ORINC

R. PORINC
The RELAP4/MOD5 Interface to ORINC

Steven B. Cliff

Prepared for the U.S. Nuclear Regulatory Commission
Office of Nuclear Regulatory Research
Agreements DOE 40-551-75 and 40-5

Contract No. W-7405-eng-26

Computer Sciences Division

RLPORINC

THE RELAP4/MOD5 INTERFACE TO ORINC

Steven B. Cliff

NOTICE
This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Department of Energy, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

Manuscript Completed - January 10, 1978
Date Published - April, 1978

Prepared for the
U.S. Nuclear Regulatory Commission
Office of Nuclear Regulatory Research
Under Interagency Agreements DOE 40-551-75 and 40-552-75

UNION CARBIDE CORPORATION, NUCLEAR DIVISION
operating the
Oak Ridge Gaseous Diffusion Plant • Oak Ridge National Laboratory
Oak Ridge Y-12 Plant • Paducah Gaseous Diffusion Plant
for the
DEPARTMENT OF ENERGY

TABLE OF CONTENTS

	Page
ACKNOWLEDGMENTS	v
ABSTRACT	1
PROGRAMMING MODIFICATIONS	1
REFERENCES	5
APPENDIX	7

ACKNOWLEDGMENTS

The support for this work came from the ORNL-PWR Blowdown Heat Transfer Separate-Effects Program of the Engineering Technology Division .

RLPORINC

The RELAP4/MOD5 Interface to ORINC

Steven B. Cliff

ABSTRACT

The Oak Ridge Inverse code, ORINC [1], calculates the temperature distribution in an electric core pin from internal temperatures and the conditions in the surrounding environment. While the internal temperatures can be obtained from thermocouples in the Thermal Hydraulic Test Facility [2], the needed environmental conditions cannot be determined as directly from experimental data. However, this information can be determined from a modified version of the Reactor Linearized Analysis Program, RELAP. In particular, the IBM form of RELAP/MOD5 UPDATE2 [3], as released by the Idaho National Engineering Laboratory, was modified to save the proper information in a convenient format. This document describes this RELAP-to-ORINC interface version of RELAP4 which has been named RLPORINC.

PROGRAMMING MODIFICATIONS

The method chosen to implement this interface was to add a subroutine (EDSAVE) which is invoked at every minor edit to save the desired information. The modifications to parts of RELAP4 itself are localized to three routines: MAIN, EDIT and TRAN. In all cases, sufficient comments were added to make the modifications self-documenting. The Appendix is a compiled listing of these three routines.

MAIN - was modified so the first line of every page of RLPORINC contains, in addition to the normal RELAP4/MOD5 messages, the phrase "(ORNL - BDHT: ORINC)". Further, the configuration

BLANK PAGE

control is noted as "NO". Thus, output produced by RLPORINC is always identifiable as such.

EDIT - was modified to invoke EDSAVE at every minor edit. Thus, normal RELAP4 minor edit controls are used to time the saving of the ORINC data.

TRAN - was modified to always produce a minor edit at time zero, and ORINC requires these initial conditions.

No other modifications were made. Hence, all calculations are program actions and should be unaltered. The Appendix has a source listing of the modified routines as well as EDSAVE.

EDSAVE - receives as an argument the number of core sections from EDIT and other variables from COMMON blocks CORPOW, JUNER, JONES, SLAB, SLABAT, SLBGM, STATUS, VOLED, and VOLER. ORINC needs information only for volumes that are to the right of heater slabs which are also core sections. A one-to-one correspondence is assumed between these heater slabs and appropriate core sections. The saved information is discussed below, and all variables are REAL*4 unless otherwise noted.

The current transient time is the first information and it is followed by an integer, the number of slabs that are core sections, NSLAB. The remainder of the information is arrays that have fixed length of 50 words but only the first NSLAB entries are defined.

The first array contains the volume numbers of the volumes under consideration, and is typed INTEGER. Next are the average pressure, temperature, and quality for these volumes.

The surface heat flux, surface heat transfer coefficient, and surface temperature of the right side of the core sections follow.

All the data saved by EDSAVE is obtained by retrieval from RELAP storage areas without modifications except the mass flow rate which is calculated by:

$$\text{MASS FLUX} = \frac{\left| \frac{\text{Weight Flow In}}{\text{Area of Inlet}} + \frac{\text{Weight Flow Out}}{\text{Area of Outlet}} \right|}{2}$$

Next, the saturation temperature of volumes is obtained. Last is the heat transfer mode flag of the right side of the slab. These flags are integers.

EDSAVE gathers and calculates the required information, writes it without format in fixed length records of 502 words, and returns to EDIT. Table I is a summary of the contents of each record.

The changes to the Job Control Language (JCL) statements to execute RLPORINC include, of course, the specification of the saved data set. It is written on FORTRAN unit 20 and typically has a data control block of (RECFM=VS,LRECL=2008,BLKSIZE=2012). Additionally, more space must be allowed for the system buffer areas associated with this data set by specifying a PARM field such as "16,400" to RLPORINC, because of the dynamic memory allocation system used by RELAP4. One extra EXCP or IO is generated per minor edit by EDSAVE and, exclusive of the above buffers, less than 3K bytes of additional storage is needed.

Table I
Format of the RLPORINC Interface Records

Word	Type	Name	Comments
1	REAL*4	TIMEX	Time of this record.
2	INTEGER		Number of core sections with volumes on the right side.
3-52	INTEGER	IVSR	Numbers of these volumes.
53-102	REAL*4	P	Pressures in these volumes.
103-152	REAL*4	TEMP	Temperatures of these volumes.
153-202	REAL*4	AVEX	Average quality of the volumes.
203-252	REAL*4	PHIR	Surface heat flux of the corresponding slab.
253-302	REAL*4	HTCR	Slab's heat transfer coefficient.
303-352	REAL*4	TP	Slab's surface temperature.
353-402	REAL*4		Mass flow rate as calculated.
403-452	REAL*4	SATT	Saturation temperature of the volumes.
453-502	INTEGER	IHTR	Heat transfer mode flag.

REFERENCES

1. R. A. Hedrick and L. J. Ott, *ORINC - A One-Dimensional Implicit Approach to the Inverse Heat Conduction Problem*, ORNL/NUREG-23 (November 1977).
2. *Project Description ORNL-PWR Blowdown Heat Transfer Separate-Effects Program - Thermal Hydraulic Test Facility (THTF)*, ORNL/NUREG/TM-2 (February 1976).
3. *RELAP4/MOD5 - A Computer Program for the Transient Thermal-Hydraulic Analysis of Nuclear Reactors and Related Systems User's Manual*, ANCR/NUREG-1335 (September 1976).

APPENDIX

A LISTING OF THOSE ROUTINES
REQUIRED TO MAKE RELPORINC
FROM RELAP4/MOD5, UPDATE 2

LEVEL 21.8 (JUN 74)

05/360 FORTRAN M

DATE 78.005/08.36.74

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=60,SIZE=0000K,
SOURCE,EBCD=C,NULIST,NODECK,LOAD,NOMAP,NODEIT,NJIS,NOMREF

```

C THIS IS A VERSION OF RELAP4 MOD 5 UPDATE 2 WHICH HAS BEEN MODIFIED
C TO SAVE THE INFORMATION NEEDED BY THE DRINC CODE OF THE JAK RIDGE
C NATIONAL LABORATORY
C MODIFICATIONS COMPLETED ON NOVEMBER 15, 1976 AT ORGDP BY STEVEN B.
C CLIFF OF THE COMPUTING APPLICATIONS DEPARTMENT FOR THE JAK RIDGE
C NATIONAL LABORATORY-BLOWDOWN HEAT TRANSFER (ORNL-BDHT) PROJECT
C FOR THE NUCLEAR REGULATORY COMMISSION
C SEE ORNL-TM
C CHANGES TO THIS, THE MAIN, ROUTINE: (1) ADDITION OF THIS BLOCK OF (36)
C COMMENTS
C (2) REPLACEMENT OF TWO STATEMENTS (RMAN0038,RMAN0045) WITH TWO
C STATEMENTS WHICH CHANGE THE
C TITLE ON EACH PAGE OF RELAP
C OUTPUT
C CHANGES TO SUBROUTINE EDIT: (1) ADDITION OF (OR) APPROPRIATE COMMENTS
C (2) ADDITION OF ONE STATEMENT TO CALL
C EOSAVE WITH EACH MINOR EDIT
C CHANGES TO SUBROUTINE TRAN: (1) ADDITION OF (OR) APPROPRIATE COMMENTS
C (2) REPLACEMENT OF ONE STATEMENT
C (TRAN0950) WITH ONE WHICH FORCES A
C MINOR EDIT AT TIME ZERO
C ADDITION OF SUBROUTINE EOSAVE WRITTEN BY S.B. CLIFF
C A PARAM FIELD OF '16,400' SHOULD BE USED WHEN RUNNING THIS PROGRAM
C TO ALLOW THE SYSTEM THE EXTRA SPACE IT NEEDS FOR UNIT 20
C NO CHANGE WAS MADE TO THE OVERLAY STRUCTURE AND EOSAVE IS IN THE
C ROOT SEGMENT; IT COULD BE PLACED IN THE TRANSIENT SEGMENT
C A TOTAL OF 3 RELAP RECORDS DROPPED; TOTAL OF 56 BDHT RECORDS ADDED
C
C PROGRAM RELAP4 (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
C * TAPE2,TAPE3,TAPE4,TAPE15,TAPE16,DEBUG=OUTPUT)
C MAIN ROUTINE FOR RELAP4
C IMPLICIT REAL*8 (A-H,O-Z)
C TIMER001 COMMON BLOCK
C COMMON /ZCLOCK/ ZSUBNM(50),ZTIMEA(50),ZCALL(50),NZSUBC,4XZJB,T
C COMMON / STATUS / OMEGA, PERIOD, POWER, TIMEX,
C 1 PQUITL, PQUITH, TQUITL, TQUITH, NNGO
C LOGICAL NNGO
C PQUITL = LOWER LIMIT ON PRESSURE IN ANY VOLUME
C PQUITH = UPPER LIMIT ON PRESSURE IN ANY VOLUME
C TQUITL = LOWER LIMIT ON TEMPERATURE IN ANY VOLUME
C TQUITH = UPPER LIMIT ON TEMPERATURE IN ANY VOLUME
C DIMENSION ITL(26)
C LOGICAL FIRST, LAST
C DATA FIRST, LAST / .TRUE., .FALSE. /
C FIRST 5 WORDS OF ITL IS USED FOR PLNT TITLE
C DATA ITL /4HRELA,
C 1 4HP4/I, 4H05 0, 4H09/03, 4H/76 , 4H( 2),
C 2 4H , 4H , 4H , 4H , 4H ,
C THE ABOVE RECORD COMMENTED OUT FOR ORNL-BDHT; DRINC
C ADD PHRASE '(ORNL-BDHT; DRINC)' TO HEADER OF ALL PAGES
C 3 4H (OR,4HNL-H,4HBDHT;,4H OR,4HNC)
C
C 3 4HRELA, 4HP4 T, 4HHERM, 4HAL M, 4HYDIA,

