179
      common/groupn/ign,ng,eg(100000)
*d groupr.2107
      common/groupn/ign,ngn,egn(100000)
*d groupr.2484
```

```
      common/groupn/ign,ngn,egn(100000)
*d groupr.3397
      common/groupn/ign,ngn,egn(100000)
*d groupr.5143
      common/groupn/ign,ngn,egn(100000)
*d groupr.5560
      common/groupn/ign,ngn,egn(100000)
*i groupr.1187
      dimension zinput(4)
*d groupr.1256
      data ngmax/100000/
*d groupr.1293
     1 1200,1000,1300,1300,99,99,99,99,99,99,99,99,99,
     2 99,99,99,3000,3100) ign
*d groupr.1294
   99 call error('gengpn','illegal groupr structure.',' ')
*i groupr.1468
c modification by C Eaton, AEA Technology, Winfrith
c Add new group structure options (ign=30,31)
c
c  mural structure
c
 3000 continue
      call infree(nsysi,zinput,3,4)
      ezero = zinput(1)
      width = zinput(2)
      ng    = nint(zinput(3))
      ng1   = ng + 1
c
      if(ng1.gt.ngmax) call error('gengpn','too many groups.',' ')
c
c
      eg(1) = float(ng) * width
      do 3010 i=2,ng1
      eg(i) = eg(i-1) - width
 3010 continue
      go to 1800
c modification by C Eaton, AEA Technology, Winfrith
c Add new group structure options (ign=30,31)
c
c dice structure
c
 3100 continue
      zinput(2) = 0.0
      next = 1
      ntopg = 1
c no. of energy ranges to follow
c print indicator npri = 0 - dont print group boundaries
      call infree(nsysi,zinput,2,4)
      nrange = nint(zinput(1))
      npri   = nint(zinput(2))
      write(nsyso,991)
      do 3110 i=1,nrange
c top energy of range , option , no groups in range
c (ranges should be given as highest energy range first)
        call infree(nsysi,zinput,4,4)
        etop = zinput(1)
        nopt = nint(zinput(2))
        ngr  = nint(zinput(3))
        width = zinput(4)
        write(nsyso,999) etop,ngr,nopt,width
        ntopg = ntopg + ngr
        if(ntopg.gt.ngmax) call error('gengpn','too many groups.',' ')
c
      if(nopt.ne.1) go to 3130
c
c lethargy width
c given as integer n where e(i) =  e(i+1) * exp(-1/n)
c
```

```
      width = -width
      rn = 0.0
      tol = width / 10.
      eg(ntopg) = etop
      if(ngr.eq.1) go to 3124
      do 3121 j=1,ngr-1
        rn = float(j) * width
        ipos = ntopg - j
        eg(ipos) = etop * exp(rn)
 3121 continue
 3124 continue
      if(i.ne.1) go to 3122
        eg(1) = eg(2) * exp(width)
        go to 3111
 3122 continue
      testeg = eg(next) * exp(width)
      if(abs((testeg-eg(next-1))/testeg).gt.0.00001)
     *  write(nsyso,1) testeg,eg(next-1)
      go to 3111
c
 3130 continue
      if(nopt.ne.2) go to 3130
c group width
      eg(ntopg) = etop
c
      if(ngr.eq.1) go to 3134
      do 3131 j=1,ngr-1
        ipos = ntopg - j
        eg(ipos) = eg(ipos + 1) - width
 3131 continue
 3134 continue
      if(i.eq.1) then
        eg(1) = eg(2) - width
c
c testing if bottom group is zero
c set to lowest energy given (1.e-05)
c note njoy routines use energy=0.0 as an initialisation input parameter
c
        if(eg(1).lt.1.0e-5)eg(1)=1.0e-5
      else
        testeg = eg(next) - width
        if(abs((testeg-eg(next-1))/testeg).gt.0.00001)
     1     write(nsyso,1) testeg,eg(next-1)
      endif
c
 3111 next = ntopg + 1
 3110 continue
      ng = ntopg - 1
      go to 1900
*i groupr.1493
      if (ign.eq.30) write(nsyso,340)
      if (nshort.gt.0.and.ign.eq.30) write(nshort,340)
      if (ign.eq.31) write(nsyso,350)
      if (nshort.gt.0.and.ign.eq.31) write(nshort,350)
      if(ign.eq.31.and.npri.eq.0) return
*i groupr.1497
    1 format(1x,'error gengpn ',1p2e14.7)
*i groupr.1515
  340 format(/35h neutron group structure......mural)
  350 format(/34h neutron group structure......dice)
  999 format(1x,1pe14.6,3x,i6,7x,i1,4x,e14.6)
  991 format(14h energy ranges/35h   top energy    no. groups   option )
*i groupr.1708
c modification by C Eaton, AEA Technology, Winfrith
c Add new weight function option (iwt=13).
c IWT=13 is similar to IWT = 12 but the Maxwellian
c join is set to 2.5KT to account for high temperatures.
c        13       vit-e with t-dep thermal part (2.5KT join)
*d groupr.1815
```

