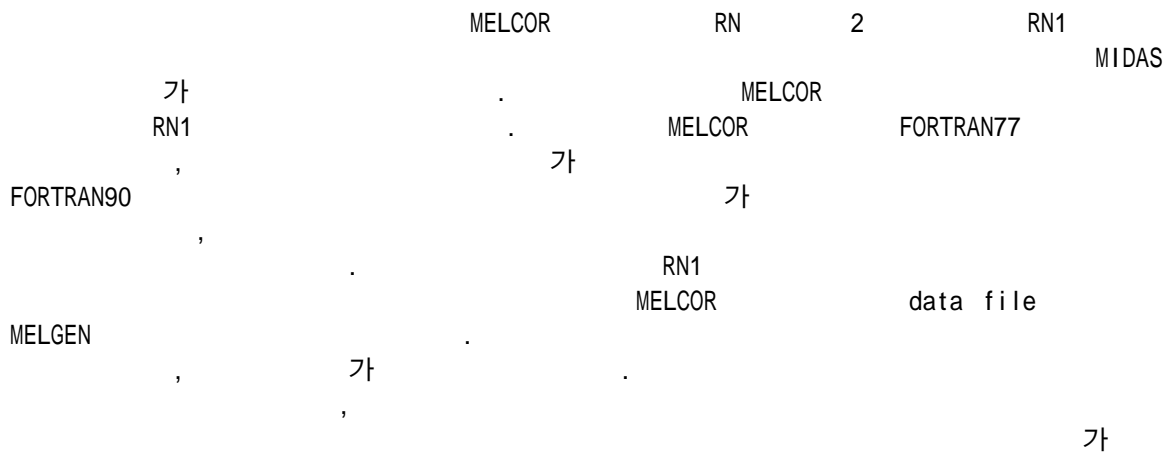


(MIDAS)

RN1

## A Restructuring of RN1 Package for MIDAS Computer Code

150



### Abstract

RN1 package, which is one of two fission product-related packages in MELCOR, has been restructured for the MIDAS computer code. MIDAS is being developed as an integrated severe accident analysis code with a user-friendly graphical user interface and modernized data structure. To do this, data transferring methods of current MELCOR code are modified and adopted into the RN1 package. The data structure of the current MELCOR code using FORTRAN77 causes a difficult grasping of meaning of the variables as well as waste of memory. New features of FORTRAN90 make it possible to allocate the storage dynamically and to use the user-defined data type, which lead to an efficient memory treatment and an easy understanding of the code. Restructuring of the RN1 package addressed in this paper includes module development, subroutine modification, and treats MELGEN, which generates data file, as well as MELCOR, which is processing a calculation. The verification has been done by comparing the results of the modified code with those from the existing code. As the trends are similar to each other, it hints that the same approach could be extended to the entire code package. It is expected that code restructuring will accelerate the code domestication thanks to direct understanding of each variable and easy implementation of modified or newly developed models.

1.

MELCOR MIDAS MELCOR 가  
 MELCOR 가  
 , 가  
 , 가  
 FORTRAN90 가  
 가 data type .[1,2,3]  
 , (readability) (DMM)  
 , 가 subroutine (derived type variables)  
 [4,5].  
 [6,7,8],

2 )  
 restart file MELGEN ( , MELCOR RN

2.

MELCOR code 3 restart file  
 MELGEN, restart file , log file plot file  
 MELCOR, PLOT  
 COR, HT, SPR, TF, RN1, RN2 20  
 가 , data  
 subroutine , message subroutine, code  
 subroutine subroutine [9].