```

```

BDHT 01
BDHT 02
BDHT 03
BDHT 04
BDHT 05
BDHT 06
BDHT 07
BDHT 08
BDHT 09
BDHT 10
BDHT 11
BDHT 12
BDHT 13
BDHT 14
BDHT 15
BDHT 16
BDHT 17
BDHT 18
BDHT 19
BDHT 20
BDHT 21
BDHT 22
BDHT 23
BDHT 24
BDHT 25
BDHT 26
BDHT 27
BDHT 28
BDHT 29
BDHT 30
BDHT 31
CUC 0005
COC 0006
RMAN0010
IRM 0015
RMAN0016
RMAN0017
RMAN0018
RMAN0019
RMAN0025
RMAN0026
RMAN0027
RMAN0028
RMAN0029
RMAN0030
RMAN0031
RMAN0033
RMAN0035
RMAN0036
RMAN0037
RMAN0038
BDHT 32
BDHT 33
BDHT 34
BDHT 35
RMAN0039

```

```

      4      4HULIC, 4H COO, 4HE , 4H CO, 4HNFIC,
      5      4HURAT, 4HION, 4H CON, 4HNTROL,
      6      4H:YES/
C THE ABOVE RECORD COMMENTED OUT FOR ORNL-BDHT: DRINC
C DRUP CONFIGURATION CONTROL
      8      4H:NO /
C
CTIMER005 INITIAL BLOCK
ISN 0010 DATA NZSUBN/ 1/
ISN 0011 CALL TIMSET
C CALL TIMSET(0)
ISN 0012 IZCALL(NZSUBN) = IZCALL(NZSUBN) + 1
ISN 0013 NZSUBS = NZSUBC
ISN 0014 NZSUBC = NZSUBN
ISN 0015 NCASE = 0
ISN 0016 NOGO = .FALSE.
ISN 0017 10 CALL INPUT(NCASE,FRST,LAST,ITL)
ISN 0018 IF(LAST) GO TO 20
ISN 0020 CALL TRAN
CTIMER002 FINAL BLOCK
ISN 0021 NZSUBC = NZSUBS
ISN 0022 CALL ZTEDIT(0)
ISN 0023 CALL ETSCON
ISN 0024 REWIND 2
ISN 0025 GO TO 10
ISN 0026 20 CONTINUE
ISN 0027 STOP
ISN 0028 END

*OPTIONS IN EFFECT* NAME= MAIN,OPT=02,LINECNT=60,SIZE=0000K,
*OPTIONS IN EFFECT* SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,NOLIO,NJREF
*STATISTICS* SOURCE STATEMENTS = 27 ,PROGRAM SIZE = 484
*STATISTICS* NO DIAGNOSTICS GENERATED
***** END OF COMPILATION *****
303K BYTES OF CORE NOT USED

```

RMAN0041
 RMAN0043
 RMAN0045
 BDHT 36
 BDHT 37
 BDHT 38
 BDHT 39
 RMAN0047
 RMAN0049
 IBM 0051
 CDC 0052
 RMAN0053
 RMAN0055
 RMAN0057
 RMAN0059
 RMAN0061
 RMAN0070
 RMAN0080
 RMAN0090
 RMAN0095
 RMAN0096
 RMAN0097
 RMAN0098
 RMAN0099
 RMAN0100
 RMAN0106
 RMAN0107
 RMAN0110

LEVEL 21.3 (JUN 74)

05/360 FORTRAN H

DATE 78.005/08.34.56

```

      COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                        SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NODEIT,NODID,NOXREF
ISN 0002      SURROUTINE EDIT (NVOL, NJUN, NPMP, NSLB, NCOR, NHTX,
      1 MAJC, MINC, DMPC, PLTC, LABL, IEMHT, IEMPS, IEMEC,DT)
      C
      C ORNL-BDHT MODS TO INVOKE EDSAVE AT MINOR EDITS:EDITJ105
      C COMPLETED NOVEMBER 15,1976, BY STEVEN B. CLIFF
ISN 0003      IMPLICIT REAL*8 (A-H,O-Z)
      C
ISN 0004      CTIMER002 COMMON BLOCK
      COMMON /ZCLOCK/ ZSUBNM(50),ZTIMEA(50),IZCALL(50),NZSUBC,MAXZUR
      * ,TTTTT
      C
      C NVOL = NUMBER OF VOLUMES
      C NJUN = NUMBER OF JUNCTIONS
      C NPMP = NUMBER OF PUMPS
      C NSLB = NUMBER OF HEAT SLABS
      C NCOR = NUMBER OF CORE SECTIONS
      C NHTX = NUMBER OF HEAT EXCHANGERS
      C IEMFG = EVALUATION MODEL FLAG
      C MAJC = IS A MAJOR EDIT WANTED
      C MINC = IS A MINOR EDIT WANTED
      C DMPC = IS A RESTART DUMP WANTED
      C PLTC = IS A PLOT RECORD WANTED
      C LABL = IS A TRAILER LABEL WANTED
ISN 0005      COMMON /BALED / AE, AMASS, BMASSW, BMASSA, DMASSW, DMASSA, EB,
      * FE, HE, PNORM, POWRI, QLOSS, UFILL, ULOSS
      C
      C AE = ENERGY ADDED
      C AMASS = MASS ADDED
      C BMASSW = MASS OF WATER
      C BMASS = MASS OF AIR
      C DMASSW = MASS OF WATER LEAKED
      C DMASSA = MASS OF AIR LEAKED
      C EB = ENERGY BALANCE
      C FE = ENERGY IN FUEL
      C HE = ENERGY EXTRACTED BY HEAT EXCHANGER
      C PNORM = NORMALIZED REACTOR POWER
      C POWRI = INITIAL REACTOR POWER
      C QLOSS = TOTAL RATE OF HEAT REMOVAL
      C UFILL = ENERGY FROM FILLS
      C ULOSS = ENERGY LEAKED
ISN 0006      COMMON / CHKVER / CV1(10), CV2(10), CV3(10), PCV(10),
      1 THDOT(100), THETA(100), IACV(10), ITCV(10), OPEN(100), LATCH(10)
ISN 0007      LOGICAL OPEN
      C
      C CV1 = FORWARD FLOW FRICTION FACTOR
      C OR DOOR AREA TIMES MOMENT ARM
      C CV2 = REVERSE FLOW FRICTION FACTOR, VALVE OPEN
      C OR DOOR MOMENT OF INERTIA
      C CV3 = REVERSE FLOW FRICTION FACTOR, VALVE CLOSED
      C OR DOOR DAMPING CONSTANT
      C PCV = BACK PRESSURE TO CLOSE CHECK VALVE
      C THDOT = DOOR ANGULAR VELOCITY
      C THETA = DOOR ANGLE

```

```

FOIT0010
FOIT0020
FOIT0030
RDHT 40
RDHT 41
RDHT 42
RDHT 43
IBM 0040
FOIT0045
FOIT0046
FOIT0047
FOIT0050
FOIT0060
FOIT0070
FOIT0075
FOIT0080
FOIT0090
FOIT0100
FOIT0105
FOIT0110
FOIT0120
FOIT0130
FOIT0140
FOIT0150
FOIT0160
FOIT0170
FOIT0180
FOIT0190
FOIT0200
FOIT0210
FOIT0220
FOIT0225
FOIT0230
FOIT0235
FOIT0240
FOIT0250
FOIT0260
FOIT0270
FOIT0280
FOIT0290
FOIT0300
FOIT0310
FOIT0320
FOIT0322
FOIT0324
FOIT0326
FOIT0328
FOIT0330
FOIT0332
FOIT0334
FOIT0336
FOIT0338
FOIT0340
FOIT0342
FOIT0344
FOIT0346