```
      go to (110,120,130,140,150,160,160,180,190,190,200,200,210) iwtt
*i groupr.1894
c modification by C Eaton, AEA Technology, Winfrith
c Add new weight function option (iwt=13).
c IWT=13 is similar to IWT = 12 but the Maxwellian
c join is set to 2.5KT to account for high temperatures.
c
c     ***vitamin-e weight function (2.5KT join)
  210 write(nsyso,75)
      if (iwtt.eq.13) write(nsyso,17)
      return
c
*i groupr.1909
   17 format(22x,'temperature dependent (2.5KT join)')
*b groupr.1936
      common/mainio/nsysi,nsyso,nsyse,ntty
*d groupr.1959
      go to (100,200,300,400,100,600,600,100,100,70,700,700,700)iwtt
*i groupr.2036
      enn1 = en1
      if(iwtt.eq.13)enn1 = 2.5*temp(jtemp)*8.61735e-5
*d groupr.2037
      if (e.ge.enn1) go to 710
*d groupr.2041
      if (iwtt.gt.11) cc=con2*exp(enn1/tt)/enn1**2
*d groupr.2043
      if (enext.gt.enn1) enext=enn1
*i groupr.2132
  920 continue
*d groupr.2130,2132
c set first energy boundary to the smallest energy given in the total
c cross-sections
c
      if (enext.gt.1.001*egn(1))then
         call mess ('genflx total not defined over energy range.',' ')
         egn1 = enext
      endif
*d groupr.2942,2943
   60 format(1x,i6,3x,i6,1x,1p10e11.3)
   65 format(1x,i6,1x,1p10e11.3)
*d groupr.2945
   75 format(1x,i6,5x,i1,2x,1p10e11.3)
*d groupr.2947
   85 format(1x,i6,3x,i6,5x,i1,2x,1p10e11.3)
*d groupr.690
     1   3h  (,i6,8h groups))
*d groupr.1916
     1  4x,14hninwt, jsigz =,i3,i6)
*/
*/
*ident upw8
*/
*/ GROUPR - 12th May 1993 - Extend the range of Hydrogen scatter
*/                        - Addition of double precisioning needed
*/                          for IBM machines.
*/                        - NJOY89 ident jul26w (26 july 1990)
*/
*i groupr.5135
c     real*8 whi,wlo
*d groupr.5260,5261
c modification by C Eaton, AEA Technology, Winfrith
c Extend the range of Hydrogen scatter
      if (whi.lt..99999995*wmin) whi=wmin
      if (whi.gt..99999995*wmax) whi=wmax
*/
*ident upw9
*/
*/ BROADR - 12th May 1993 - Add additional input parameter to BROADR
*/                        - Remove subroutine DIGITS
```

