LA-UR- - , Vol. III
NUREG/CR-

TRAC-M

Fortran 77, Version 5.5

PROGRAMMERS GUIDE

VOLUME III

by

R. G. Steinke, S. J. Jolly-Woodruff, and J. W. Spore

Nuclear Systems Design and Analysis Group TSA-10

Technology and Safety Assessment Division

Los Alamos National Laboratory

Los Alamos, New Mexico

April 1999

TRAC-M
Fortran 77, Version 5.5
PROGRAMMERS GUIDE
VOLUME III

by
R. G. Steinke, S. J. Jolly-Woodruff, and J. W. Spore

ABSTRACT

The Transient Reactor Analysis Code (TRAC) was developed to provide advanced best-estimate predictions of postulated accidents in light-water reactors. The TRAC-M program provides this capability for pressurized water reactors and for many thermal-hydraulic test facilities. The code features a one- (1-), two- (2-), and three-dimensional (3D) treatment of the pressure vessel and its associated internals. The code includes a two-fluid nonequilibrium hydrodynamics model with a noncondensable-gas field and solute tracking, flow-regime-dependent constitutive equation treatment, optional reflood tracking capability for bottom- and top-flood and falling-film quench fronts, and a consistent treatment of the entire accident sequences, including the generation of consistent initial conditions. The stability-enhancing two-step numerical algorithm is used in the 1-, 2-, and 3D hydrodynamics, and permits violation of the material Courant limit. This technique permits large timesteps, hence the running time for slow transients is reduced.

TRAC-M has a heat-structure (HTSTR) component and a radiation heat-transfer model that allows the user to model heat transfer accurately for complicated geometries. An improved reflood model based on mechanistic and defensible models has been added. TRAC-M also contains improved constitutive models and additions and refinements for several components.

This manual is the third volume of a four-volume set of documents on TRAC-M. This guide was developed to assist the TRAC-M programmer and contains information on the TRAC-M Version 1.10+ code and data structure, the TRAC-M calculational sequence, and memory.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

Many people contributed to recent TRAC-P and TRAC-M code development and to this report. We would like to acknowledge the contributions of Laura A. Guffee, who was a major contributor to an earlier version of the TRAC code. Because it was a team effort, there was considerable overlap in responsibilities and contributions. The participants are listed according to their primary activity. Those with the prime responsibility for each area are listed first.

Principal Investigators:	Jay W. Spore, Susan J. Jolly-Woodruff, Ju-Chuan Lin, Ralph A. Nelson, Robert G. Steinke, and James F. Dearing
Fluid Dynamics:	Jay W. Spore, Susan J. Jolly-Woodruff, Ju-Chuan Lin, and Robert G. Steinke
Heat Transfer:	Ralph A. Nelson, Kemal Pasamehmetoglu, Norman M. Schnurr, and Cetin Unal
Neutronics:	Robert G. Steinke and Jay W. Spore
Code Development and Programming:	James F. Dearing, Susan J. Jolly-Woodruff, Paul T. Giguere, Ju-Chuan Lin, Victor Martinez, Jay W. Spore, and Robert G. Steinke
Control Procedure:	Robert G. Steinke
Graphics:	James F. Dearing, Victor Martinez, and Michael R. Turner
Report compilation:	Jay W. Spore, Paul T. Giguere, and Robert G. Steinke
Editing:	Gloria E. Mirabal

In addition to those listed on the previous page, we acknowledge all others who contributed to earlier versions of TRAC. In particular, the two-step numerics and network solver developed by John Mahaffy are a major part of the TRAC-P and TRAC-M codes. Dennis R. Liles contributed heavily to the thermal-hydraulics modeling and to the overall direction of MOD1 code development. Frank L. Addessio developed the steam-generator component, and Manjit S. Sahota developed the critical-flow model and the turbine component. Thad D. Knight provided direction for improvements to TRAC based on assessment-calculation feedback and he coordinated the development of the MOD1 Correlation and Models document. Richard J. Pryor, Sandia National Laboratories, and James Sicilian, Flow Science, Inc., provided major contributions to the code architecture. We also acknowledge useful discussions and technical exchanges with Louis M. Shotkin and Novak Zuber, US Nuclear Regulatory Commission; Terrence F. Bott, Francis H. Harlow, David A. Mandell, and Burton Wendroff, Los Alamos National Laboratory; John E. Meyer and Peter Griffith, Massachusetts Institute of Technology; S. George Bankoff, Northwestern University; Garrett Birkhoff, Harvard University; and Ronald P. Harper, Flow Science Inc.

1. INTRODUCTION

The TRAC-M Programmers Guide has been developed to assist the Transient Reactor Analysis Code (TRAC) programmer. Much of the information presented here is provided in appendices. The appendices are self-contained and are meant to be used as references. Topics of discussion addressed in the body of this manual include the TRAC-M calculational sequence, the TRAC-M code structure and data structure, computer-memory management, and various machine configurations supported by TRAC-M, FORTRAN 77, Version 5.5.

2. CODE ARCHITECTURE

The description of the TRAC-M code architecture given here is divided into two areas: code structure and data structure. Because the data structure for the one-dimensional (1D) hydraulic components differs from that of the three-dimensional (3D) VESSEL component, the structures are detailed separately in the following discussion.

2.1. Code Structure

TRAC was developed in a modular fashion in an effort to strive for a code structure that minimizes the problems of maintaining and extending the code. This modularity manifests itself in two important ways. First, because TRAC analyzes nuclear-reactor systems that consist of specific component types, the code is written to utilize subroutines that handle specific component types. For example, data and calculations pertaining to a PIPE component are handled separately from data and calculations for a VESSEL component. The different TRAC-M components are described in greater detail in the TRAC-M Users Guide, the second volume of manual documentation. Second, the TRAC program is written to be functionally modular; that is, each TRAC subprogram performs a specific function. Some low-level subprograms are used by all components, thereby strengthening this modularity. Appendix A lists all TRAC-M subroutine and function routines and their descriptions. For each routine, Appendix B lists all routines that it calls and all routines from which it is called.