```

```

C      IACV = INDEX FOR AREA VS ANGLE TABLE
C      ITCV = TYPE OF CHECK VALVE
C      OPEN = .TRUE. MEANS CHECK VALVE IN OPEN POSITION
C
1SN 0008      COMMON /CONTR/ FILID(30), INDEX(30), FILSIZ(30)
1SN 0009      INTEGER FILSIZ
C      CONTR HOLDS INFORMATION OF DYNAMIC ARRAYS.
C      FILID = FTS NUMBER OF ARRAY, ZERO MEANS ARRAY NOT BEING USED.
C      NEGATIVE VALUE MEANS ARRAY NEED NOT BE SAVED FOR RESTART.
C      INDEX = POSITION OF FIRST WORD OF ARRAY.
C      FILSIZ = LENGTH OF ARRAY, POS. IF IN /FAST/, NEG. IF IN /FTSLCM/.
C      CONTR(1) IMPLIES FILID(1), INDEX(1), AND FILSIZ(1).
C      CONTR(1) = INPUT DATA IN INPUT PHASE AND SCRATCH IN TRAN PHASE.
C      CONTR(2) = ICE CONDENSER ARRAYS
C      CONTR(3) = FILL DATA
C      CONTR(4) = WATER PROPERTY DATA
C
1SN 0010      COMMON / CORNW / CLTI(50), CTR(50), CTRL(50), RRO(50), MWREAC
1SN 0011      LOGICAL MWREAC
C
C      CLTI = INITIAL CLAD THICKNESS, FT
C      CTR = DEPTH REACTION HAS PENETRATED CLADDING AT END OF ANY TIME
C      STEP, FT
C      CTRL = DEPTH REACTION HAS PENETRATED CLADDING AT START OF ANY TIME
C      STEP, FT
C      RRO = ORIGINAL FUEL PIN RADIUS, FT
C      MWREAC = .TRUE. INDICATES METAL-WATER REACTION MODEL DESIRED
C
1SN 0012      COMMON / CORPOW / POW, FQ(50), LAMBDA, MQ(50), PPOW, PPRMPT,
1          QDMOD(50), QFRAC(50), QN(21), QPMOD(50), SE(50), TAJ, TM(50),
1SN 0013      2 TS(50), VW(50), ISLB(50), IVOL(50)
C      REAL*8 LAMBDA, MQ
C      REAL LAMBDA, MQ
C
C      FQ = NUCLEAR POWER GENERATION IN CORE HEAT SLAB, MW
C      QFRAC = FRACTION OF POWER IN CORE SECTION
C      QN = HEAT GENERATION RATE, BTU/FT**3-SEC
C      TM = AVERAGE TEMPERATURE OF CORE HEAT SLAB
C      TS = SURFACE TEMPERATURE OF CORE HEAT SLAB
C      VW = WATER SPECIFIC VOLUME
C      ISLB = HEAT SLAB NUMBER
C
1SN 0014      COMMON / CSLBQC / AZ(100), AZI(100), BZ(100), CZ(100),
1          EZ(100), EZI(100), FZ(100), HZ(100), WED(100), NODT(4,50)
C
C      AZ - HZ = COEFFICIENTS FOR CRITICAL HEAT FLUX CORRELATIONS
C      WED = WETTED EQUIVALENT DIAMETER, IN
C      NODT = MAJOR EDIT CONTROL FOR CORR NODE TEMPERATURES
C
1SN 0015      COMMON / EDITER / FOUT(10,50), IDC, IFLT, IMIN(9,6), JXX,
1          IKFLT(18), NS(9)
C      LEVEL 2,FOUT
C
C      FOUT = FLOATING OUTPUT ARRAY
C      IDC = TAPE DUMP COUNT
C      IFLT = FLOATING EDIT COUNT
C      IMIN = MINOR HEADINGS
C      JXX = MINOR EDIT LINE COUNT

```

```

C      KFLT = FLOATING EDIT INDICES
C      NS   = MINOR EDIT VALUES INDEX
C
ISN 0016 COMMON / FNYAJ / FLOWA(75),FLOWL(75),FLOWRA(75),WVBAR(75),
1 WVHARD(75),SPVZ(75),SPVZO(75),AINERV(75),CMAS(75),DIAMV(75),
2 FANING(75),ELOD(75),TPMV(75),CMACHV(75),JTPMV(75),
3 IFAN(75),JVISC(75),IAMBLO(75),IIN(75),IOUT(75)
C
C      FLOWA = VOLUME FLOW AREA (FT2), INPUT
C      FLOWL = VOLUME FLOW PATH LENGTH (FT)
C      FLOWRA = 1/(144*GC*FLOWA**2)
C      WVBAR = AVERAGE VOLUME FLOW (LBM/SEC)
C      WVHARD = PREVIOUS TIME STEP VALUE OF WVBAR
C      SPVZ = VOLUME AVERAGE SPECIFIC VOLUME (FT3/LBM)
C      SPVZO = PREVIOUS TIME STEP VALUE OF SPVZ
C      AINERV = HALF-VOLUME INERTIA (L/2*A*144*GC)
C      CMAS = CENTER OF MASS HEIGHT
C      DIAMV = EQUIVALENT DIAMETER OF FLOW AREA
C      FANING = FANNING FRICTION FACTOR
C      ELOD = (L/D)/(144*GC*A**2)
C      TPMV = TWO-PHASE MULTIPLIER FOR FANNING FRICTION TERMS
C      CMACHV = MACH NUMBER SQUARED FOR AVERAGE VOLUME FLOW
C      JTPMV = INDEX FOR TWO-PHASE FRICTION 0=NO, 1=YES
C      IFAN = MEMORY INDEX FOR FANNING FRICTION CALCULATION
C      JVISC = MEMORY INDEX FOR VISCOSITY CALCULATION
C      IAMBLO = INDEX FOR DETERMINING WHETHER TO CALCULATE LIQUID LEVE.
C      IIN,IOUT = PRIMARY INLET AND OUTLET JUNCTION NUMBERS
C
C      COMMON BLOCK CONTAINING FAST MEMORY ARRAY FOR DYNAMIC STORAGE.
ISN 0017 COMMON /FAST/ FA(1)
ISN 0018 INTEGER IA(2,1)
C      INTEGER IA(1)
ISN 0019 EQUIVALENCE (FA(1),IA(1,1))
C      EQUIVALENCE (FA(1),IA(1))
C      COMMON BLOCK CONTAINING LCM MEMORY ARRAY FOR DYNAMIC STORAGE.
C      FOR COMPUTERS WITHOUT LCM, THIS ARRAY IS EQUIVALENCED TO FAST
C      ARRAY.
ISN 0020 REAL*8 LFA(1)
C      COMMON /FTBLCM/ LFA(1)
ISN 0021 INTEGER LIA(2,1)
C      INTEGER LIA(1)
C      LEVEL 2, LFA,LIA
ISN 0022 EQUIVALENCE (LFA(1),LIA(1,1),FA(1))
C      EQUIVALENCE (LFA(1),LIA(1))
C
C      ICEVOL -- STRUCTURE OF ICE VOLUME ARRAY
C      NVOLI NUMBER OF ICE VOLUMES
C      VOLNOI VOLUME NUMBER
C      REMNICE REMAINING ICE MASS
C      DELUIN EFFECTIVE HEAT OF FUSION
C      ALSUBC CONDENSATE LAYER LENGTH -- USED AS FIT PARAMETER
C      SURTOM RELATES SURFACE TO MASS
C      HTCICE HEAT TRANSFER COEFFICIENT TO ICE
C      HTRATE HEAT TRANSFER RATE
C      INTICE INITIAL ICE MASS
C      HTCLIM UPPER LIMIT OF HEAT TRANSFER COEFFICIENT
C      C1 CONSTANT TO ADJUST HEAT TRANSFER COEFF. TO ICE
C      C2 CONSTANT TO ADJUST HEAT TRANSFER COEFF. TO ICE

```



```

1      REAL*8 INERTA      SRCOS(100), ZJUN(100), ZVOL(75), KCHOKE(100)      EDIT1190
ISN 0030      REAL INERTA      EDIT11200
C      AJUN = JUNCTION FLOW AREA      EDIT11201
C      ELEV = VOLUME ELEVATION      EDIT11210
C      SRCOS = SQUARE ROOT OF THE COS OF ANGLE BETWEEN VOLUMES CONNECTED      EDIT11220
C      INERTA = JUNCTION INERTIA      EDIT11230
C      ZJUN = JUNCTION ELEVATION      EDIT11240
C      ZVOL = VOLUME HEIGHT      EDIT11250
C      KCHOKE = CHOKING INDICATOR (0=UNCHOKED)      EDIT11260
C      EDIT11270
C      EDIT11280
C      EDIT11290
ISN 0031      COMMON / JUNET / CHKVK(100), DELP(100), PUMPK(100), PUMPP(100)      EDIT11300
C      CHKVK = CHECK VALVE FRICTION FACTOR      EDIT11310
C      DELP = VOLUME TO VOLUME PRESSURE DIFFERENCE      EDIT11320
C      PUMPK = PUMP FRICTION FACTOR      EDIT11330
C      PUMPP = PUMP HEAD      EDIT11332
C      EDIT11333
C      EDIT11335
ISN 0032      COMMON / LEEKE / CONCO(100), SINK(5), TAREA(10,5),
1      TAREA(5), ICHOKE(100), ITLEAK(5), NAREA(5)      EDIT11336
C      CONCO = CONTRACTION COEFFICIENT      EDIT11337
C      SINK = SINK PRESSURE      EDIT11338
C      TAREA = TABLE OF AREA VS TIME      EDIT11339
C      TAREA = CURRENT POSITION IN AREA VS TIME TABLE      EDIT11340
C      ICHOKE = LIQUID PHASE CHOKING CONDITION (0=CHOKED, 1=NOT CHOKED)      EDIT11341
C      ITLEAK = TRIP SIGNAL NUMBER TO OPEN LEAK      EDIT11342
C      NAREA = NUMBER OF AREA VS TIME DATA POINTS      EDIT11343
C      EDIT11344
C      EDIT11345
ISN 0033      COMMON / LIQLEV / VZLVC(10), ZLVC, ILVC(20), ITLVC, NLVC, NTLVC      EDIT11346
C      VZLVC = HEIGHT(1), VOLUME(1), . . . HEIGHT(NTLCV), VOLUME(NTLCV)      EDIT11351
C      ZLVC = EQUIVALENT LIQUID LEVEL      EDIT11352
C      ILVC = VOLUMES TO BE SUMMED      EDIT11353
C      ITLVC = CURRENT TABLE POSITION      EDIT11354
C      NLVC = NUMBER OF VOLUMES TO BE SUMMED      EDIT11355
C      NTLVC = NUMBER OF PAIRS IN VZLVC      EDIT11356
ISN 0034      COMMON / PUMPV / CAVCON(12), FPUMP(12), PFLOWR(12), PHEADR(12),
1      PINRTA(12), POMGA(12), POMGAR(12), PTORKR(12), SPEED(12),
2      SPUM(12), TORKF(4,2), TORQUE(12), VRHO(12),
3      TORKMR(12), IMT(12), IMZ(12),
4      IPC(12), IPM(12), IRP(12), ITPUMP(12), IVPUMP(12),
5      JDISRG(12), JPHD(12), JPHD4(12), JPHM(12), JPTK(12), JPTK4(12),
6      JPTM(12), JSUCTN(12), NUMPMP      EDIT11360
C      SPEED = PUMP SPEED (RPM)      EDIT11361
C      TORQUE = NORMALIZED PUMP TORQUE      EDIT11362
C      IVPUMP = PUMP VOLUME NUMBER      EDIT11364
C      EDIT11366
C      EDIT11368
C      EDIT11370
C      EDIT11372
C      EDIT11374
C      EDIT11376
C      EDIT11378
C      EDIT11380
C      EDIT11381
ISN 0035      COMMON / PAEM / PIN(50), EF(50), EC(50), HCOND(50), TJUMP(50), FRACT(50),
1      AFR(7,50), BLKJUN(100), RROIN(50), CTRIN(50),
2      PITCH, AMASSO, VINPUB, TAB1(50), TAB2(50), TAB3(50), TAB4(50),
3      ISWJUN(100), ITRUPFG(50), IRCPIN, IRGGAP, IRGSW, IACCV(4), ITRIP(4),
4      IDNCMV, IRENV(10), ITP(100), ISLBAJ(50), ISWTAB(50), NTAB1, NTAB2, NTAB3,
5      SNTAB4, IMSPR      EDIT11382
C      LEVEL 2, PIN      EDIT11384
C      EDIT11386
C      EDIT11388
C      EDIT11390
C      EDIT11392
C      EDIT11393