```
*/                             - NJOY89 ident sep19w
*/
*i broadr.55
c     *    ndigit   no. significant digits (default=7)
*d broadr.153,154
     2  '' errmax[20*errthn], errint[errthn/10000], ndigit(7).'')')
      nz=5
*i broadr.157
      z(5)=7
*i broadr.164
      ndigit=nint(z(5))
*d broadr.169
     1   thnmx,errmax,errint,ndigit
*d broadr.231
*d broadr.243
*d broadr.526
     a    40h errint for thinning .................. ,1pe13.5,/,
     b    40h no. significant figures .............. ,i10/)
*d broadr.641,659
*/
*ident upw10
*/
*/ GROUPR -- 20 May 93 -- Modification to subroutine GETFF of GROUPR
*/                        to process the normalized
*/                        fission spectrum correctly (mf=5,mt=542)
*/                        and delayed fission spectrum (mf=5,mt=455)
*/                        Test cases 17 and 20 were failing in FINDF
*/                        trying to locate the reaction mf5 mt452 on
*/                        the PENDF tape.
*/ GROUPR -- 16 July 93 -- Modification to subroutine GETSED of
*/                         GROUPR to process secondary energy
*/                         data with more than one tabular spectrum.
*/
*d groupr.3478
c modification by C Eaton, AEA Technology, Winfrith
c  Modification to subroutine GETFF of GROUPR to process the normalized
c  normalized fission spectrum correctly (mf=5,mt=542)
c  and delayed fission spectrum (mf=5,mt=455)
c  Test cases 17 and 20 were failing in FINDF trying to locate the
c  reaction mf5 mt452 on the PENDF tape.
      mtt=mtd
      if (mtd.eq.452) mtt=18
      if (mtd.eq.455) mtt=455
      if (mtd.eq.456) mtt=19
      if (e.ne.0.) then
c
c normalized fission spectrum
c
         if (mfd.eq.5) then
            e=1.e6
            write(nsyso,999)e
 999        format (' Normalized fission spectrum at ',1pe12.5,' eV')
            if (mtd.eq.455) then
               e=2.e6
               write(nsyso,999)e
            endif
         endif
      endif
      call getsed(e,en,idis,a(ised),egn,ngn,nk,matd,5,mtt,nendf,a)
c end of modification
*i groupr.7054
c modification by C Eaton, AEA Technology, Winfrith
c Modification to subroutine GETSED of
c GROUPR to process secondary energy
c data with more than one tabular spectrum.
      il = 1 - ic
*/
*ident upw11
*/
```

```
*/ RECONR - 25th May 1993 - correct the processing of unresolved
*/                           resonance parameters. (Cm245)
*/                           One change has been made as up38.
*/                         - NJOY89 ident nov92w
*/
*d reconr.786
c modification by C Eaton, AEA Technology, Winfrith
c correct the processing of unresolved resonance parameters. (Cm245)
      if (ener.lt.el.or.ener.ge.eh) go to 340
*/
*ident upw12
*/
*/ GROUPR - 19th June 1993 - Allow output of flux for all values of
*/                           sigma zero.
*/                         - NJOY89 ident mar91w
*/
*/ Allow output of flux for all values of sigma zero.
*/
*d groupr.235
      common/gstore/a(250000)
*d groupr.251
      iamax=250000
*i groupr.2314
C if jsigz lt 0 flux will be output for all background cross-sections.
      if(jsigz.gt.0.0) then
*i groupr.2317
      endif
*i groupr.2335
C if jsigz lt 0 will be output for all background cross-sections.
      if(jsigz.lt.0.0) then
      write(ninwt) e,(fout(isz),isz=2,(nl*nsigz),2)
      else
*i groupr.2337
      endif
*d groupr.2379
C if jsigz lt 0 will be output for all background cross-sections.
      if(jsigz.lt.0.0) then
      write(ninwt) e,(fout(isz),isz=2,(nl*nsigz),2)
      else
      write(ninwt) e,fout(indx),xc
      endif
*/
*ident upw13
*/
*/ BROADR - 2nd June 1993 - Increase maximum number of allowed
*/                          temperatures to 15.
*/
*d broadr.39
c     *     ntemp2   number of final temperatures (maximum=15)          *
*d broadr.102
      dimension temp2(15),z(7),tt(10),mtr(10),mti(10)
*d broadr.149
      if (ntemp2.gt.15) call error('broadr',
*/
*ident upw14
*/
*/ UNRESR - 2nd June 1993 - Increase maximum number of allowed
*/                          temperatures to 15.
*/
*d unresr.25
c     *     ntemp    no. of temperatures (15 max)                       *
*d unresr.45
      dimension temp(15),sigz(10),bkgz(4),sigu(5,10)
*d unresr.49
      data ntmax/15/, nzmax/10/, ipr/1/
*/
*ident upw15
*/
*/ THERMR - 2nd June 1993 - Increase maximum number of allowed
```