### 2.1 Restart file

data  
 RESTART file read write subroutine MXXRS MXXRSW , MELCOR  
 read write , subroutine  
 xyzPRS 1, 2 , subroutine xyzRIO  
 coefficient , real, integer, logical,  
 character 4 array . (xyz :  
 )

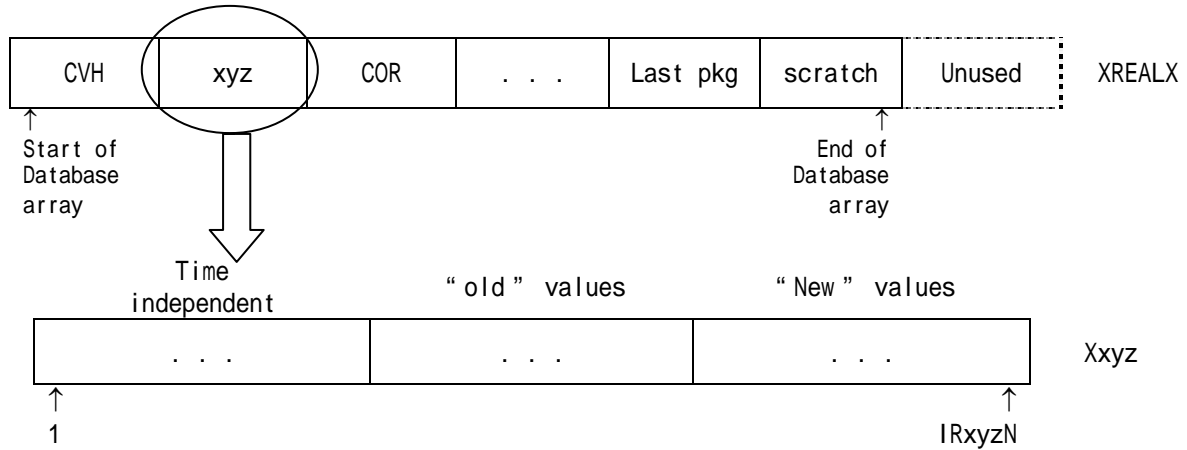
### 2.2 Database

MELCOR data  
 4가

- XREALX : floating point variables (500,000 array)

- INTEGE : integer variables (50,000 array)
- LOGICA : logical variables (5,000 array)
- CHARAC : character variables (30,000 array)

( 1 ),



1.

Database array subroutine 274  
 Database array argument 1  
 2

```

SUBROUTINE xyzDBC
*- INCLUDE BLANK
COMMON /DBREAL/ NEDREL, IRELCS, IRELES
COMMON / / DREALX(NUMREL/2)
DOUBLE PRECISION DREALX C
DIMENSION XREALX(NUMREL)
EQUIVALENCE (XREALX(1), DREALX(1))
DOUBLE PRECISION VREALX(NUMREL/2)
EQUIVALENCE (VREALX(1), DREALX(1))
*- INCLUDE xyzDB
COMMON /xyzDB/ IRxyzF , IRxyzN , IxyzF , IxyzN ,
1 ILxyzF , ILxyzN , ICxyzF , ICxyzN
CALL xyzDBD(
1 IRxyzN , XREALX(IRxyzF) ,
2 IxyzN , INTEGE(IxyzF) ,
3 ILxyzN , LOGICA(ILxyzF) ,
4 ICxyzN , CHARAC(ICxyzF), . . . )
C
RETURN
END

```

2. 1 database subroutine

Database 2  
array argument , subroutine 3 .

```

SUBROUTINE xyzDBD (
1 NxyzR , Rxyz ,
2 NxyzI , Ixyz ,
3 NxyzL , Lxyz ,
4 NxyzC , Cxyz , . . . )
C
  DIMENSION Rxyz(NxyzR), Ixyz(NxyzI), . . .
C
*- INCLUDE xyzPNT
COMMON /xyzPNT/ Nvar1, Nvar2, . . .
C
CALL xyzRUN (Nvar1, Rxyz(Ivar1), Nvar2, Ixyz(Ivar2), . . . )
C
RETURN
END