```

ISN 0036	C **	COMMON / R4FRAP / FSWELL(50), EPSMXP(50), GSMOL(50), PINF(50),	EDIT1394
	1	RPIN(50), RSHD(50), VPLEN(50)	EDIT1396
	2	FPEL(50), AS(50), VSP(50), PMOL(50), QPL(50), TPL(5,2,50),	EDIT1399
	3	GAMFPL(50)	EDIT1400
	C	LEVFL 2,FSWELL	EDIT1402
	C		CDC 1404
	C		EDIT1406
	C	RPIN = OUTSIDE FUEL PIN RADIUS	EDIT1408
	C		EDIT1410
ISN 0037	C	COMMON / REACES / RC, RCCAL, RD, RDCAL, RF, RFCAL,	EDIT1412
		RV, RVCAL, RW, RWCAL	EDIT1414
	C		EDIT1416
	C	RC = CONTROL REACTIVITY	EDIT1418
	C	RCCAL = CALCULATED CONTROLLED REACTIVITY AT START	EDIT1420
	C	RD = DOPPLER REACTIVITY	EDIT1422
	C	RDCAL = CALCULATED DOPPLER REACTIVITY AT START	EDIT1424
	C	RF = FUEL REACTIVITY	EDIT1430
	C	RFCAL = CALCULATED FUEL REACTIVITY AT START	EDIT1440
	C	RV = VOID REACTIVITY	EDIT1460
	C	RVCAL = CALCULATED VOID REACTIVITY AT START	EDIT1470
	C	RW = COOLANT REACTIVITY	EDIT1480
	C	RWCAL = CALCULATED COOLANT REACTIVITY AT START	EDIT1490
	C		EDIT1500
ISN 0038	C	COMMON / RNDER / PHI(2, 20), PREAC, SLOPE, SS, SUM, T, W, WJ, KK	EDIT1510
	C	PHI(1,1) = OLD EMITTER CONCENTRATION	EDIT1520
	C	PHI(2,1) = NEW EMITTER CONCENTRATION	EDIT1530
	C	PREAC = TOTAL REACTIVITY	EDIT1540
	C	SLOPE = RATE OF CHANGE OF REACTIVITY	EDIT1550
	C	SS = TIME INDEPENDENT SOURCE	EDIT1560
	C	SUM = DELAYED NEUTRON SOURCE	EDIT1570
	C	T = IREKIN TIME STEP SIZE	EDIT1580
	C	W = AVERAGE RECIPROCAL PERIOD	EDIT1590
	C	WJ = INSTANTANEOUS RECIPROCAL PERIOD	EDIT1600
	C	KK = RUNGA KUTTA STAGE	EDIT1610
	C		EDIT1620
ISN 0039	C	COMMON / RETALS / FJUNF(100), FJUNR(100), HKIN(100), SPVJ(100),	EDIT1621
	1	SPVJO(100), RESOK(100), AJUNT(100), DIAMJ(100), TPMJ(100),	EDIT1622
	2	CMACHJ(100), PBO(100),	EDIT1624
	3	KMACH(100), JVERTL(100), JCHOKE(100), JCALCI(100), MVMIX(100)	EDIT1626
	C		EDIT1628
	C	FJUNF = SPECIFIC ENERGY LOSS COEFFICIENT FOR FORWARD FLOW	EDIT1630
	C	FJUNR = SPECIFIC ENERGY LOSS COEFFICIENT FOR REVERSE FLOW	EDIT1632
	C	HKIN = JUNCTION KINETIC ENERGY, (V*2)/2 BTU/LBM	EDIT1634
	C	SPVJ = JUNCTION SPECIFIC VOLUME, FT3/LBM	EDIT1636
	C	RESOK = RESIDUAL JUNCTION FRICTION FACTOR FOR STEADY STATE	EDIT1638
	C	AJUNT = TIME DEPENDENT JUNCTION AREA	EDIT1640
	C	DIAMJ = JUNCTION DIAMETER	EDIT1642
	C	TPMJ = TWO-PHASE FRICTION MULTIPLIER FOR FORM-LOSS	EDIT1644
	C	CMACHJ = MACH NUMBER SQUARED AT JUNCTION	EDIT1646
	C	KMACH = MACH TABLE MEMORY INDEX	EDIT1648
	C	JVERTL = VERTICAL JUNCTION CONTROL INDEX	EDIT1650
	C	JCHOKE = JUNCTION CHOKING INDEX	EDIT1652
	C	JCALCI = INITIAL CONDITION CALCULATION TYPE CONTROL INDEX	EDIT1654
	C	MVMIX = MOMENTUM MIXING INDEX	EDIT1656
ISN 0040	C	COMMON / SLAB / AHTL(50), AHTR(50), HDML(50), HDNR(50),	EDIT1658
		* CHNL(50), CHNR(50), DHCL(50), DHNR(50), FCHL(50), FCHNR(50),	EDIT1660
			EDIT1662


```

C      DELTM = TIME STEP SIZE                                ED1T2280
C      DTHIN = MINIMUM ALLOWABLE TIME STEP SIZE             ED1T2290
C      TLAST = END OF INTERVAL                               ED1T2300
C      DTOLDS = PREVIOUS TIME STEP                           ED1T2305
C      IDMP = NUMBER OF COMPLETE EDITS SINCE LAST RESTART SAVE ED1T2310
C      IMAJ = NUMBER OF BRIEF EDITS SINCE LAST COMPLETE EDIT ED1T2320
C      IMIN = NUMBER OF TIME STEPS SINCE LAST BRIEF EDIT     ED1T2330
C      IPLT = NUMBER OF TIME STEPS SINCE LAST PLOT SAVE       ED1T2340
C      NCHK = TIME STEP CONTROL OPTION (0, BLANK OR NEG = YES, POS = NO) ED1T2350
C      NDMP = NUMBER OF COMPLETE EDITS PER TAPE SAVE FOR RESTART ED1T2360
C      NMAJ = NUMBER OF BRIEF EDITS PER COMPLETE EDIT         ED1T2370
C      NMIN = NUMBER OF TIME STEPS PER BRIEF EDIT             ED1T2380
C      NSET = NUMBER OF TIME STEPS PER TAPE SAVE OF PLOTS VARIABLES ED1T2390
C      NSET = CURRENT INTERVAL INDEX                           ED1T2400
C      NSTEP = CURRENT TIME STEP NUMBER                       ED1T2410
C      NTSFT = NUMBER OF TIME STEP SETS                       ED1T2420
C      NTSPT = TOTAL NUMBER OF TIME STEPS USED                ED1T2430
C      ENDSTP = .TRUE. INDICATES LAST TIME STEP 1/ INTERVAL REDUCED ED1T2440
C      REDUC = .TRUE. INDICATES PREVIOUS TIME STEP REDUCED   ED1T2450
C
ISN 0050 COMMON / VOLD / AVED(75), AVEX(75), SATP(75), SATY(75),
C          SATVF(75), SATVG(75)                                ED1T2460
C
C          AVED = AVERAGE DENSITY                             ED1T2470
C          AVEX = AVERAGE QUALITY                             ED1T2480
C          SATP = SATURATION PRESSURE                           ED1T2490
C          SATY = SATURATION TEMPERATURE                       ED1T2500
C          SATVF = SATURATION SPECIFIC VOLUME OF LIQUID        ED1T2510
C          SATVG = SATURATION SPECIFIC VOLUME OF GAS           ED1T2520
C
ISN 0051 COMMON / VOLR / A(75), HW(75), P(75), TEMP(75), ZL(75),
C          ZM(75), ZMO(75), IQIN(75), IREAD(75)                ED1T2530
C
C          A = VOLUME AREA                                     ED1T2540
C          HW = VOLUME SPECIFIC ENTHALPY                       ED1T2550
C          P = VOLUME PRESSURE                                 ED1T2560
C          TEMP = VOLUME TEMPERATURE                           ED1T2570
C          ZL = LIQUID LEVEL                                   ED1T2580
C          ZM = MIXTURE LEVEL                                   ED1T2590
C          ZMO = MIXTURE LEVEL FOR PREVIOUS TIME STEP          ED1T2600
C          IQIN = HEAT CALCULATION INDEX                       ED1T2610
C          IREAD = INDEX FOR TIME DEPENDENT VOLUMES            ED1T2620
C
ISN 0052 COMMON / VOLESI / GASM(75), GASV(75), LIQV(75), FMASS(75), MIXV(75),
C          * ARMAS(75), SATHF(75), SATHG(75), U(75), V(75), WPTIME, ED1T2630
C          * IWPAC(100), IWPEDT                                ED1T2640
ISN 0053 REAL*8 MIXV, LIQV                                     ED1T2650
C          REAL MIXV, LIQV                                     ED1T2660
C
C          GASM = GAS MASS (INCLUDES BUBBLE MASS)              ED1T2670
C          GASV = GAS VOLUME                                     ED1T2680
C          LIQV = LIQUID VOLUME                                 ED1T2690
C          FMASS = TOTAL WATER MASS                             ED1T2700
C          MIXV = MIXTURE VOLUME                                ED1T2710
C          ARMAS = AIR MASS                                     ED1T2720
C          SATHF = SPECIFIC ENTHALPY OF SATURATED LIQUID       ED1T2730
C          SATHG = SPECIFIC ENTHALPY OF SATURATED GAS          ED1T2740
C          U = TOTAL ENERGY                                    ED1T2750