Functional modularity within TRAC-M is taken a step further by grouping routines into modules. Figure 1 displays a calling-tree representation of the TRAC modules. A brief description of the function of each module is presented in Table 1. Use of a module overlay structure was mandated originally by computer-memory

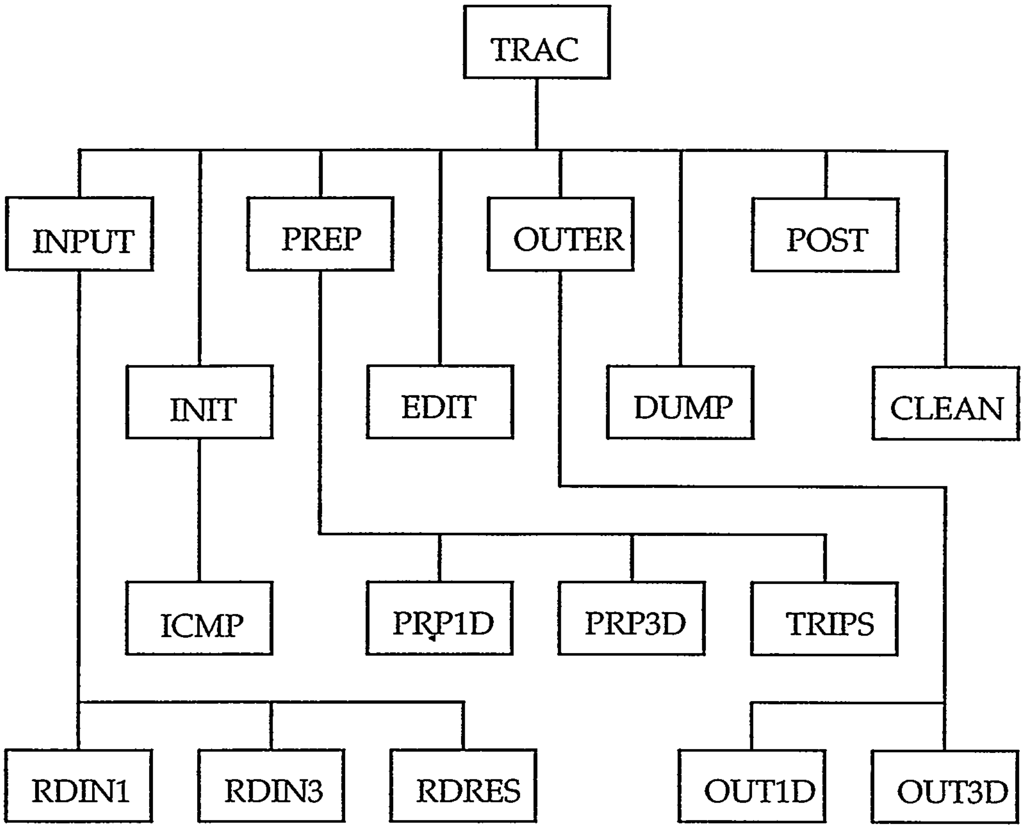

Fig. 1.
TRAC-M module structure.

size limitations where only selected module overlays would be in the computer memory at any given time. This is no longer true because sufficient computer memory is available for the entire TRAC-M executable on computer platforms on which TRAC-M is currently run. The module overlay structure in TRAC-M is no longer maintained for users with small-memory computers. Subsequent references to modules refer to that portion of TRAC-M coding that used to be loaded into and unloaded from computer memory as a block but is now resident in computer memory at all times during a calculation.

The overall sequence of the calculation is directed by the TRAC main program. Module INPUT always is invoked at the start of each TRAC execution to read control-procedure and component input data. Component data are initialized by the module INIT. A steady-state calculation (if requested) is performed by subroutine STEADY. During a steady-state calculation, the reactor-core power is initially zero and is set to its steady-state power level after fluid flow has been

TABLE 1
TRAC-M MODULES

Module	Description
TRAC	Controls overall flow of the calculation (also contains many service routines used throughout the code).
CLEAN	Closes all output files.
DUMP	Performs restart dumps.
EDIT	Adds an output edit at the current time to the TRCOUT file.
ICMP	Initializes component data.
INIT	Controls initialization of component and graphics data.
INPUT	Controls reading input and restart files and analyzes piping loops.
OUTER	Controls one complete outer iteration for all components.
OUT1D	Performs one outer iteration on the basic finite-difference flow equations for all 1D hydraulic components.
OUT3D	Performs one outer iteration for all 3D VESSEL components.
POST	Performs postpass for all hydraulic and heat-structure components.
PREP	Performs prepass for all components.
PRP1D	Performs the prepass calculations for 1D components.
PRP3D	Performs prepass calculation for all 3D VESSEL components.
RDIN1	Inputs and stores 1D hydraulic-component data.
RDIN3	Inputs and stores 3D VESSEL-component data.
RDRES	Reads and stores data from the restart-data TRCRST file.
TRIPS	Evaluates signal variables, control blocks, and trips for the control procedure.

established. This prevents high rod temperatures early in the steady-state calculation when the input fluid state generally starts from a stagnant (no-flow) condition. A transient calculation (if requested) is performed by subroutine TRANS. Modules EDIT and DUMP are called during a steady-state calculation by subroutine STEADY and during a transient calculation by subroutine TRANS by calling subroutine PSTEPQ to generate current-time output results at the user's request. Overlay CLEAN is called to close all output files at the end of the calculation or when a fatal error occurs, aborting the calculation.

2.2. Data Structure

TRAC-M divides the data for each component into four blocks: generic-component table, specific-component table, pointer table, and array data. The first three blocks are stored in memory as COMMON blocks FLTAB, compCOM, and PTAB, respectively. COMMON Block compCOM has the comp part of its name replaced by the component name; i.e., BREAK, FILL, PIPE, PRIZ, PUMP, ROD, TEE, TURB, VALVE, and VSS. The structure of COMMON area FLTAB is the same for all components. Variables in COMMON blocks compCOM and PTAB differ from one component to another component. The specific-component and pointer tables for each component are described with definitions of their variables in Appendix C. Section D.38 in Appendix D describes the generic-component table of the include file FIXEDLT.H with definitions of its variables.

Array data are stored in computer memory within the dynamic-storage container A array of blank COMMON. The location of an individual array variable is determined by the value of its pointer variable in the component pointer table for 1D hydraulic (BREAK, FILL, PIPE, PRIZER, PUMP, SEPD, TEE, TURB, and VALVE), 3D VESSEL, and heat-structure ROD or SLAB components. Dynamic storage of data arrays permits effective use of computer memory for different size problems. Aspects of computer-memory management are discussed further in Sec. 4.

In addition to the data that refer to a particular component, TRAC-M uses many variables to describe the overall solution state evaluated by the calculation. These variables are grouped in labeled COMMON blocks according to their use. Appendix D documents these COMMON blocks and lists their variables and the variable definitions.