```

3. 2 database subroutine

Database argument ,  
subroutine 4 .

```

SUBROUTINE xyzRUN(
1 Nvar1, var1, Nvar2, var2, . . .)
C
  DIMENSION var1(Nvar1), . . .
C
FROM HERE ON, SIMPLY USE var1 AND var2.
FORGETTING ALL DETAILS OF WHERE AND HW THEY ARE STORED.
C
CALL xyz. . .
C
RETURN
END

```

4. database subroutine

2.3

subroutine	xyzDBD	subroutines	database	comment	argument	subroutine	xyzDBC
					pass		. RN1
(1) RN1	1						
1					( 5),		
subroutine		' RN1DB '	common block		( 6), 4	data	
type	2						
(2) RN1	2						
2						subroutine	
' xyzPNT '			common block		, 4	data type	
database			가		RN1		
	( )	7					



```

*- INCLUDE RN1PNT
C
C   RN1 DATABASE POINTERS
C
C   COMMON/RN1PNT/ IPMSEC,
+   IT1AGO, IT1ALO, IT1VGO, IT1VLO, ITADPO, ITVDPO,
+   IT1AGN, IT1ALN, IT1VGN, IT1VLN, ITADPN, ITVDPN,
+   IR1AGO, IR1ALO, IR1VGO, IR1VLO, IRADPO, IRVDPO,
+   IR1AGN, IR1ALN, IR1VGN, IR1VLN, IRADPN, IRVDPN,
+   IPMCRO, IPMCVO, IPMCRN, IPMCVN, IPEADO, IPEADN, IPCLCO,
+   IPRSRA, IPRSrv, IPIsRA, IPIsRV, IPSRCA, IPSRCV, IPIRSP,
+   IPRRSP, IPCN1B, IPCN2A, IPCN2B, IPCN3, IPCN4, IPCDEP,
+   IPCGRW, IPIVDS, IPDSUR, IPFRCR, IPRSMO, IPRSMN, IPRCOR,
+   IPRCAV, IPRFDI, IPMRLO, IPMRLN, IPCRCL, IPCLVN, IPVNCL,
+   IPMVNO, IPMVNN, IPMFDO, IPMFDN, IPMISC, IPNCCB, IPICCB,
+   IPXCCB, IORRN1, IOIRN1, NUMSEC, NUMCMP, NUMCLS, NUMSRA,
+   NUMSRV, NNSRA, NNSRV, NUMCOF, NUMSUR, NNSUR, IFLAGR,
+   NCNODE, NCMATA, NCLC, NCLCT, NNCLC, NNCLCT, NVANCL,
+   IPMCRI, ICVPTH, NAXLC, NRADC, NGAPC, KCMPC, NFMATC,
+   IXMFUC, IPFRLO, IPSDTO, IPFRLN, IPSDTN, IPDHAO, IPDHDO,
+   IPDHAN, IPDHDN, IPVDCO, IPRVCO, IPVDCN, IPRVCN, IPCAST,
+   IPCLCM, IPCAON, NUMCA, NNCA, IVCNDO, IVCNDN
C PARAMETERS ADDED BY R.K.COLE, JULY 1991
C POINTERS ADDED BY R.C.SMITH, ST. PATTY'S DAY 1992
C
C POINTERS ADDED BY M.F.YOUNG FOR CHEMISORPTION
C IPVDCO = pointer for old chemisorbed fission product masses
C IPRVCO = pointer for old radioactive fission product masses
C IPVDCN = pointer for new chemisorbed fission product masses
C IPRVCN = pointer for new radioactive fission product masses
C IPCAST = pointer for chemisorption class surface type
C IPCLCM = pointer for chemisorbed-to-fission product vapor class map
C IPCAON = pointer for CA class activation
C NUMCA = number of chemisorption classes
C NNCA = MAX(1,NUMCA)
C
C POINTERS ADDED BY R. SMITH
C IVCNDO = POINTER FOR OLD VOLUMETRIC CONDENSATION/EVAPORATION
C IVCNDN = POINTER FOR NEW VOLUMETRIC CONDENSATION/EVAPORATION
*-