```



```

ISN 0168      WRITE (6,650) L, IVSR(L), (NODT(N,J), TNID(N), NFI,NUP)
ISN 0169      150 CONTINUE
C FM
ISN 0170      IF (TEMPS.EQ. 0) GO TO 886
ISN 0172      DO 885 I=1,NJUN
ISN 0173      IF (ISLJUN(I).EQ.0) GO TO 885
ISN 0175      987 J=ABS(15WJUN(I))
ISN 0176      K=ISCO(J)
ISN 0177      WRITE (6,805) J
ISN 0178      805 FORMAT (//39H SWELLING HAS OCCURRED IN HEAT SLAB NO. (3)
ISN 0179      IF (IRUPFG(K).EQ.1) WRITE (6,811) J
ISN 0191      811 FORMAT (36H *** CLAD HAS RUPTURED FOR HEAT SLAB (5,4H ***
ISN 0192      IF (ISLHAJ(J).NE.1) WRITE (6,806) 1,BLKJUN(I)
ISN 0184      806 FORMAT (5X,9H JUNCTION,13,3H 15.E12,5,16H PERCENT BLOCKED)
ISN 0185      IF (ISLHAJ(J).NE.2) WRITE (6,812) RRO(K),CLT((K),IRGSW
ISN 0187      812 FORMAT(/22H SWOLLEN RADIUS ***** E12,5,3H FT/
          5      22H NEW CLAD THICKNESS * C12,5,3H FT/
          5      22H CLAD REGION NO. ***** (3)
ISN 0188      IF (ISLHAJ(J).EQ.2) WRITE (6,814)
ISN 0190      814 FORMAT (/43H SURFACE AREA OF CLAD HAS NOT BEEN ADJUSTED)
ISN 0191      985 CONTINUE
C FM
C C
C C      WRITE MW9 AND OTHER VALUES FOR CORE HEAT SLABS
ISN 0192      896 WRITE (6,660)
ISN 0193      DO 160 J=1,NCOR
ISN 0194      L = ISL(J)
ISN 0195      QMWRT = QMWRT(J) * SLEN(L) * 3600.0
ISN 0196      160 WRITE (6,670) L, IVSR(L), QMWRT, CTR(J), CTRIN(J),
          1      TP(1,L), TN(J), FQ(J)
ISN 0197      165 IF (NMTX.LE.0) GO TO 170
C C
C C      WRITE HEAT EXCHANGER VALUES
ISN 0199      WRITE (6,680) IH(2), IH(24), IH(33), IH(44)
ISN 0200      DO 167 I=1,NVOL
ISN 0201      IF (IQIN(I).GE.0) GO TO 167
ISN 0203      J = -IQIN(I)
ISN 0204      HEAT = -HTO(J)
ISN 0205      WRITE (6,612) I, J, HEAT
ISN 0206      167 CONTINUE
C C
C C      CHECK FOR NEW PAGE
ISN 0207      170 IF (41.LT. NVOL + NCOR + NJUN + NMTX) CALL SUBPG
C C
ISN 0209      WRITE JUNCTION VALUES
          *      WRITE(6,608) IH(27), IH(28), IH(29), IH(11), IH(27), IH(37),
          *      IH(39), IH(48), IH(33), IH(38),
          *      IH(58), IH(59), IH(56), IH(49), IH(29),
          2      IH(49), IH(30), IH(49), IH(31), IH(49), IH(32), IH(49)
ISN 0210      DO 180 I=1,NJUN
ISN 0211      IT=ITP(I)
ISN 0212      IF (KCHOKE(I).EQ.0) IT=1
ISN 0214      IF (KCHOKE(I).EQ.2) IT=4
ISN 0216      180 WRITE (6,609) I,IW(1,1),IW(1,2),KCHOKE(I),CHUKE(I),TYP(IT),WP(I),
          1      MP(1),SPVJ(1),DELP(1),DELE(1),DELA(1),PUMPR(1)
ISN 0217      IF (ISLIP.EQ. 0) GO TO 191
ISN 0218      IF (NJUN+3.GT. LINES(0)) CALL SUBPG
          *      FDI13750
          *      FDI13760
          *      FDI13765
          *      FDI13766
          *      FDI13767
          *      FDI13772
          *      FDI13774
          *      FDI13775
          *      FDI13777
          *      FDI13778
          *      FDI13779
          *      FDI13780
          *      FDI13781
          *      FDI13782
          *      FDI13783
          *      FDI13784
          *      FDI13785
          *      FDI13786
          *      FDI13787
          *      FDI13788
          *      FDI13789
          *      FDI13790
          *      FDI13793
          *      FDI13795
          *      FDI13800
          *      FDI13810
          *      FDI13820
          *      FDI13825
          *      FDI13830
          *      FDI13831
          *      FDI13840
          *      FDI13850
          *      FDI13860
          *      FDI13870
          *      FDI13880
          *      FDI13890
          *      FDI13900
          *      FDI13910
          *      FDI13920
          *      FDI13930
          *      FDI13940
          *      FDI13950
          *      FDI14100
          *      FDI14110
          *      FDI14120
          *      FDI14130
          *      FDI14140
          *      FDI14142
          *      FDI14150
          *      FDI14160
          *      FDI14170
          *      FDI14173
          *      FDI14174
          *      FDI14175
          *      FDI14180
          *      FDI14190
          *      FDI14195
          *      FDI14200

```

```

ISN 0221      WRITE(6,621)
ISN 0222      DO 190 I = 1,NJUN
ISN 0223      IF (MOD(I,50) .NE. 0) GO TO 185
ISN 0225      CALL SUBPG
ISN 0226      WRITE(6,621)
ISN 0227      185  CONTINUE
ISN 0228      IF (XWL(I)) 181,182,183
ISN 0229      181  MLDON = MLOUT(I)
ISN 0230      GO TO 184
ISN 0231      182  MLDON = 0.0
ISN 0232      GO TO 184
ISN 0233      183  MLDON = MLIN(I)
ISN 0234      184  IF (XWG(I)) 186,187,188
ISN 0235      186  MGDON = MGOUT(I)
ISN 0236      GO TO 189
ISN 0237      187  MGDON = 0.0
ISN 0238      GO TO 189
ISN 0239      188  MGDON = MGIN(I)
ISN 0240      189  HWT = 0.0
ISN 0241      IF (WP(I).NE.0.0) HWT = (XWL(I)*MLDON + XWG(I)*MGDON) / WP(I)
ISN 0243      WRITE (6,604) 1,VSLA(I),VLQA(I),VVAPA(I),XWL(I),XWG(I),
      1      MLDON,MGDON,HWT
ISN 0244      190  CONTINUE
ISN 0245      191  CONTINUE
ISN 0246      IF (JC.GT.0) WRITE (6,690) ZLVC, ILVC(I)
ISN 0248      GO TO 30

C
C      MINOR EDIT
ISN 0249      200  JXX = JXX + 1
ISN 0250      FOUT(JXX) = TIMEX

C
C      PICK UP CORRECT MINOR VALUES
ISN 0251      DO 300 I=1,IFLT
ISN 0252      J1 = NS(I)
ISN 0253      J2 = KFLT(2*I)
ISN 0254      J3 = I+1
ISN 0255      GO TO (201, 202, 203, 204, 205, 206, 207, 208, 209, 210,
      1      211, 212, 213, 214, 215, 216, 217, 218, 219, 220,
      2      221, 222, 223, 224, 225, 226, 227, 228, 229, 230,
      3      231, 232, 233, 234, 235, 236, 237, 238, 239, 240,
      4      241, 242, 243, 244, 245, 246, 247, 248, 249, 250,
      5      251, 252, 253, 254, 255, 256, 257, 258, 259, 260,
      6      261, 262, 263, 264, 265, 266, 267, 268, 269, 270,
      7      271, 272, 273, 274, 275, 276, 277, 278, 279, 280), J1
ISN 0256      201  FOUT(J3,JXX) = P(J2)
ISN 0257      GO TO 300
ISN 0258      202  FOUT(J3,JXX) = FMASS(J2)
ISN 0259      GO TO 300
ISN 0260      203  FOUT(J3,JXX) = U(J2)
ISN 0261      GO TO 300
ISN 0262      204  FOUT(J3,JXX) = TEMP(J2)
ISN 0263      GO TO 300
ISN 0264      205  FOUT(J3,JXX) = AVED(J2)
ISN 0265      GO TO 300
ISN 0266      206  FOUT(J3,JXX) = HW(J2)
ISN 0267      GO TO 300
ISN 0268      207  FOUT(J3,JXX) = AVEX(J2)
ISN 0269      GO TO 300

```

```

EDIT4202
EDIT4204
EDIT4206
EDIT4208
EDIT4210
EDIT4212
EDIT4214
EDIT4216
EDIT4218
EDIT4220
EDIT4222
EDIT4224
EDIT4226
EDIT4228
EDIT4230
EDIT4232
EDIT4234
EDIT4236
EDIT4238
EDIT4240
EDIT4242
EDIT4244
EDIT4246
EDIT4248
EDIT4310
EDIT4320
EDIT4330
EDIT4340
EDIT4350
EDIT4360
EDIT4370
EDIT4380
EDIT4390
EDIT4400
EDIT4410
EDIT4420
EDIT4430
EDIT4435
EDIT4440
EDIT4445
EDIT4450
EDIT4455
EDIT4460
EDIT4465
EDIT4470
EDIT4480
EDIT4490
EDIT4500
EDIT4510
EDIT4520
EDIT4530
EDIT4540
EDIT4550
EDIT4560
EDIT4570
EDIT4580
EDIT4590
EDIT4600