2.2.1. 1D Data Structure. The pointer tables for all 1D hydraulic components have a similar structure consisting of four groups of pointer variables and one component-type group. The first group of pointer variables contained in include file DUALPT.H locates dual-time hydrodynamic- and thermodynamic-parameter information. The main group of pointer variables contained in include file HYDROPT.H locates the remaining single-time hydrodynamic- and thermodynamic-parameter information.

Integer data stored as REAL*8 values are located using the third group of pointer variables contained in include file INTPT.H. The fourth group of pointer variables contained in include file HEATPT.H locates data for wall-heat transfer in those hydraulic components that support the wall heat-transfer calculation. Array data that are specific to a particular component type (if any such arrays exist) are located using the last group of pointer variables in the pointer table. The first three groups of pointer-table variables and the last group of pointer-table variables for each component type are defined in Appendix C.

2.2.1.1. Adding a 1D database variable. The standard guidelines given below are followed to add a new database variable to all 1D hydraulic components. Using these guidelines, a programming example, provided in Appendix E, adds a new variable to each group discussed above.

1. Create new pointer names for the new array variables and add them to the pointer tables of the appropriate include files.
   a. If the new array variable requires both old-time and new-time REAL*8 values, then two new pointers must be added to the DUALPT.H include file. If the pointers become the first two pointers of the DUALPT.H include file because of alphabetic considerations, the EQUIVALENCE statement in DUALPT.H of LALP to PT(1) must be changed to reflect this.
   b. If the new array variable with a single-time REAL*8 value is associated with the hydrodynamic calculation, its new pointer is added to the HYDROPT.H include file.
   c. If the new array variable with a single-time integer value is stored as a REAL*8 value, its new pointer is added to the INTPT.H include file.
   d. If the new array variable with a single-time REAL*8 value is associated with the wall heat-transfer calculation, its new pointer is added to the HEATPT.H include file.
2. Initialize the new pointers.
   a. If new pointer variables were added to DUALPT.H, these new pointer variables are initialized in subroutine S1DPTR during module INPUT in the DUALPT.H pointer section.

      – If the new array variable is one for which old-time and new-time values are the same at the start of the OUTER module (that is, the new-time value is reset to the old-time value in the event of a backup due to 1D component water packing, for instance), then the new old-time pointer should be initialized *after* the LALP pointer and *before* the LVV pointer. Similarly, the new new-time pointer should be initialized *after* the LALPN pointer and *before* the LVVN pointer in the same relative position as the new old-time pointer.

– If the new array variable is one for which old-time and new-time values are not the same at the start of the OUTER module (that is, the new-time value is not reset to the old-time value in the event of a backup due to 1D component water packing, for instance), then the new old-time pointer should be initialized *after* the LBIT pointer but *before* the LVVTO pointer. Similarly, the new new-time pointer should be initialized *after* the LBITN pointer but *before* the LVVT pointer in the same relative position as the new old-time pointer.

– Define the value of the array pointer initialized directly after each new pointer you add to reflect correctly the size of the new pointer array storage requirement. Increment the value of LENPTR in the DUALPT.H pointer section of S1DPTR by the number of pointer variables added to the DUALPT.H include file.

b. If a new array pointer variable was added to HYDROPT.H, it is initialized in subroutine S1DPTR. The new pointer should be added *before* the LNXT pointer in the HYDROPT.H section of S1DPTR. Adjust the value of the LNXT pointer to reflect the size of the array storage of the newly added pointer variable. Increment the value of the variable LENPTR by the number of pointer variables added to the HYDROPT.H pointer section of S1DPTR.

c. If a new array pointer variable was added to INTPT.H, it is initialized in subroutine S1DPTR. The new pointer should be added *before* the LNXT pointer in the INTPT.H section of S1DPTR. Adjust the value of the LNXT pointer to reflect the size of the array storage of the newly added pointer. Increment the value of the variable LENPTR by the number of pointer variables added to the INTPT.H pointer section of S1DPTR.

d. If a new array pointer variable was added to HEATPT, it is initialized in subroutine S1DPTR. The new pointer should be added *before* the LNXT pointer in the HEATPT.H section of S1DPTR. Adjust the value of the LNXT pointer to reflect the size3 of the array storage of the newly added pointer. Increment the value of the variable LENPTR by the number of pointer variables added to the HEATPT pointer section of S1DPTR.

3. If the new array variables are to be output to the data-dump TRCDMP file, include a call to BFOUT in subroutine DCOMP for each variable to have its data output. If the new array variable being output is a cell-edge quantity with a size of NCELLS+1 elements, then increment LVEDGE by one. If the new array variable being output is a cell-center quantity with a size of NCELLS elements, then increment LVCNTR by one. If the new variable has dimensions other than NCELLS or NCELLS+1, increase LCOMP by the size of the new array variable.
4. To read in the new array variables from the data-dump file for restarting, file TRCRST, add calls to BFIN in subroutine RECOMP in the same order as the BFOUT calls were added to DCOMP (note that RECOMP must be changed if DCOMP is changed).
5. Add the new array variables to the argument list of the subroutines in which they will be calculated. Also include DIMENSION statements for these arrays. Perform the necessary calculations to determine the new variable values within the subroutines.
6. Add the new array variables to the argument list of all calling statements to the subroutines in which the new variables are calculated.

2.2.2. 3D Data Structure. The data structure used for the VESSEL hydrodynamic data in TRAC-M is mesh-wise, in contrast to the cell-wise data structure used in the TRAC-P implementation. VESSEL coding is defined directly in terms of two-dimensional (2D) and 3D arrays. This new data structure was chosen primarily to simplify code development, improve code readability, and eliminate the use of EQUIVALENCE statements required by the inverted cell-wise database. Now the first subscript dimension is the I-direction cell number with a stride of 1 rather than a stride of NV for the total number of array parameters.

2.2.2.1. Mesh-wise vs cell-wise data storage. Data defined on a computational mesh can be stored in two ways: mesh-wise and cell-wise. For mesh-wise storage in TRAC-M, all values for a given data parameter or a given array (e.g., all pressures) are stored contiguously in computer memory. For cell-wise storage in TRAC-P, however, values for all the different data parameters associated with a single mesh cell (e.g., pressure, temperature, volume, etc.), are stored contiguously in computer memory. Reference to consecutive mesh cells of a given array parameter using cell-wise storage will, of course, necessitate the use of a stride equal to the total number of data parameters stored for each cell.