```

7. RN1

2

( )

RN

2

RN1

, RN1

. MELCOR

subroutine, RN1



dimension  
1

array member variable



```

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!! RN1 (Radio Nuclide 1) package data !!!!!!!!!!!!!!!!!!!!!!!
!
!      IMPLICIT NONE
!      MODULE RN1_MDL
!
!      ***** RadioNuclide-1 GLOBAL DATA *****
!      INTEGER :: IFLAGR, IRSPA, KCMPC, NCMATA, NCNODE, NFMATC, NGAPC
!      INTEGER :: NAXLC, NRADC, NUMCLS, NUMCMP, NUMCOF, NUMSEC, NVANCL
!      INTEGER :: NUMCA, NNCA, NCLC, NNCLC, NCLCT, NNCLCT
!      INTEGER :: NDHSUR, NNDHSR, NDHVOL, NNDHVL, NRPSTT, NNPSTT
!      INTEGER :: NRPVOL, NNPVOL, NRCLSF, NNRCLF, NRCLST, NNRCLT
!      INTEGER :: NRCT, NNRCT, NRTRN, NNRTRN, NUMSET, NNSUR
!      INTEGER :: NUMSRA, NNSRA, NUMSRV, NNSRV, NUMSUR, NNSUR
!      INTEGER :: NTRCFP, NNTRFP, NTRN, NNTRN, NTRCTP, NNTRTP
!      INTEGER :: NTRVLP, NNTRVP, NTRSTP, NNTSTP, NTRRN, NNTRRN, NRN1_FLAG
!      INTEGER :: NRN1_IACTV, RN1_NNCAV, RN1_NNFD1, RN1_NFDDEP, RN1_NNVOL
!
!      TYPE RN1_N01 ; SEQUENCE
!      INTEGER :: ICLCO, ICLVAN
!      REAL :: XCORCL(8)
!      END TYPE
!      TYPE (RN1_N01), ALLOCATABLE :: RN1_NC0(:)
!
!      TYPE RN1_N02 ; SEQUENCE
!      REAL :: ADEPO, VDEPO, RADEPO, RVDEPO, &
!      ADEPN, VDEPN, RADEPN, RVDEPN
!      END TYPE
!      TYPE (RN1_N02), ALLOCATABLE :: RN1_NC1(:, :)
!
!      . . . . .
!
!      END MODULE RN1_MDL
!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

9. RN1

1. Array

ADEP(NCLS, NNSUR)	RN1_NC1(NUMCLS, NNSUR)%ADEPO
VDEP(NCLS, NNSUR)	RN1_NC1(NUMCLS, NNSUR)%VDEPO
RADEP(NCLS, NNSUR)	RN1_NC1(NUMCLS, NNSUR)%RADEPO
RVDEP(NCLS, NNSUR)	RN1_NC1(NUMCLS, NNSUR)%RVDEPO
ADEPN(NCLS, NNSUR)	RN1_NC1(NUMCLS, NNSUR)%ADEPN
VDEPN(NCLS, NNSUR)	RN1_NC1(NUMCLS, NNSUR)%VDEPN
RADEPN(NCLS, NNSUR)	RN1_NC1(NUMCLS, NNSUR)%RADEPN
RVDEPN(NCLS, NNSUR)	RN1_NC1(NUMCLS, NNSUR)%RVDEPN
RGAPO(NCND, NCLS)	RN1_NC3(NCNODE, NUMCLS)%RGAPO
TGAPO(NCND, NCLS)	RN1_NC3(NCNODE, NUMCLS)%TGAPO
RGAPN(NCND, NCLS)	RN1_NC3(NCNODE, NUMCLS)%RGAPN
TGAPN(NCND, NCLS)	RN1_NC3(NCNODE, NUMCLS)%TGAPN

subroutine

<> subroutine RN1DBD