```

```

15N 0270      208 FOUT(J3,JX) = BUHM(J2)
15N 0271      GO TO 300
15N 0272      209 FOUT(J3,JX) = ZM(J2)
15N 0273      GO TO 300
15N 0274      210 FOUT(J3,JX) = SATV(J2)
15N 0275      GO TO 300
15N 0276      211 FOUT(J3,JX) = SATVG(J2)
15N 0277      GO TO 300
15N 0278      212 FOUT(J3,JX) = SATMF(J2)
15N 0279      GO TO 300
15N 0280      213 FOUT(J3,JX) = SATMG(J2)
15N 0281      GO TO 300
15N 0282      214 FOUT(J3,JX) = SATT(J2)
15N 0293      GO TO 300
15N 0294      215 FOUT(J3,JX) = SATP(J2)
15N 0295      GO TO 300
15N 0296      216 FOUT(J3,JX) = LIOM(J2)
15N 0297      GO TO 300
15N 0298      217 FOUT(J3,JX) = WU(J2) * 3600.
15N 0299      GO TO 300
15N 0291      218 DD 2191,K=1,NPMP
15N 0293      IF (1VPUMP(K).EQ.J2) GO TO 2182
15N 0294      CONTINUE
15N 0295      FOUT(J3,JX) = ZERO
15N 0296      GO TO 300
15N 0297      2182 FOUT(J3,JX) = POMGA(K)
15N 0298      GO TO 300
15N 0299      219 DD 2191,K=1,NPMP
15N 0291      IF (1VPUMP(K).EQ.J2) GO TO 2192
15N 0293      CONTINUE
15N 0295      FOUT(J3,JX) = ZERO
15N 0303      2192 FOUT(J3,JX) = TURQUE(K)
15N 0304      GO TO 300
15N 0305      220 FOUT(J3,JX) = WVAR(J2)
15N 0306      GO TO 300
15N 0307      221 FOUT(J3,JX) = ARMAS(J2)
15N 0308      GO TO 300
15N 0309      222 FOUT(J3,JX) = ZERO
15N 0310      GO TO 300
15N 0311      223 FOUT(J3,JX) = ZERO
15N 0312      GO TO 300
15N 0313      224 FOUT(J3,JX) = ZERO
15N 0314      GO TO 300
15N 0315      225 FOUT(J3,JX) = WP(J2)
15N 0316      GO TO 300
15N 0317      226 FOUT(J3,JX) = WP(J2)
15N 0318      GO TO 300
15N 0319      227 FOUT(J3,JX) = XP(J2)
15N 0320      GO TO 300
15N 0321      228 FOUT(J3,JX) = SPV(J2)
15N 0322      GO TO 300
15N 0323      229 FOUT(J3,JX) = DELP(J2)
15N 0324      GO TO 300
15N 0325      230 FOUT(J3,JX) = DELF(J2)
15N 0326      GO TO 300
15N 0327      231 FOUT(J3,JX) = DELE(J2)
15N 0328      GO TO 300

```

```

ISN 0330      232 FOUT(J3,JXX) = PUMPP(J2)
ISN 0331      GO TO 300
ISN 0332      233 FOUT(J3,JXX) = DELA(J2)
ISN 0333      GO TO 300
ISN 0334      234 FOUT(J3,JXX) = KCHOKF(J2)
ISN 0335      GO TO 300
ISN 0336      235 FOUT(J3,JXX) = WP(J2) * SPVJ(J2)
ISN 0337      GO TO 300
ISN 0338      236 FOUT(J3,JXX) = THFTA(J2)
ISN 0339      GO TO 300
ISN 0340      237 IF(AJUNT(J2) .EQ.0.) GO TO 2370
ISN 0341      FOUT(J3,JXX) = WP(J2)*SPVJ(J2)/AJUNT(J2)
ISN 0342      GO TO 300
ISN 0343      2370 FOUT(J3,JXX) = 0.
ISN 0344      GO TO 300
ISN 0345      235 IF(AJUNT(J2) .EQ.0.) GO TO 2380
ISN 0346      CONST = 1.996826D-5
ISN 0348      C      CONST = 1.996826E-5
ISN 0349      FOUT(J3,JXX) = CONST*(WP(J2)*SPVJ(J2)/AJUNT(J2))**2
ISN 0350      GO TO 300
ISN 0351      2380 FOUT(J3,JXX) = 0.
ISN 0352      GO TO 300
ISN 0353      239 IF(AJUNT(J2) .EQ.0.) GO TO 2390
ISN 0354      RGE = 3.1080997D-2
ISN 0355      C      RGE = 3.1080997E-2
ISN 0356      FOUT(J3,JXX) = (RGE*SPVJ(J2)/AJUNT(J2))*WP(J2)**2
ISN 0357      GO TO 300
ISN 0358      2390 FOUT(J3,JXX) = 0.
ISN 0359      GO TO 300
ISN 0360      240 FOUT(J3,JXX) = ZERO
ISN 0361      GO TO 300
ISN 0362      241 FOUT(J3,JXX) = TP(1,J2)
ISN 0363      GO TO 300
ISN 0364      242 M = 1GDM(J2)
ISN 0365      NN = NSI(M)
ISN 0366      FOUT(J3,JXX) = TP(NN,J2)
ISN 0367      GO TO 300
ISN 0368      243 FOUT(J3,JXX) = HTCL(J2)
ISN 0369      GO TO 300
ISN 0370      244 FOUT(J3,JXX) = MTCR(J2)
ISN 0371      GO TO 300
ISN 0372      245 FOUT(J3,JXX) = PHIL(J2)
ISN 0373      GO TO 300
ISN 0374      246 FOUT(J3,JXX) = PHIR(J2)
ISN 0375      GO TO 300
ISN 0376      247 FOUT(J3,JXX) = WOCL(J2)
ISN 0377      GO TO 300
ISN 0378      248 FOUT(J3,JXX) = WGR(J2)
ISN 0379      GO TO 300
ISN 0380      249 FOUT(J3,JXX) = IHTL(J2)
ISN 0381      GO TO 300
ISN 0382      250 FOUT(J3,JXX) = IHTR(J2)
ISN 0383      GO TO 300
ISN 0384      251 FOUT(J3,JXX) = FCHL(J2)
ISN 0385      GO TO 300
ISN 0386      252 FOUT(J3,JXX) = FCHR(J2)
ISN 0387      GO TO 300
ISN 0388      253 FOUT(J3,JXX) = ZERO

```

```

ED115200
ED115210
ED115220
ED115230
ED115240
ED115250
ED115260
ED115270
ED115280
ED115290
ED115300
ED115310
ED115320
ED115330
ED115340
ED115350
ED115360
ED115370
ED115380
ED115390
ED115400
ED115410
ED115420
ED115430
ED115440
ED115450
ED115460
ED115470
ED115480
ED115490
ED115500
ED115510
ED115520

```



```

ISN 0502      617 FORMAT(39H0 VOLUME          PUMP SPEED  PUMP NORM /
1              39H  NUMBER          (PPM)          TORQUE          )
ISN 0503      620 FORMAT(1X,15,7H LEFT,13,19,3X,1P7E13.5)
ISN 0504      621 FORMAT(53H0 JUNCTION          SLIP VEL.  LIQUID VEL.  VAPOR VEL..
*              51H  JCT. FLOW-L  JCT. FLOW-G  SAT. H-L  SAT. H-G
*              19H  FLOW-WEIGHTED H
*              /      51H  NUMBER          (FT/SEC)  (FT/SEC)  (FT/SEC)
*              51H  (LBM/SEC)  (LBM/SEC)  (BTU/LBM)  (BTU/LBM)
*              13H  (BTU/LBM)
ISN 0505      630 FORMAT(1X,15,7H RIGHT,13,19,3X,1P7E13.5)
ISN 0506      640 FORMAT(16H0 HEAT SLAB VOL, 18X, 70H NODE          TEMP          )
IF              TEMP          NODE          TEMP/1AH NUMBER          NUM)
ISN 0507      650 FORMAT(1X,15,7H RIGHT,13,12X,3(110,3X,1P7E13.5))
ISN 0509      660 FORMAT(16H0 HEAT SLAB VOL,13X,76H HEAT GEN          DPTH HEAC          DPTH
1PEAC          CENT TEMP          AVG. TEMP          FUEL POWER/
2              1AH NUMBER          NUM,13X,76H ZR-H2O HEAC  EXTERNALLY  INTER
3              (F)          (F)          (MW)          /
4              29X,31H (BTU/HR)          (FT)          (FT)
ISN 0509      670 FORMAT (1X,15,7H RIGHT,13, 12X, 1P6E13.5)
ISN 0510      680 FORMAT(19H0 VOLUME, 9X,12H HEAT EXCGR., 79X, A4, 1X, A4 /
1              9H  NUMBER, 9X, 7H NUMBER          , 84X, A4, 1X, A4 )
ISN 0511      690 FORMAT (15H0 LIQUID LEVEL =,1P7E13.5,
1 26H FT ABOVE BOTTOM OF VOLUME,14 )
ISN 0512      691 FORMAT( 45H ***** DUE TO PREVIOUS ERROR LOOP INDEX IS,
* JAH UNREASONABLE, THUS EDITING OMITTED. )
C
ISN 0513      695 FORMAT (10 ICE VOL,17X,ICE MASS,5X,ICE AREA,5X,MT,COFF,15X
* MT,TR,RATE  INIT ICE  FRACT LEFT  MELT LOSS,1/
* NUMBER,9X,(LBM),8X,(FT2),8X,BTU/FT2-HR-F  HTU/SEC,6X,MASS (LBM),
* 3X,IN VOL,7X,RATE (LBS/SEC),1)
ISN 0514      696 FORMAT (16,9X,1P7E13.5)
ISN 0515      697 FORMAT ('1')
C
ISN 0516      END

```


FORTRAN M ERROR MESSAGES

NAME	ERROR NO	LEVEL	ERROR MESSAGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE
NAME	FA 1EKJ051	04	THE EQUIVALENCE STATEMENT CONTAINS AN ARRAY WITH A SUBSCRIPT WHICH IS OUT OF RANGE

OPTIONS IN EFFECT NAME= MAIN.OPT=02.LINECNT=60.SIZE=0000K.