Rather than using cell-wise storage as in TRAC-P, TRAC-M uses mesh-wise storage for the VESSEL 2- and 3D data. This methodology was chosen because it has certain advantages over cell-wise storage. These advantages include simpler code development and code maintenance through the avoidance of an NV stride applied to the I-direction cell number and not requiring EQUIVALENCE of the 2- and 3D arrays to the container A array. The 2- and 3D arrays are now stored in their own labeled COMMON blocks.

2.2.2.2. Mesh-wise implementation for 3D data. TRAC-M stores its 2- and 3D arrays in COMMON blocks VSSWHAT and VSSARCOM, respectively. The 2D arrays are used to pass (I,J,K) array data as (I,J) data for a given axial level K to/from the heat-transfer calculation of a heat-structure component ROD or SLAB that the VESSEL is thermally coupled to. Include file EQUIV.H defines these COMMON blocks with the form:

```
      REAL*8 hla(ni,nj,nk),hva(ni,nj,l),q3drl(ni,nj,nk), . . .
      . . . 286 different parameter arrays . . .
     & sc2m(ni,nj,nk), scd2m(ni,nj,nk)
!
      COMMON /vssArCom/ hla,hva,q3drl, . . .
      . . . 286 different parameter-array names . . .
     &  . . . ,sc2m,scd2m
!
      REAL*8 ualpag(ni,nj),ualprw(ni,nj),ualpsm(ni,nj), . . .
      . . . 16 different parameter arrays . . .
     &  . . . ,vztb(ni,nj),mrefld(ni,nj)
!
      COMMON /vssWhat/ ualpag,ualprw,ualpsm, . . .
      . . . 16 different parameter-array names . . .
     &  . . . ,vztb,mrefld
```

In the TRAC-M implementation of the 2D array, NI and NJ are used. For the 3D arrays, NK is used for the third dimension. All are defined by PARAMETER-statement constants. This results in an input-data limit on the number of radial rings or x-direction cells (first dimension), azimuthal sectors or y-direction cells (second dimension), and axial levels or z-direction cells (third dimension). As discussed in Sec. 5, hard-coded array dimensions have both code development and code debugging advantages over variable array dimensions; however, they also have disadvantages, including the possibility of having to change the source code for VESSEL input data with a larger dimension requirement.

The TRAC-M implementation of the VESSEL data may seem very similar to static-memory allocation. In fact, the implementation is flexible and dynamic in that the NI, NJ, and NK PARAMETER constants can be easily changed to redimension the 2- and 3D arrays, and the number of 3D VESSEL components is arbitrary. Some computer-memory space will be wasted when a single VESSEL input model requires dimensions less than the NI, NJ, and NK constants and when a multi-VESSEL input model has individual VESSEL dimensions that are different and less than the NI, NJ, and NK constants.

We note that most implementation difficulties experienced with either cell-wise or mesh-wise storage could be avoided by using widely available but nonstandard POINTER construct that associates arrays with variable starting addresses in a container array. The approach taken in TRAC-M, however, has been to use standard FORTRAN to ensure code portability.

2.2.2.2.1. Include file PARSET1.H. All 2- and 3D array data for the TRAC-M VESSEL component are declared in the include file EQUIV.H. The dimension PARAMETER constants NI, NJ, and NK are defined in the include file PARSET1.H.

All 2D or 3D arrays in the mesh-wise storage scheme have the same dimension. This is accomplished by the use of the following INTEGER PARAMETER constants defined in the include file PARSET1.H.

`NV=1`	`Defines the stride between I-direction adjacent cells.`
`NXRMX=24`	`The maximum number of radial rings or x-direction cells in the 2D or 3D mesh.`
`NYTMX=24`	`The maximum number of azimuthal sectors or y-direction cells in the 2D or 3D mesh.`
`NZMX=50`	`The maximum number of axial or z-direction cells in the 3D mesh.`

NXBCM=2	The number of phantom or boundary cells next to radial ring or x-direction cell 1.
NYBCM=2	The number of phantom or boundary cells next to azimuthal sector or y-direction cell 1.
NZBCM=2	The number of phantom or boundary cells next to axial or z-direction cell 1.
NXBCP=1	The number of phantom or boundary cells next to radial ring or x-direction cell NXRMX.
NYBCP=1	The number of phantom or boundary cells next to azimuthal sector or y-direction cell NYTMX.
NZBCP=1	The number of phantom or boundary cells next to axial or z-direction cell NZMX.

Combinations of these PARAMETER constants are then used to determine the array dimensions, i.e.,

N1CN=NXRMX+NXBCM+NXBCP=27	The total number of radial rings or x-direction cells.
NI=N1CN*NV=27	The first dimension of the 2D or 3D arrays.
NJ=NYTMX+NYBCM+NYBCP=27	The total number of azimuthal sectors or y-direction cells and the second dimension of the 2D or 3D arrays.
NK=NZMX+NZBCM+NZBCP=53	The total number of axial or z-direction cells and the third dimension of the 3D arrays.

The TRAC-M user should not change any of these PARAMETER constants, except for NXRMX=24, NYTMX=24, and NZMX=50 when their maximum array dimensions are either inadequate (too small) or too wasteful of computer memory. Further discussion on using phantom or boundary cells is in Sec. 2.2.2.5.

2.2.2.2.2. Loop limits. All array-dimension loop-limit variable names have the same naming convention with the first letter, i.e., I, J, and K, indicating the first (radial- or x-direction), second (azimuthal- or y-direction), and third (axial- or z-direction) array dimensions, respectively. The letter C in a name denotes a limit suitable for looping over cells, and the letter F denotes a limit suitable for looping over cell faces. The convention for cell-face variables in the TRAC-M VESSEL is the same as in TRAC-P: the cell-face data at the outer r or x, forward θ or y, or upper z face of a cell have the same index as the data at the cell center. Note that, as indicated below, cell faces at the VESSEL boundaries are only included in the cell-face loops when their velocities need to be calculated as a result of using the generalized boundary-condition IVSSBF option for a pressure boundary condition.

The numeral 0 in a name denotes a lower limit, and the letter X denotes an upper limit. The suffix M denotes a lower limit that includes the phantom cell adjacent to the first physical cell, whereas the suffix MM denotes a lower limit that includes all the low-numbered phantom cells. The suffix P denotes an upper limit that includes the phantom cell adjacent to the last physical cell, whereas the suffix ALL denotes an upper limit that includes all the high-numbered phantom cells.