```
SUBROUTINE RN1DBD (IRRN1, XRN1, IIRN1, IRN1, IRRN2, XRN2, IIRN2, &
IRN2, IRCVH, XCV, IICVH, ICV, IRHS, XHS, IDHS, DHS, IHS, &
IRSC, XSC, IISC, ISC)
. . . . .
!
!   CHEMISTRY TRANSFER PACKAGE
!
CALL RN1RN5 (NNVOL, NUMSEC, NUMCLS, NNSUR, XRN1 (IT1AGN - IR), &
XRN1 (IT1VGN - IR), XRN1 (IT1ALN - IR), XRN1 (IT1VLN - IR), &
XRN1 (IR1AGN - IR), XRN1 (IR1VGN - IR), XRN1 (IR1ALN - IR), &
XRN1 (IR1VLN - IR), XRN1 (ITADPN - IR), XRN1 (ITVDPN - IR), &
XRN1 (IRADPN - IR), XRN1 (IRVDPN - IR), XRN1 (IRAPLN - IRS), &
XRN1 (IRVPLN - IRS), XRN1 (ITAPLN - IRS), XRN1 (ITVPLN - IRS), &
TIME, DT, XCV (KCVV00 + IC), IRN1 (ITICLF), XRN1 (ITXCLF), &
IRN1 (ITPCLF), IRN1 (ITICLT), XRN1 (ITXCLT), IRN1 (ITPCLT), &
XRN1 (ITXTRN), IRN1 (ITPTRN), IRN1 (ITVOL), IRN1 (ITPVOL), &
IRN1 (ITSTTF), IRN1 (ITSTTT), IRN1 (ITSTTW), IRN1 (ITPSTT), &
IRN1 (ITCFM), IRN1 (ITCFE), IRN1 (ITNUM), NTRN, NNTRN, NNTRFP, &
NNTRTP, NNTRRN, NNTRVP, NNTSTP, IRN1 (IPDSUR), XCV (KDELE), &
NUMHS, NNUMHS, IHS (NHSNUM), XHS (NHSM LN - IH), DHS ( (NETOTO + &
IH + 1) / 2), XHS (NHSECN - IH), ISC (IP18), ISC (IP19), XSC ( &
IPS53), XSC (IPS60) )
. . . . .
RETURN
END SUBROUTINE RN1DBD
```

<> subroutine RN1DBD

```
!
!   SUBROUTINE RN1DBD (IRRN1, XRN1, IIRN1, IRN1, IRRN2, XRN2, IIRN2, &
!   IRN2, IRCVH, XCV, IICVH, ICV, IRHS, XHS, IDHS, DHS, IHS, &
!   IRSC, XSC, IISC, ISC)
!! NOTICE: Converted by MELtoMID
SUBROUTINE RN1DBD (
IRRN2, XRN2, IIRN2, &
IRN2, IRCVH, XCV, IICVH, ICV, IRHS, XHS, IDHS, DHS, IHS, &
IRSC, XSC, IISC, ISC)
!
!   USE RN1_MDL
!   . . . . .
!
!   CHEMISTRY TRANSFER PACKAGE
!
!! NOTICE: Converted by hand
!
CALL RN1RN5 (NNVOL, NUMSEC, NUMCLS, NNSUR, XRN1 (IT1AGN - IR), &
XRN1 (IT1VGN - IR), XRN1 (IT1ALN - IR), XRN1 (IT1VLN - IR), &
XRN1 (IR1AGN - IR), XRN1 (IR1VGN - IR), XRN1 (IR1ALN - IR), &
XRN1 (IR1VLN - IR), XRN1 (ITADPN - IR), XRN1 (ITVDPN - IR), &
XRN1 (IRADPN - IR), XRN1 (IRVDPN - IR), XRN1 (IRAPLN - IRS), &
XRN1 (IRVPLN - IRS), XRN1 (ITAPLN - IRS), XRN1 (ITVPLN - IRS), &
TIME, DT, XCV (KCVV00 + IC), IRN1 (ITICLF), XRN1 (ITXCLF), &
IRN1 (ITPCLF), IRN1 (ITICLT), XRN1 (ITXCLT), IRN1 (ITPCLT), &
XRN1 (ITXTRN), IRN1 (ITPTRN), IRN1 (ITVOL), IRN1 (ITPVOL), &
IRN1 (ITSTTF), IRN1 (ITSTTT), IRN1 (ITSTTW), IRN1 (ITPSTT), &
```