```
*/                            temperatures to 15.
*/
*d thermr.93
     1  tempr(15),tol,emax,law,nbuf,fact,nwscr,mtref,ncdse
*d thermr.95
      dimension z(10),ex(2),eftemp(15),eftmp2(15)
*d thermr.528
     1  tempr(15),tol,emax,law,nbuf,fact,nwscr,mtref,ncdse
*d thermr.929
     1  tempr(15),tol,emax,law,nbuf,fact,nwscr,mtref,ncdse
*d thermr.1154
     1  tempr(15),tol,emax,law,nbuf,fact,nwscr,mtref,ncdse
*d thermr.1936
     1  tempr(15),tol,emax,law,nbuf,fact,nwscr,mtref,ncdse
*/
*ident upw16
*/
*/ GROUPR - 2nd June 1993 - Increase maximum number of allowed
*/                            temperatures to 15.
*/
*d groupr.221
      common/temper/ntemp,temp(15)
*d groupr.712
      common/temper/ntemp,temp(15)
*d groupr.716
      data ntmax/15/, nzmax/10/
*d groupr.1935
      common/temper/ntemp,temp(15)
*d groupr.2106
      common/temper/ntemp,temp(15)
*d groupr.2407
      common/temper/ntemp,temp(15)
*/
*ident upw17
*/
*/ BROADR - 7th June 1993 - Modification by E. Webster.
*/                            Use NAG Error function routine
*/                            S15ADF.
*/
*d broadr.1230
      ifail = 0
      f(1)=0.5*s15adf(a,ifail)
*/
*/
*ident upw18
*/
*/ NJOY - 19th June 1993 - Increase size of format in routine RESERV.
*/                          and RELEAS,
*/
*d njoy.2582
   30 format(57x,'id ',a4,1x,i3,'/',i9)
*d njoy.2624
     1    '(''need '',i9,'' more words for id '',a4,''.'')') need,id
*d njoy.2688
   40 format(57x,3hxx ,a4,5x,i9)
*/
*ident upw19
*/
*/ BROADR - 19 August 1993 - Modification to subroutine BFILE3
*/                             to process JEF2.2 H1 . Was failing
*/                             because there were less points on
*/                             the file than the page size used
*/                             to doppler broaden. A dummy point
*/                             has been added above the top energy.
*/                             The code previously failed with
*/                             a segmentation violation. The same
*/                             correction is required to NJOY89.62w.
*/
*i broadr.699
```

```
c
c modification by C Eaton, AEA Technology, Winfrith
c if less points on file than a page then add extra point above
c top energy
c
      if(n2in.lt.npage)then
        e(mhigh+1) = e(mhigh)*1.00001
        do 116 i=1,nreac
 116      s(i,mhigh+1)=tt(i+1)
      endif
c modification end
*/
*ident upw20
*/
*/ RECONR - 20 May 1994 - Correction to ident up84 to process JEF2.2 A127
*/                        Subroutine LUNION modified to include additional
*/                        call to MOREIO.
*/
*d reconr.1260
      if (sr.lt.1e-15.and.srnext.lt.1e-15) then
          if ((ibase+2*ir).lt.nw.or.nb.eq.0) go to 205
          call moreio(nin,nout,0,a(ibase+1),nb,nw)
          kr=kr+ir
          ir=0
          nw=ibase+nw
          go to 205
       endif
*ident upw21
*/
*/ COVR - 20 May 1994 - Replace hollerith strings by character strings.
*/                      Update subroutine SETFOR to use character variables
*/
*d covr.1655
      dimension a(1),ia(1),xa(nr,nc)
      character*12 ivft,icft
*d covr.1827,1832
      character*12 ift(14),ivft,icft
      data ift /
     1 '(80i1)        ', '(40i2)        ', '(26i3)        ',
     2 '(20i4)        ', '(16i5)        ', '(13i6)        ',
     3 '(11f7.4)      ', '(10f8.5)      ', '(1p8e9.2)     ',
     4 '(1p8e10.3)    ', '(1p7e11.4)    ', '(1p6e12.5)    ',
     5 '(1p6e13.6)    ', '(1p5e14.7)    ' /
*d covr.1842,1844
      ivft=ift(nvf)
  110 icft=ift(ncf)
*ident upw22
*/
*/ GROUPR - 14 June 1994 - Thermal Problem in GROUPR
*/                        - Modification suggested by Van der Stad of Petten
*/                          Removal of part of up64
*/
*d groupr.5565,5566
      save llo,nlo,elo
      save lhi,nhi,ehi
      save ibrag,jbrag
      data nbin/30/
*d groupr.5673
  235 lhi=loc
      if (ltt.eq.6) go to 240
      call bina(a(13),a(lhi),nbin)
      loc=loc+nbin+1
      if (loc-iaes.gt.nwmax)
*d groupr.5675
  240 nw2=loc-13
*i groupr.5599
      if (ltt.eq.6) go to 130
      call bina(a(12),a(llo),nbin)
      loc=loc+nbin+1
```