The variable names for the radial- or x-direction are:

IC0MM	Lower limit for loop over all radial rings or x-direction cells in the computational mesh.
IC0M	Lower limit for loop over radial rings or x-direction cells in the physical mesh and the adjacent low-numbered phantom or boundary radial ring or x-direction cell.
IC0	Lower limit for loop over all radial rings or x-direction cells in the physical mesh.
IF0	Lower limit for loop over all radial-ring faces or x-direction-cell faces at which velocities are calculated.
ICX	Upper limit for loop over all radial rings or x-direction cells in the physical mesh.
IFX	Upper limit for loop over all radial-ring faces or x-direction-cell faces at which velocities are calculated.
ICXP	Upper limit for loop over radial rings or x-direction cells in the physical mesh and the adjacent high-numbered phantom or boundary radial ring or x-direction cell.
IALL	Upper limit for loop over all radial rings or x-direction cells in the computational mesh.

The variable names for the azimuthal or y-direction loop limits can be obtained by replacing the leading I with a J, and those for the axial or z-direction loops by replacing the leading I with a K.

There is no reason why the code developer should have to change any of the coding of the loop limits in either include file PARSET2.H or in subroutine RVSSL. In fact, this is a major advantage of the TRAC-M VESSEL-data implementation: all maintenance of the memory-management functionality can be accomplished by changing only three variables in the include file PARSET1.H: NXRMX, NYTMX, and NZMX. The coding of the loop limits is described here merely for completeness.

Certain loop limits can, of course, be hard-coded with PARAMETER statements, which are defined as follows in the include file PARSET2.H:

```
JC0P   =NYBCM+1
JC0MP  =JC0P-1
JC0MMP=JC0P-NYBCM
KC0P   =NZBCM+1
KC0MP  =KC0P-1
KC0MMP=KC0P-NZBCM
```

The letter P in these names stands for "PARAMETER" because they are PARAMETER constants. These constants are copied to the corresponding COMMON variables JC0, JC0M, JC0MM, KC0, KC0M, and KC0MM using the standard naming convention in subroutine RVSSL.

Additional radial- or x-direction, azimuthal- or y-direction, and axial or z-direction lower loop limits, as well as additional upper loop limits, are defined dynamically for each 3D VESSEL component in subroutine RVSSL. This coding is reproduced below (in a restructured form) where

NXR is the input number of physical radial rings or x-direction cells,

NYT is the input number of physical azimuthal sectors or y-direction cells,

NZZ is the input number of physical axial levels or z-direction cells,

IGEOM is 0 for cylindrical geometry and 1 for Cartesian geometry,

IGBCXR is nonzero for generalized radial- or x-direction boundary conditions,

IGBCYT is nonzero for generalized azimuthal- or y-direction boundary conditions, and IGBCZ is nonzero for generalized axial-direction boundary conditions.

In the current version of TRAC-M, IGBCXR and IGBCYT are always 0 and IGBCZ is only nonzero when the VESSEL outer boundary-condition input flag, IVSSBF, is nonzero.

For the first index representing the radial or x direction:

```
      ic0mm  =  lasti  + 1
      ic0    =  ic0mm  + nxbcm*nv
      ic0m   =  ic0    - nv
      icx    =  ic0    + (nxr - 1)*nv
      icxp   =  icx    + nv
      iall   =  icx    + nxbcp*nv
      lasti  =  lasti  + iall
      if0    =  ic0
      IF (igeom.EQ.1.AND.igbcxr.EQ.1) if0=ic0m
!
!     calculate nxrv, the number of radial-ring or x-direc-
!     tion-cell faces where velocities must be calculated.
!
      IF (igeom.EQ.0) THEN
        if0 = ic0
        IF (igbcxr.EQ.0) THEN
          nxrv = nxr - 1
        ELSE
          nxrv = nxr
        ENDIF
      ELSE
        IF (igbcxr.EQ.0) THEN
          if0  = ic0
          nxrv = nxr - 1
        ELSE
          if0  = ic0m
          nxrv = nxr + 1
        ENDIF
      ENDIF
      ifx = if0 + (nxrv-1)*nv
```

The inclusion of constant NV in this coding for the radial- or x-direction loop limits is a holdover from cell-wise storage where NV=291, the total number of different array parameters. The PARAMETER (NV=1) statement in the include file PARSET1.H converts this defining form to mesh-wise storage. LASTI=0 for the first 3D VESSEL component and is incremented by IALL for each succeeding VESSEL component in a multi-VESSEL problem.

For the second index representing the azimuthal or y direction:

```
      jcx   =  jc0 + nyt - 1
      jcxp  =  jcx + 1
      jall  =  jcx + nybcp
      jf0   =  jc0
```

```
      IF (igeom.EQ.1.AND.igbcyt.EQ.1) jf0=jc0m
!
!     calculate nytv, the number of azimuthal-sector or y-
!     direction-cell faces where velocities must be calculated.
!
      IF (igeom.EQ.0) THEN
        jf0 = jc0
        IF (nyt.GT.1) THEN
          nytv = nyt
        ELSE
          nytv = 0
        ENDIF
      ELSE
        IF (igbcyt.EQ.0) THEN
          jf0  = jc0
          nytv = nyt - 1
        ELSE
          jf0  = jc0m
          nytv = nyt + 1
        ENDIF
      ENDIF
      jfx = jf0 + nytv - 1
```

For the third index representing the axial or z direction:

```
      kcx  =  kc0 + nzz - 1
      kcxp =  kcx + 1
      kall =  kcx + nzbcp
      kf0  =  kc0
      IF (igbcz.EQ.1) kf0=kc0m
!
!     calculate nzzv, the number of axial-cell
!     faces where velocities must be calculated.
!
      IF (igbcz.EQ.0) THEN
        kf0  = kc0
        nzzv = nz - 1
      ELSE
        kf0  = kc0m
        nzzv = nz + 1
      ENDIF
      kfx = kf0 + nzzv - 1
```

2.2.2.2.3. Mesh-wise storage for one variable in one level. TRAC-M uses temporary mesh-wise storage of a single axial level for input and output of a single 3D array variable. A temporary mesh-wise array sufficient to hold one level of data for one array is allocated with the pointer LTEMPS in subroutine RVSSL. Subroutine LEVELR is a generic procedure for transferring data from this temporary array to the appropriate axial level of the permanent array. Subroutine LEVELI is a generic procedure for transferring data for a specific axial level from the permanent array to this temporary array.

As an example of using subroutine LEVELR, all VESSEL mesh data input in subroutine RVSSL is read into the temporary array on a level-by-level and array-by-array basis. After each "read," as processed by the LOAD routine, the data are transferred from the temporary array to the axial level of the indicated permanent array via the RLEVEL routine, which calls the LEVELR procedure. The LEVELR procedure is also used directly from routine REVSSL to transfer data when reading the data-dump restart TRCRST file.