OPTIONS IN EFFECT SOURCE=EBCCIC.NOLIST.NODECK.LOAD.NOMAP.NOEDIT.NOIO.NDREF

STATISTICS SOURCE STATEMENTS = 515 .PROGRAM SIZE = 15110

STATISTICS 12 DIAGNOSTICS GENERATED. HIGHEST SEVERITY CODE IS 4

***** END OF COMPILATION *****

163K BYTES OF CORE NOT USED

DATE 78.005/08.37.23

31


```

C      (0 = NO, 1 = INCLUDE)
C      IEMEC = EM ECC BYPASS LOGIC FLAG (0 = NO, 1 = INCLUDE)
C      ISPRNG = PROGRAM OPTION FLAG--STANDARD(0 = R4, 1 = FM, 2 = FLOOD)
C      NCHLK = LENGTH OF THE COMMON BLOCK LENGTH ARRAY
C      LENGT = ARRAY OF COMMON BLOCK LENGTHS
C
ISN 0019 COMMON / STATUS / OMEGA, PERIOD, POWER, TIMEX,
C      LOGICAL NOGO
ISN 0019
C      OMEGA = CRANK-NICOLSON IMPLICIT-EXPLICIT PROPORTION MULTIPLIER
C      PERIOD = REACTOR PERIOD
C      POWER = REACTOR POWER
C      TIMEX = ELAPSED PROBLEM TIME
C      PQUITL = LOWER LIMIT ON PRESSURE IN ANY VOLUME
C      PQUTH = UPPER LIMIT ON PRESSURE IN ANY VOLUME
C      TQUITL = LOWER LIMIT ON TEMPERATURE IN ANY VOLUME
C      TQUTH = UPPER LIMIT ON TEMPERATURE IN ANY VOLUME
C      NOGO = .TRUE. MEANS PROBLEM FAILURE
ISN 0020 DIMENSION STDATA(30)
ISN 0021 COMMON/STCON/ FC3,FC4,FC5,FC6,FC7,FC8,FC9,
C      * XOPTON(5),LOPTON(5),ITSPL(100),KNTDT(10),IEVFRY(20)
C
C      FC3,FC4,... = TIME STEP CONSTANT FACTORS
C      ITSPL = COUNT FOR VOLUME OR JUNCTION CONTROLLING TIME STEP
C      KNTDT = COUNT FOR TIME STEP CONTROL USED
C      IEVFRY = 0 FOR EVERY ACTUAL TIME STEP PRINT OUT
C      XOPTON(2) = ENDCPU (CPU PROBLEM TERMINATION TIME)
C
ISN 0022 LOGICAL IFLAG
ISN 0023 LOGICAL TRIP, IND1, IND2, IND3, IND4, IND5
ISN 0024 DIMENSION MFSAGE(6,7)
CTIMER002 FLAG FOR DUMPING OUT TIMERS AFTER EACH RESTART DUMP
ISN 0025 LOGICAL ZTDUMP
ISN 0026 DATA MESSAGE /
1 4H INP,4HUT E,4HRRQR,4H, .4H .4H .
2 4H END,4H TRI,4HPS S1,4HUNAL,4H, .4H .
3 4H END,4H OF ,4HTIME,4H S1,4HPS, .4H .
4 4H FA1,4HLURE,4H IN ,4HFLD,4HHRH, .4H .
5 4H FA1,4HLURE,4H IN ,4HHAL,4H, .4H .
6 4H REA,4HCTOR,4H KIN,4HETIC,4HMS E4,4HRRQR,4H
7 4H OUT,4HPUT ,4H ERK,4HON, .4H .4H /
C
C      STDATA = THERMODYNAMIC STATE PROPERTIES RETURNED FROM STATE ROUT.
C      SEE SUBROUTINE BAL FOR DEFINITIONS
C
ISN 0027 DATA ZERO / 0. /
ISN 0028 ZTDUMP = .FALSE.
ISN 0029 IFLAG = .FALSE.
C
CTIMER006 INITIAL BLOCK
DATA NZSUBN/ 3/
CALL TIME1(ZDELTY)
ZTIMEA(NZSUBC) = ZTIMEA(NZSUBC) + ZDELTY
IZCALL(NZSUBN) = IZCALL(NZSUBN) + 1
NZSUBS = NZSUBC

```

```

TRAN0532
TRAN0533
TRAN0534
TRAN0535
TRAN0536
TRAN0538
TRAN0539
TRAN0540
TRAN0550
TRAN0560
TRAN0570
TRAN0580
TRAN0590
TRAN0600
TRAN0601
TRAN0602
TRAN0603
TRAN0604
TRAN0610
TRAN0634
TRAN0636
TRAN0638
TRAN0640
TRAN0642
TRAN0644
TRAN0644
TRAN0646
TRAN0650
TRAN0652
TRAN0654
TRAN0656
TRAN0660
TRAN0670
TRAN0672
TRAN0673
TRAN0680
TRAN0690
TRAN0700
TRAN0710
TRAN0720
TRAN0730
TRAN0740
TRAN0750
TRAN0760
TRAN0766
TRAN0767
TRAN0768
TRAN0769
TRAN0770
TRAN0774
TRAN0776
TRAN0780
TRAN0782
TRAN0783
TRAN0784
TRAN0785
TRAN0786
TRAN0787

```

```

ISN 0035      NZSUBC = NZSUBM
ISN 0036      TEST PC= INPUT ERROR
ISN 0037      I = 0
              IF (NOG) GO TO 110
CTIMER002    CALL TIMER(DEL)
              T = T + DELT
              TMAJC = Y
              TDMPC = Y
C
C      INITIALIZE READ-IN VOLUMES
              IF (.LUMP.GT.(-3))
                1 CALL HAL(NVOL,NJUN,NSLB,NCOR,IPROGM,TEMHT,TEMPS,TEMEC,ZERO,
                2 STDATA)
              IF (.LUMP.GE. -3) GO TO 8
              LDAP = LUMP + 3
              3 CONTINUE
              4 IF (NOGO) GO TO 110
C
C      TEST FOR CHECKOUT ONLY
              I = 1
              10 IF (TRIP(1,DUM)) GO TO 110
C
C      INITIAL EDIT
              IND1 = .TRUE.
              IND2 = .FALSE.
C      THE ABOVE RECORD COMMENTED OUT FOR ORML-BOMT: OR INC
C
C      FORCE MINOR EDIT
              IND2 = .TRUE.
C
C
              IND3 = .NOT.NOGO.AND.LDMP.LT.0
              IND4 = .NOT.NOGO.AND.LDMP.LT.(-1).AND.TIMEX.LE.ZERO
              IND5 = .FALSE.
              I = 6
              DT = 0.0
              CALL EDIT (NVOL, NJUN, NPMP, NSLB, NCOR, NMTH,
              1 IND1, IND2, IND3, IND4, IND5, TEMHT, TEMPS, TEMEC,DT)
              IF (NOGO) GO TO 210
C
C      GET NEW TIME STEP
              I = 2
              19 I = 2
              20 DT = TSTP(IND1,IND2,IND3,IND4)
CTIMER001    ZTDUMP = IND3
              IF (NOGO) GO TO 100
C
C      GET NEW JUNCTION FLOWS, VOLUME MASSES AND EN RGIF'S
              I = 3
              * CALL FLOSRHINVOL,NJUN,NPMP,NSLB,NJSMD,MAXPMP,MAXSLB,MAXGOM,
              * MAXMAT, MAXCOR, MAXREG, MAXMOD, MAXTHL, MAXJUN, MAXVOL, MAXSMD,
              2 IPROGM, TEMHT, TEMPS, TEMEC, IDEBUG,DT)
              * CALL FLOSRHINVOL,NJUN,NPMP,NSLB,NJSMD,MAXPMP,MAXSLB,MAXGOM,
              1 MAXMAT, MAXCOR, MAXREG, MAXMOD, MAXTHL, MAXJUN, MAXVOL, MAXSMD,
              2 IPROGM, TEMHT, TEMPS, TEMEC, IDEBUG,DT)
              IF (NOGO) GO TO 100
ISN 0071      I = 3
ISN 0072
ISN 0073

```

```

TRAN0788
TRAN0790
TRAN0800
TRAN0810
TRAN0813
TRAN0814
TRAN0815
TRAN0816
TRAN0817
TRAN0820
TRAN0830
TRAN0840
TRAN0845
TRAN0847
TRAN0848
TRAN0851
TRAN0852
TRAN0861
TRAN0870
TRAN0880
TRAN0890
TRAN0900
TRAN0910
TRAN0920
TRAN0930
TRAN0940
TRAN0950
RDMT 52
RDMT 53
RDMT 54
RDMT 55
RDMT 56
TRAN0960
TRAN0970
TRAN0980
TRAN0990
TRAN0995
TRAN1000
TRAN1001
TRAN1010
TRAN1020
TRAN1030
TRAN1040
TRAN1050
TRAN1060
TRAN1064
TRAN1065
TRAN1070
TRAN1080
TRAN1090
TRAN1100
IRN 1110
IRN 1120
IRN 1130
CDC 1132
CDC 1134
CDC 1136
TRAN1140

```



```

TRAN1414
TRAN1420
TRAN1430
TRAN1440
TRAN1450
TRAN1460
TRAN1470
TRAN1480
TRAN1490
TRAN1500
TRAN1510
TRAN1520
TRAN1530
TRAN1540
TRAN1550
TRAN1560
TRAN1570
TRAN1580
TRAN1590
TRAN1600
TRAN1610
TRAN1620
TRAN1630
TRAN1640
TRAN1650
TRAN1652
TRAN1654
TRAN1660
TRAN1670
TRAN1680
TRAN1681
TRAN1682
TRAN1683
TRAN1685
TRAN1686
TRAN1687
TRAN1688
TRAN1690
TRAN1700
TRAN1710
TRAN1720
TRAN1725
TRAN3000

C      100      END OF PROBLEM
              IND1 = .TRUE.
              IND2 = NED1.GT.0
              IND3 = LOMP.LT.0
              IND4 = LOMPLY.(-I)
              IND5 = LOMP.LT.0
              GO TO 200

C      110      INPUT LINK ONLY. DID NOT GET INTO TRANSIENT
              IND1 = .TRUE.
              IND2 = .FALSE.
              IND3 = .FALSE.
              IND4 = .FALSE.
              IND5 = .FALSE.
              GO TO 200

C      FAILURE DURING TRANSIENT EDIT
              IND1 = .TRUE.
              IND2 = NED1.GT.0
              IND3 = .FALSE.
              IND4 = .FALSE.
              IND5 = .FALSE.

C      DO FINAL EDIT AND RETURN
              CALL EDIT(NVOL,NJUN,NMPC,NSLB,MCON,NMTX,
     1             IND1,IND2,IND3,IND4,IND5,TENMT,IEMPS,IEMEC,DT)
              IF(IPLAG) WRITE(6,1000)

210      CALL SURP
           WRIT('A.620')(MESSAGE(J,I+1),J=1,6)
           IF (.NOT. MOCO.OH. I.LE. I) GO TO 220
           CALL ETSCON
           CALL ZTEDY(I)

220      CONTINUE
CTIMER003   FINAL BLOCK
           NZSUBC = NZSUBS
           CALL TIME(ZDELTI)
           ZTIME(NZSUBN) = ZTIME(NZSUBN) + ZDELTY
           RETURN

C
C      620 FORMAT (1M0/1M0/1M0/1M0/1M0/1M0/1M0/1M0/1M0/5AX.5AA)
1000  FORMAT(1. .........ENDDCPU TIME.....)

***** OPTIONS IN EFFECT ***** NAME= MAIN,OPT=02,LINECNT=60,SIZE=0000K.
***** OPTIONS IN EFFECT ***** SOURCE,EBCDIC,NOLIST,MODECK,LOAD,NOMAP,NODEDIT,NOLC,VCRREF
***** STATISTICS ***** SOURCE STATEMENTS = 160 ,PROGRAM SIZE = 2738
***** STATISTICS ***** NO DIAGNOSTICS GENERATED
***** ***** END OF COMPILATION *****
***** STATISTICS ***** 12 DIAGNOSTICS THIS STEP, HIGHEST SEVERITY CODE IS 4
***** 263K BYTES OF CORE NOT USED *****