```
*d groupr.5602,5603
  130 nw1=loc-12
*d groupr.5692,5693
  320 j=j+1
      ll=llo+j-1
      lh=lhi+j-1
      ej=a(ll)+(a(lh)-a(ll))*(e-elo)/(ehi-elo)
      ejp=a(ll+1)+(a(lh+1)-a(ll+1))*(e-elo)/(ehi-elo)
      if (ejp.le.egp) go to 320
      egp1=a(ll)+(a(ll+1)-a(ll))*(egp-ej)/(ejp-ej)
      egp2=a(lh)+(a(lh+1)-a(lh))*(egp-ej)/(ejp-ej)
*d groupr.5699,5704
  335 if (egp.lt.ejp) go to 340
      if (j.eq.nbin) go to 340
      j=j+1
      ll=ll+1
      ej=ejp
      lh=lh+1
      ejp=a(ll+1)+(a(lh+1)-a(ll+1))*(e-elo)/(ehi-elo)
      go to 335
  340 egp1=a(ll)+(a(ll+1)-a(ll))*(egp-ej)/(ejp-ej)
      egp2=a(lh)+(a(lh+1)-a(lh))*(egp-ej)/(ejp-ej)
*i groupr.5859
c
      subroutine bina(a,b,nbin)
c     ******************************************************************
c     divide the spectrum record in a into nbin regions
c     ******************************************************************
      character*60 strng
      dimension a(10),b(1)
c
      ncyc=nint(a(6))
      np=nint(a(5))/ncyc
      fract=1./nbin
      sum=0.
      i=0
      j=0
  100 i=i+1
      xl=a(7+ncyc*(i-1))
      yl=a(8+ncyc*(i-1))
      y=a(8+ncyc*i)
      if (y.eq.0.) go to 100
      b(1)=xl
  110 i=i+1
      x=a(7+ncyc*(i-1))
      y=a(8+ncyc*(i-1))
  120 add=0.5*(y+yl)*(x-xl)
      if (i.eq.np.and.j.eq.nbin-1) go to 130
      if (sum+add.ge.fract) go to 140
      sum=sum+add
      xl=x
      yl=y
      go to 110
  130 xn=x
      j=j+1
      go to 160
  140 j=j+1
      if (abs(y-yl).gt..00001*(y+yl)) go to 145
      xn=xl+(fract-sum)/yl
      if (xn.gt.x) go to 155
      if (xn.gt.xl.and.xn.le.x) go to 160
  145 f=(y-yl)/(x-xl)
      disc=(yl/f)**2+2.*(fract-sum)/f
      if (disc.gt.0.) go to 150
      write(strng,'(''discriminant='',1p,e12.4)') disc
      call mess('bina',strng,'set to abs value and continue.')
      disc=abs(disc)
  150 if (f.gt.0.) xn=xl-(yl/f)+sqrt(disc)
      if (f.lt.0.) xn=xl-(yl/f)-sqrt(disc)
```

```
      if (xn.gt.xl.and.xn.le.x) go to 160
      if (xn.gt.xl.and.abs(xn-x).le.abs(xn)*.0001) go to 160
  155 xn=x
  160 yn=yl+(y-yl)*(xn-xl)/(x-xl)
      b(j+1)=xn
      xl=xn
      yl=yn
      sum=0.
      if (j.lt.nbin.and.xl.ge.x) go to 110
      if (j.lt.nbin) go to 120
      b(nbin+1)=a(7+ncyc*(np-1))
      return
      end
*i groupr.4409
      epnn=0.
*/ This change probably not needed.
*/ d up64.253
*/      t=t+p(l)*0.5*(2*l-1)*tt
*ident upscy23
*/
*/ NJOY  - 7th May 1992 - Zero store at start of NJOY.
*/
*/ NJOY modification from Caroline RAEPSAET at Saclay
*/
*i njoy.2577
      do 128 ij=index,index+nwords-1
        ia(ij)=0
  128 continue
*ident versw
*/ update version number and date to correspond to last ident
*d vers.4
c     *        version 94.10w23 -- 22 april 96                         *
*d njoy.290
*d vers.6
      character vers*8,time*8,date*8,mx*4,lab*5
      data vers/'94.10w23'/
*d njoy.309
     3    17x,'*',1x,'vers.',a8,1x,'*'/
```