The LEVELI procedure for transferring data for a specific axial level from the permanent array to the temporary array is used by two output procedures: DLEVEL to write a restart data dump for one level and one array, and WLEVEL to write output data to the TRCOUT file for one level and one array.

Routines LEVELR, LEVELI, RLEVEL, DLEVEL, and WLEVEL are all generic routines and should not need to be modified unless the TRAC-M code developer wishes to make a major change in implementation.

2.2.2.3. Classification of array variables. Two basic categories of array variables are present in the VESSEL hydrodynamic database: single-time and dual-time array variables. Both categories have subcategories leading to seven classes of array variables:

1. Single-time array variables:
   - 1.1 Single-time, cell-centered (but not old-old-time) array variables that are either cell-centered, defined at the higher numbered cell faces, or defined at the lower numbered radial (x-direction) or axial cell faces.
   - 1.2 Old-old-time array variables that store values at the start of the previous timestep to create an ad hoc "triple-time" capability.
   - 1.3 Single-time, cell-face array variables defined at the backwards or lower-numbered azimuthal sector or y-direction cell face.
2. Dual-time array-variable pairs:
   - 2.1 Old-time array variables for which the new-time values are calculated before the OUTER hydrodynamic stage.
   - 2.2 Old-time array variables for which the new-time values are not calculated before the OUTER hydrodynamic stage.

2.3 New-time array variables for which their values are calculated before the OUTER hydrodynamic stage.

2.4 New-time array variables for which their values are not calculated before the OUTER hydrodynamic stage but may have been incorrectly calculated during OUTER before a backup.

The class of an array variable is determined according to how the array variable needs to be updated as the calculation progresses. Currently, no provision exists for an array variable belonging to more than one class.

Single-time array variables in Class 1.1 do not need to be automatically updated. This does not mean necessarily that their values don't change with time. Single-time array variables in Class 1.2 (currently only the gas volume fraction) are updated in subroutines TIMUPD and BAKUP in a manner analogous to that for dual-time array variables as described below. Single-time array variables in Class 1.3 require special logic, implemented in subroutine SETBDT, to ensure that values defined for azimuthal or y-direction phantom cells have the proper identification with the values for the actual cells.

Dual-time array variables are automatically updated, i.e., the old-time array variables take on the values of new-time array variables at the start of a timestep calculation. This coding is in subroutine TIMUPD for the VESSEL (note that this is the only mechanism for defining old-time values). In addition, the provision for separate classes of dual-time array variables allow for the code to back up (repeat a calculation with a different timestep size or other parameter) starting either at the beginning of a timestep or at the beginning of the OUTER hydrodynamic stage. Both backup procedures are in subroutine BAKUP. The differences in the two types of backups are discussed more thoroughly in Sec. 3.4.

Although an in-depth discussion of implementing the generic procedures applied to the different classes of array variables is outside the scope of this section, two aspects of the implementation affect the addition of array variables: the current implementation uses the relative position of an array variable in the database to determine its classification, and the relative positions of the array variables are known to the code through six parameters that rely on the database having a certain structure. In other words, the code developer must insert a new array variable in a position appropriate to its class and must ensure the maintenance of the assumed structure.

The relative position in memory of a mesh-wise array variable is referred to here either as its position or as its position in the database. This position is specific to its location in the vssArCom COMMON block defined in the include file EQUIV.H. In any case, the use of the word *position* here refers to the relative position of the array in computer memory when TRAC-M is executed.

Implementation procedures used for the VESSEL 3D database rely on a particular structure. This leads to a number of restrictions that must be observed when the code is modified by inserting an array variable. The major restrictions are related to the classification of array variables and are discussed in the next section. Special restrictions on the elements of array variables are given in Sec. 2.2.2.3.2, and some miscellaneous restrictions are given in Sec. 2.2.2.3.3.

2.2.2.3.1. Relation of position and classification. The current implementation of the generic procedures described above relies on the fact that the various classes of the VESSEL database are in the following order in the include file EQUIV.H, according to the position of the array variables in the class:

1) 1.1 [arrays HLA to SCD3M except ALPO] and 1.2 [array ALPO] (may be intermixed)
2) 2.1 [arrays BIT to CONCO]
3) 2.2 [arrays PA to OWLXR]
4) 2.3 [arrays BITN to CONC] (in one-to-one correspondence with 2.1)
5) 2.4 [arrays PAN to WLXR] (in one-to-one correspondence with 2.2)
6) 1.3 [arrays SPIFZ to SCD2M].

Because the implementation makes implicit use of these restrictions, it is essential that array variables that are added to include file EQUIV.H conform to these restrictions. TRAC-M allows for Class 1.1 array variables immediately before the Class 1.3 array variables. We do not recommend doing this because it complicates code maintenance. These particular restrictions were chosen to simplify implementing the generic procedures and to allow these procedures to be efficient on vector processors.

2.2.2.3.2. Special restriction on ordering components of array variables. For a subset of the cell-face array variables, the coding relies on the three components of the cell-face arrays being adjacent in memory and being ordered with the azimuthal-sector or y-direction component first, the axial or z-direction component second, and the radial-ring or x-direction component third. For example, in the include file EQUIV.H,

```
REAL*8 . . . ,fayt(ni,nj,nk),faz(ni,nj,nk),faxr(ni,nj,nk), . . .
COMMON /vssArCom/ . . .,fayt,faz,faxr, . . .
```

This restriction also applies to the cell-face array variables referenced in subroutine SVSET3 for evaluating signal variables. Consequently, insertion of new variables must not change the relative order of the components for these cell-face array variables. We recommend, for readability as well as for prevention of future coding errors, that all cell-face array variables be stored so that the components are adjacent and ordered as above.

2.2.2.3.3. Miscellaneous restrictions on positions of VESSEL array variables. Coding in subroutine SVSET3 for evaluating signal variables for VESSEL parameters relies on variable HLA being the first array variable. We are not aware of any other restrictions other than those listed here explicitly. However, if new variables are added, we recommend that they not be put as the first variable of their class. Code developers familiar with TRAC-M have assumed that they can depend on those variables that are now first in their class to remain in that relative position.

2.2.2.4. Referencing 3D arrays for VESSEL coding. All VESSEL hydrodynamic routines are coded in TRAC-M with direct use of 3D arrays for the mesh data, i.e., ALP(I,J,K). This implementation considerably reduces the possibility of coding errors. Naturally, with typical TRAC noding, this use of 3D arrays does not provide long vector lengths for inner do-loops. TRAC-M has been coded with the loop over axial levels as the innermost loop because that dimension is generally the largest. Achievement of long vector lengths by looping over the entire mesh would require a change to indirect addressing to encode the mesh connectivity in a vectorizable manner.