```

!   IRN1 (ITCFM), IRN1 (ITCFE), IRN1 (ITNUM), NTRN, NNTRN, NNTRFP,   &
!   NNTRTP, NNTRRN, NNTRVP, NNTSTP, IRN1 (IPDSUR), XCV (KDELE),     &
!   NUMHS, NNUMHS, IHS (NHSNUM), XHS (NHSMLN - IH), DHS ( (NETOTO + &
!   IH + 1) / 2), XHS (NHSECN - IH), ISC (IPI8), ISC (IPI9), XSC (   &
!   IPS53), XSC (IPS60) )
!! NOTICE: Converted by MELtoMID
CALL RN1RN5 (NNVOL,                                               &
            DT, XCV (KCVVOO + IC),                                &
                                     XCV (KDELE),                 &
            NUMHS, NNUMHS, IHS (NHSNUM), XHS (NHSMLN - IH), DHS ((NETOTO + &
            IH+ 1) / 2), XHS (NHSECN - IH), ISC (IPI8), ISC (IPI9), XSC ( &
            IPS53), XSC (IPS60) )
. . . . .
!
RETURN
END SUBROUTINE RN1R5

```

<> subroutine RN1RN5

```

SUBROUTINE RN1RN5 (NVOL, NSEC, NCLS, NSUR, AER1GN, VAP1GN, AER1LN,&
VAP1LN, RDA1GN, RDV1GN, RDA1LN, RDV1LN, ADEPN, VDEPN, RADEPN,   &
RVDEPN, RAPOLN, RVPOLN, TAPOLN, TVPOLN, TIME, DELT, CVOL, ITICLF, &
XTCLF, ITPCLF, ITICLT, XTCLT, ITPCLT, XTTRN, ITPTRN, ITVOL,     &
ITPVOL, ITSTTF, ITSTTT, ITSTTW, ITPSTT, ITCFM, ITCFE, ITNUM, NTRN,&
NNTRN, NNTRFP, NNTRTP, NNTRRN, NNTRVP, NNTSTP, IDSUR, DELE, NUMHS,&
NNUMHS, IHSNUM, HSMULT, ETOTO, HSECMN, NHS, IERRX, HSETRN, RADF)
. . . . .

IF (NX.EQ.1) THEN
    IV1 = 1
    IST1 = 1
    ICF1 = 1
    ICT1 = 1
    ITR1 = 1
ELSE
    IV1 = ITPVOL (NX - 1) + 1
    IST1 = ITPSTT (NX - 1) + 1
    ICF1 = ITPCLF (NX - 1) + 1
    ICT1 = ITPCLT (NX - 1) + 1
    ITR1 = ITPTRN (NX - 1) + 1
ENDIF
IV2 = ITPVOL (NX)
IST2 = ITPSTT (NX)
ICF2 = ITPCLF (NX)
ICT2 = ITPCLT (NX)
ITR2 = ITPTRN (NX)
NUMT = ICT2 - ICT1 + 1
!
. . . . .