```

LEVEL 21.9 (JUN 74)

05/360 FORTAN M

```

COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=63,SIZE=0000K,
SUBROUTINE EDSAVE(INCPR)
C THIS SUBROUTINE IS CALLED FROM SUBROUTINE EDIT AT EACH
C WINDOW EDIT TO SAVE THE INFORMATION NEEDED FOR THE RELAP43J2
C TO ORIGIN INTERFACE
C SEE ORNL-TM
C COMPLETED ON NOVEMBER 15, 1974, BY STEVEN B. CLIFF
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL NOGO
PEALER LAMBDA, MU, INERTA
COMMON /C/PROP, DPOW, FQ(150), LAMBDA, MU(50), PPOW, PRMPT,
1 QMOD(50), OPRAC(50), ON(21), OPMOD(50), SF(50), TAU,
2 TM(50), TS(50), VS(50), ISLR(50), IVOL(50),
COMMON /J/ER/MP(100), WP(100), WSLP(100), WTP(100), VSLIP,
* FMPRAC(100), IJ(100), IK(100), IPUMP(100), IVALVF(100),
* IMOCOR(100), IYAH(100), IZ(100-2), ISTATP,
COMMON /JUNF/ JAJUN(100), JELFV(75), INERTA(100),
* SRCOS(100), ZVOL(75), KCHOK(100),
COMMON /SLAL/ AML(50), AMTR(50), MDL(50), MDNR(50),
* CMML(50), CHKR(50), DMEL(50), DMER(50), DCL(50), DCM(50),
1 MYCL(50), MYCR(50), PHIL(50), PHIR(50), SLPN(50), VOLS(50),
2 MYCL(50), WOCR(50), XSL(50), XSLR(50), ZROT(50), ZTOP(50),
3 IMCL(50), IMCR(50), IMYL(50), IMYR(50), IMCL(50), IMC(50),
4 ISCO(50), ISHO(50), ISSB(50), ISSY(50), IVSL(50), IVSR(50), IX-3(50),
COMMON /SLARAT/ AP(21,50), TP(21,50),
1 LGOM(50), ITPC(50), ITPK(6,50), ITPX(6,50)
COMMON /SLARGM/ ASUL(20), ASUR(20), POFR(6,20), Y(21,20),
1 YP(21,20), VREG(6,20), VR(6,20), XDCR(6,20),
2 LGAP(6,20), IKC(6,20), NII(6,20), NSI(20),
COMMON /STATUS/ OMEGA, PERIOD, POWER, TIMEK,
1 POUTL, POUTM, TOUTL, TOUTM, NOGO,
COMMON /VOLED/ AVUL(75), AVX(75), SATP(75), SATY(75),
1 SATVP(75), SATVG(75)
COMMON /VOLER/ A(75), HB(75), P(75), TEMP(75), ZI(75),
1 ZM(75), ZMO(75), IOIN(75), IRRAD(75)
DIMENSION SARRAY(502), NVOL(50), BP(50), STEMP(50),
* SAVEX(50), SPHIR(50), SMICH(50), STP(50), SMSFLX(50),
* SSATT(50), SMYRI(50)
EQUIVALENCE
( SARRAY(1), STEMP ),
( SARRAY(2), NSLRS ),
( SARRAY(3), NVOL(1) ),
( SARRAY(53), SP(1) ),
( SARRAY(103), STEMP(1) ),
( SARRAY(153), SAVEX(1) ),
( SARRAY(203), SPHIR(1) ),
( SARRAY(253), SMYRI(1) ),
( SARRAY(303), STP(1) ),
( SARRAY(353), SMSFLX(1) ),
( SARRAY(403), SSATT(1) ),
( SARRAY(453), SMYRI(1) )
C GET TIME VALUE
STEPM = TIMEK
C BEGIN LOOP THROUGH SLABS LOOKING AT ALL VOLUMES TO THE
C RIGHT OF THE HEAT SLAB-
I = 0
C LOOP THROUGH ALL CORE SECTIONS

```

F03V0010
 F03V0020
 F03V0030
 F03V0040
 F03V0050
 F03V0060
 F03V0070
 F03V0080
 F03V0090
 F03V0100
 F03V0110
 F03V0120
 F03V0130
 F03V0140
 F03V0150
 F03V0160
 F03V0170
 F03V0180
 F03V0190
 F03V0200
 F03V0210
 F03V0220
 F03V0230
 F03V0240
 F03V0250
 F03V0260
 F03V0270
 F03V0280
 F03V0290
 F03V0300
 F03V0310
 F03V0320
 F03V0330
 F03V0340
 F03V0350
 F03V0360
 F03V0370
 F03V0380
 F03V0390
 F03V0400
 F03V0410
 F03V0420
 F03V0430
 F03V0440
 F03V0450
 F03V0460
 F03V0470
 F03V0480
 F03V0490
 F03V0500
 F03V0510
 F03V0520
 F03V0530
 F03V0540
 F03V0550
 F03V0560


```

ISN 0020      DO 300 J=1,NCOR
ISN 0021      C GET SLAB ASSOCIATED WITH THIS CORE SECTION
ISN 0022      L = ISLB(J)
ISN 0023      C GET & SAVE VOLUME ID TO RIGHT OF THIS SLAB
ISN 0024      K = IVSR(L)
ISN 0025      IF(K.EQ.0) GO TO 300
ISN 0026      I = I + 1
ISN 0027      NVOL(I) = K
ISN 0028      SP(I) = P(K)
ISN 0029      C GET AVERAGE PRESSURE FOR THIS VOLUME
ISN 0030      C GET AVERAGE TEMPERATURE FOR THIS VOLUME
ISN 0031      STEMP(I) = TEMPI(K)
ISN 0032      C GET AVERAGE QUALITY OF THIS VOLUME
ISN 0033      SAVED(I) = AVEX(K)
ISN 0034      C GET SURFACE FLUX OF THE RIGHT SIDE OF THE SLAB
ISN 0035      SPHR(I) = PHIR(L)
ISN 0036      C GET SURFACE HEAT TRANSFER COEFFICIENT FOR THE RIGHT SIDE OF THE SLAB
ISN 0037      SHTR(I) = HTR(L)
ISN 0038      ME = IGOM(L)
ISN 0039      MM = NSI(M)
ISN 0040      C GET INPUT JUNCTION ID FOR THIS VOLUME
ISN 0041      MI = 0
ISN 0042      IF(MI.EQ.1) GO TO 100
ISN 0043      C GET OUTPUT JUNCTION ID FOR THIS VOLUME
ISN 0044      MO = 0
ISN 0045      IF(MO.EQ.2) GO TO 200
ISN 0046      C CALCULATE & STORE MASS FLUX RATE
ISN 0047      MSFLX(I) = ABS(SGOL((MP(MI)/AJUNIM1)+MP(MO)/AJUNIM2))/2.00))
ISN 0048      C GET SATURATION TEMPERATURE FOR THIS VOLUME
ISN 0049      SSAT(I) = SAT(K)
ISN 0050      C GET HEAT TRANSFER MODE AT RIGHT SLAB SURFACE
ISN 0051      SHTR(I) = HTR(L)
ISN 0052      300 CONTINUE
ISN 0053      C GET NUMBER OF SLABS
ISN 0054      NSLBS=I
ISN 0055      C WRITE THIS EDIT VALUE OUT
ISN 0056      WRITE(20) SARRAY
ISN 0057      RETURN
ISN 0058      END

*OPTIONS IN EFFECT*      NAME= MAIN,OPT=02,LINENCT=60,SIZE=0000K,
*OPTIONS IN EFFECT*      SOURCE=EBDCIC,NOLIST,NODECK,LOAD,NOMAP,NODEIT,NOLIO,NOLREF
*STATISTICS*      SOURCE STATEMENTS = 49 ,PROGRAM SIZE = 2816
*STATISTICS*      NO DIAGNOSTICS GENERATED
***** END OF COMPILATION *****

```

299K BYTES OF CORE NOT USED

```

EDSV0570
EDSV0580
EDSV0590
EDSV0600
EDSV0610
EDSV0620
EDSV0630
EDSV0640
EDSV0650
EDSV0660
EDSV0670
EDSV0680
EDSV0690
EDSV0700
EDSV0710
EDSV0720
EDSV0730
EDSV0740
EDSV0750
EDSV0760
EDSV0770
EDSV0780
EDSV0790
EDSV0800
EDSV0810
EDSV0820
EDSV0830
EDSV0840
EDSV0850
EDSV0860
EDSV0870
EDSV0880
EDSV0890
EDSV0900
EDSV0910
EDSV0920
EDSV0930
EDSV0940
EDSV0950
EDSV0960
EDSV0970
EDSV0980

```