Reference to neighboring cells in the VESSEL mesh is straightforward using 3D arrays. From the standpoint of the cell at (I,J,K), the adjacent cell in the inner radial or x direction is (I–NV,J,K) and in the outer radial or x direction is (I+NV,J,K). The necessity for the stride, NV, arises from the cell-wise data storage of TRAC-P. TRAC-M defines NV=1 for mesh-wise data storage. The adjacent cell in the lower azimuthal or y direction is (I,J–1,K) and in the higher azimuthal or y direction is

(I,J+1,K). Finally, the adjacent cell in the lower axial or z direction is (I,J,K–1) and in the higher axial or z direction is (I,J,K+1).

Having an abstract method for referencing individual array variables also is convenient. Such a reference is currently used to implement the signal-variable evaluation logic in subroutine SVSET3. Pointers are used for this purpose for 1D data. For 3D data, we have chosen to use the subroutine getVSAR to access the (I,J,K) cell value of an array with name vsvName.

The positions of the VESSEL 1- and 2D array variables in the container A array are defined dynamically in subroutine PNTVSS. Their pointers are stored in COMMON block PTAB of include file VSSLPT.H.

Unless the TRAC-M user is adding a new variable to the signal-variable evaluation logic, it is not necessary to define an identifier for the variable in PTRS and to add the identifier to include file VSSLPT.H. To minimize changes to the code as well as minimize the amount of unused code, we recommend that identifiers not be added unless they are to be used.

2.2.2.5. Boundary or phantom cells. The VESSEL mesh in TRAC-M is constructed with two planes of boundary cells outside the mesh in each of the three lower-numbered directions and with one plane of boundary cells in each of the higher-numbered directions. The extra plane in the lower-numbered directions is necessary to accommodate face-centered data. The number of boundary cells in each direction is set by PARAMETER constants as described in Sec. 2.2.2.2.1. The use of boundary cells allows all references from cells within the physical mesh to neighboring cells outside the physical mesh to be valid.

When using a 3D VESSEL component to model a typical cylindrical-geometry reactor vessel with outer-boundary walls, the data in the bottom and top axial-boundary cells and in the outer radial-boundary cells do not affect the calculation. However, the inner radial boundary cells can be used to incorporate the effect of radial-momentum convection across the center of the VESSEL. Such a model was implemented using a different mechanism in TRAC-PF1/MOD1. This model, which is partially implemented in subroutine VRBD, is not currently activated in TRAC-M. The azimuthal-boundary cells are used to avoid the special logic necessary to indicate that the first physical azimuthal sector is adjacent to the last physical azimuthal sector. This is accomplished by subroutine SETBDT, which copies the data from the cells in the first and last physical sectors to their appropriate phantom cells.

The boundary-cell implementation makes it simple to include generalized boundary conditions at the bottom- and top-axial and outer-radial boundaries of a cylindrical VESSEL and at all external boundaries of a 3D Cartesian-geometry VESSEL. TRAC-M contains the appropriate coding in all VESSEL hydrodynamic routines to allow for fixed-pressure (BREAK) or fixed-velocity (FILL) boundary conditions at any of these boundaries. However, this coding for the radial or x and azimuthal or y boundaries has not yet been tested. In the currently released version of TRAC-M, there is no input-data mechanism to activate this coding. An input option, IVSSBF, only activates the generalized boundary conditions at the lower and upper axial faces. There currently is no coding to allow for the generalized boundary conditions to be time dependent. However, implementing such a capability should not require major changes to TRAC-M.

In addition to providing for the new generalized boundary conditions, using phantom cells allows for improved implementation of the standard hydrodynamic algorithms. Without the use of phantom cells, special program logic is required to calculate expressions that include gradients and fluxes for cells at the edge of the physical mesh. Such logic would increase the probability of coding errors and inhibit vectorization on hardware such as a Cray computer.

For typical coarse-mesh VESSEL components, a large percentage of the cells are at the edges of the mesh. For example, a 3D VESSEL component with four radial rings and four azimuthal sectors on each level actually has only 4 of the 16 cells on a level that has neither a radial nor an azimuthal boundary. Because straightforward vectorization generally reduces computation time by more than a factor of 5, it is obviously desirable to design implementations that are vectorizable for all cells.

As stated previously, if phantom cells are not used, special logic would be required to carry out calculations for cells at the edge of the physical mesh. On the other hand, when phantom cells are used, additional procedures are required to define the values associated with the phantom cells. The amount of code that must be maintained is similar in either case; however, the phantom-cell methodology is more easily modularized.

The major disadvantage in using phantom cells is the potential for significantly increased computer-memory requirements for coarse-mesh VESSEL components. In our previous example, a VESSEL with 4 radial rings, 4 azimuthal sectors, and 10 axial levels has only 4 x 4 x 10 or 160 physical mesh cells. However, it will have (4 + 3) x (4 + 3) x (10 + 3) or 637 computational mesh cells when including the boundary cells. Naturally, the percentage of boundary cells is smaller for more finely noded problems. The current VESSEL array data contains ~300 different variables; thus, this example would require ~200,000 words of computer memory for the 48,000 words of physical mesh-cell array data. For most modern computer hardware, however, this is not a large amount of memory, and the cost-benefit ratio of this memory increase is extremely favorable when compared with more efficient coding.

Because both of the lowest-numbered planes of phantom cells in each direction only are used in conjunction with the generalized boundary-condition option associated with a fixed-pressure boundary condition, it should be possible to reduce the memory requirements by changing from 2 to 1 the PARAMETER constants defining the number of lower-numbered phantom cells for the radial or x and azimuthal or y directions. However, doing so has not been tested.

2.2.2.6. Adding or deleting a 3D database array variable. The three steps to adding an array variable to the VESSEL hydrodynamic (mesh-wise) database in COMMON block vssArCom of include file EQUIV.H are summarized below. Note that these steps are incomplete for the case of an old-old-time array variable because the new array variable would replace ALPO as either the first or last array variable in that classification.

1. Determine an appropriate position in the database for the new array variable or dual-time array variable pair according to the classification of the array variable and the structure of the database (see Sec. 2.2.2.3.1).
2. Add the new array-variable name or dual-time array-variable pair names at the appropriate position/s in COMMON block vssArCom of include file EQUIV.H.
3. Add the new array-variable name or dual-time array-variable pair names at the appropriate position/s in the REAL*8 declaration of include file EQUIV.H.