RETURN
END SUBROUTINE RN1RN5

```

<> subroutine RN1RN5

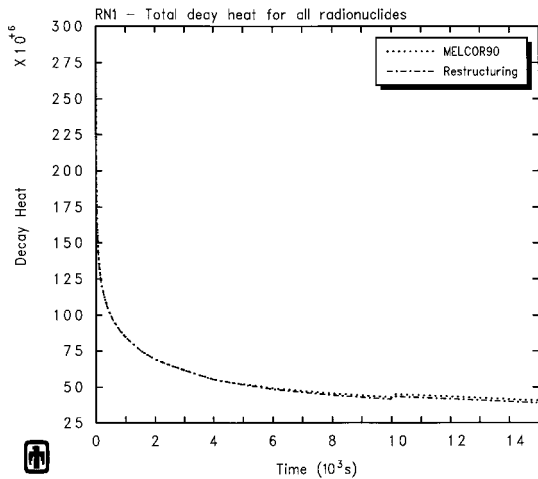
```

SUBROUTINE RN1RN5 (NVOL, NSEC, NCLS, NSUR, AER1GN, VAP1GN, AER1LN,&
! VAP1LN, RDA1GN, RDV1GN, RDA1LN, RDV1LN, ADEPN, VDEPN, RADEPN, &
! RVDEPN, RAPOLN, RVPOLN, TAPOLN, TVPOLN, TIME, DELT, CVOL, ITICLF, &
! XTCLF, ITPCLF, ITICLT, XTCLT, ITPCLT, XTTRN, ITPTRN, ITVOL, &
! ITPVOL, ITSTTF, ITSTTT, ITSTTW, ITPSTT, ITCFM, ITCFE, ITNUM, NTRN,&
! NNTRN, NNTRFP, NNTRTP, NNTRRN, NNTRVP, NNTSTP, IDSUR, DELE, NUMHS,&
! NNUMHS, IHSNUM, HSMULT, ETOTO, HSECMN, NHS, IERRX, HSETRN, RADF)
!! NOTICE: Converted by MELtoMID
SUBROUTINE RN1RN5 (NVOL, DELT, CVOL, &
DELE, NUMHS,&
NNUMHS, IHSNUM, HSMULT, ETOTO, HSECMN, NHS, IERRX, HSETRN, RADF)
USE RN1_MDL
. . . . .
! GET INDICES
IF (NX.EQ.1) THEN
IV1 = 1
IST1 = 1
ICF1 = 1
ICT1 = 1
ITR1 = 1
ELSE
!! NOTICE: Converted by MELtoMID
! IV1 = ITPVOL (NX - 1) + 1
IV1 = RN1_TRO(NX - 1)%ITPVOL + 1
!! NOTICE: Converted by MELtoMID
! IST1 = ITPSTT (NX - 1) + 1
IST1 = RN1_TRO(NX - 1)%ITPSTT + 1
!! NOTICE: Converted by MELtoMID
! ICF1 = ITPCLF (NX - 1) + 1
ICF1 = RN1_TRO(NX - 1)%ITPCLF + 1
!! NOTICE: Converted by MELtoMID
! ICT1 = ITPCLT (NX - 1) + 1
ICT1 = RN1_TRO(NX - 1)%ITPCLT + 1
!! NOTICE: Converted by MELtoMID
! ITR1 = ITPTRN (NX - 1) + 1
ITR1 = RN1_TRO(NX - 1)%ITPTRN + 1
ENDIF
!! NOTICE: Converted by MELtoMID
! IV2 = ITPVOL (NX)
IV2 = RN1_TRO(NX)%ITPVOL
!! NOTICE: Converted by MELtoMID
! IST2 = ITPSTT (NX)
IST2 = RN1_TRO(NX)%ITPSTT
!! NOTICE: Converted by MELtoMID
! ICF2 = ITPCLF (NX)
ICF2 = RN1_TRO(NX)%ITPCLF
!! NOTICE: Converted by MELtoMID
! ICT2 = ITPCLT (NX)
ICT2 = RN1_TRO(NX)%ITPCLT
!! NOTICE: Converted by MELtoMID
! ITR2 = ITPTRN (NX)
ITR2 = RN1_TRO(NX)%ITPTRN
NUMT = ICT2 - ICT1 + 1
!
. . . . .
RETURN
END SUBROUTINE RN1RN5
```

RN1 data RN1 CVH(1 subroutine), FL(1 subroutine), RN1  
 (9 subroutine) SPR (3 subroutine),  
 subroutine

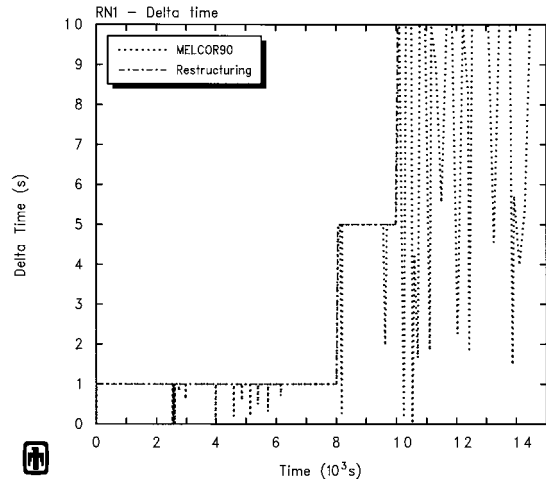
4.

FORTRAN90  
 RN1 FORTRAN77 MELCOR FORTRAN90  
 library execution file FORTRAN90  
 MELCOR RN1MELCOR RN1  
 RN1 pool scrubbing, containment  
 failure 가 , flow path convection  
 RN1 convection switch 4300MW<sub>t</sub> BWR type  
 , RN 가 RN1000 card  
 15,000  
 MELGEN MELCOR subroutine data  
 file(restart file) read/write . MELGEN restart file  
 write가 , MELCOR data file(restart file) read ,  
 RN1  
 Unformatted print 가 subroutine RN1  
 file write UNIX diff command text file . text data  
 file(restart file) read/write  
 , FORTRAN90 MELCOR  
 RN1MELCOR RN1 , 10  
 11



EXERCISE 6  
 WCIWEDNQL 9/23/03 22:46:36 MELCOR ULTRIX

10.



EXERCISE 6  
 WCIWEDNQL 9/23/03 22:46:36 MELCOR ULTRIX

11.

(dt)

10 6000 (delta time)  
 11  
 2300 8000 10000  
 가 ,  
 RN1 CPU time  
 / even/odd cycle array

5.

MIDAS MELCOR  
 RN1 , RN1  
 RN1 , RN1  
 subroutine , RN1  
 , graph ,  
 BH, COR, ESF, HS double precision single  
 precision ,  
 (dt) (dt)  
 (dt) , 가  
 graph , 가  
 , ,  
 set , benchmarking , problem  
 / , MELCOR code  
 RN1 ,  
 가 ,  
 RN1  
 [11],  
 가 가

[1] A Multi-Dimensional Thermal-Hydraulic System Analysis Code, MARS 1.3.1, Vol.31, Number 3, pp.344-363, June 1998.  
 [2] Realistic Thermal-Hydraulic System Code Development Workshop, '98 Fall KNS Conference, 1998.  
 [3] 3<sup>rd</sup> MARS Users Group Meeting, '99 Dec. 2 KAIST Conference, 1999.  
 [4] , , , Development of a Computer Program for Automatic Variable Conversion in MELCOR Code, , , 2000

- [5] , , MIDAS , KAERI/TR-2220/2002, Jun, 2002.
- [6] , , , A Restructuring Proposal for MIDAS, , 2000.
- [7] S.H.Park, H.D.Kim, D.H.Kim, Y.M.Song, B.D.Chung, Development of Restructuring Template for MELCOR, 5<sup>th</sup> PSAM Conference, Japan, 2000.11.26 – 2000.12.2
- [8] , , , A Restructuring of TF package for MIDAS Computer Code, , 2002.
- [9] , , MELCOR , KAERI/TR-1543/00, June, 2000.
- [10] , , , MELCOR , KAERI/TR-1536/2000, March, 2000
- [11] , 12 , , KAERI/RR-2216/2001, pp 302-321, May, 2002.