Once a new VESSEL array variable has been successfully added to the VESSEL hydrodynamic database, one then needs to modify the necessary subroutines to calculate, dump/restart, or output the new array variable. The following three guidelines give step-by-step instructions on how this should be accomplished.

1. Program the necessary evaluations to determine the value of the new array variable within the appropriate subroutines.
2. If the new array variable needs to be written to the data-dump TRCDMP file for restart purposes, include a call to subroutine DLEVEL in subroutine DVSSL. In addition, increment the variable LV by one in subroutine DVSSL. To read in the new array variable from the data-dump TRCRST file when restarting, add calls to subroutines BFIN and LEVELR in subroutine REVSSL in the same position that the call was added to subroutine DVSSL (note that REVSSL must be changed if DVSSL is changed).
3. If the new array variable is to be output to the TRCOUT file, add a call to subroutine WLEVEL in subroutine WVSSL to output the new array variable.

3. TRAC-M CALCULATIONAL SEQUENCE

The full TRAC-M calculational sequence involves several stages: input processing; initialization; prepass, outer-iteration, and postpass calculations; timestep advancement or backup; and output processing. Each of these stages is discussed in greater detail from a programmer's point of view in the sections that follow. First, a summary of the overall calculational sequences for transient and steady-state calculations is given.

3.1. General Summary

Depending on the values of the input parameters STDYST and TRANSI (Main-Data Card 4), TRAC-M performs a steady-state calculation, a transient calculation, or both. The general control sequences of each type of calculation are outlined below, and specific details of the calculational sequence are discussed in more detail in the subsections that follow.

A transient calculation is directed by subroutine TRANS. The system state is advanced a timestep through time by a sequence of prepass, outer-iteration, and postpass calculations that TRANS requests by calling subroutines PREP, HOUT, and POST, respectively. In each of these calculations, one or more sweeps are made through all the components in the system model. To provide output results required by the user, TRANS invokes the EDIT and DUMP modules by calling subroutine PSTEPQ. Subroutine TRANS is structured, as shown in Fig. 2. The major control variables within the timestep loop are: NSTEP, the current timestep number; TIMET, the time since the transient began; DELT, the current timestep size;

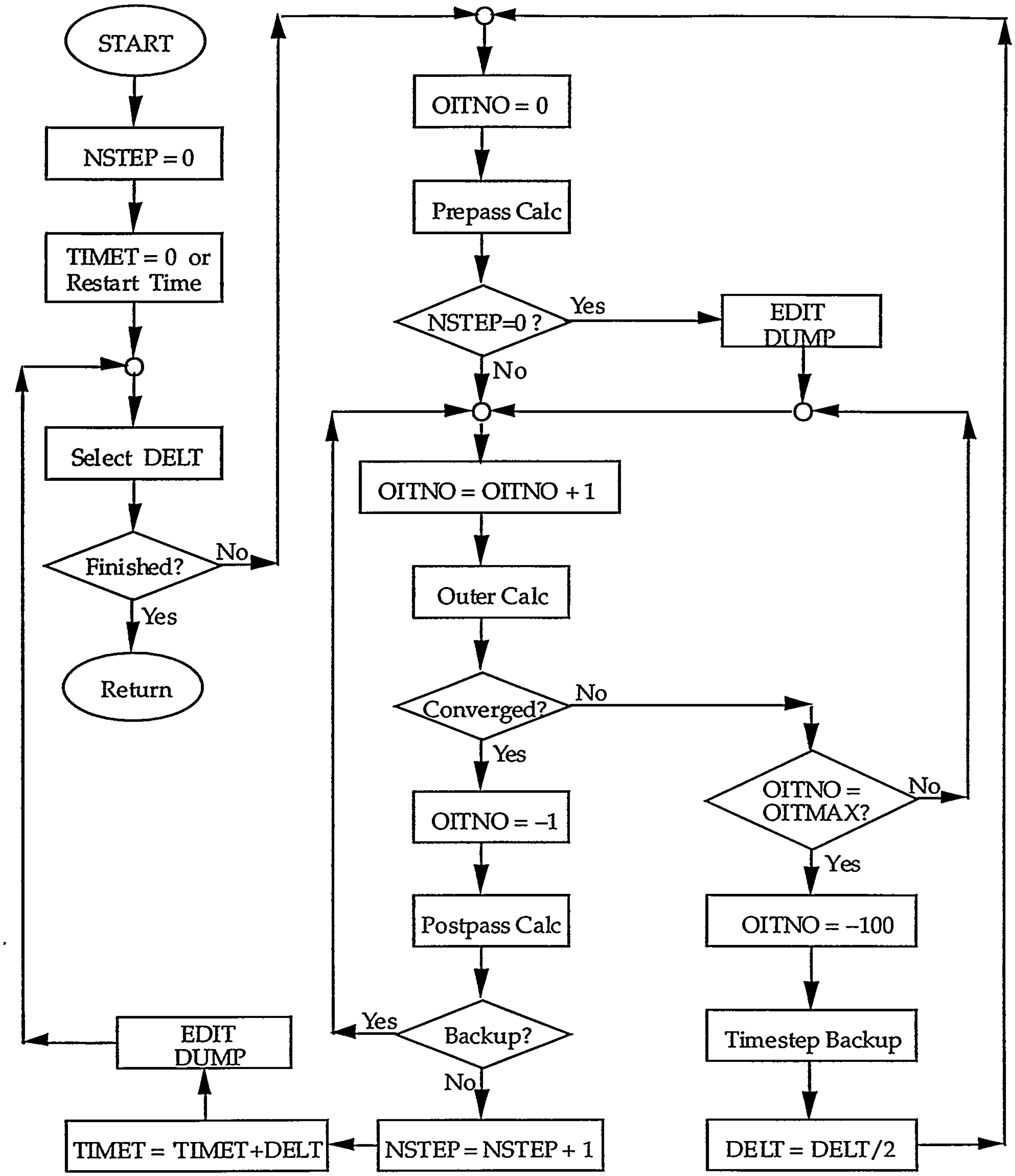

Fig. 2.
Transient calculation flow diagram.

and OITNO, the current outer-iteration number. The timestep loop begins with the selection of the timestep size, DELT, by subroutine TIMSTP. A prepass is performed for each component by module PREP to evaluate the control parameters, stabilizer motion equations, and phenomenological coefficients. At this point, if the current timestep number is zero, TRANS calls the EDIT module to print the system-state parameter values and the DUMP module to output a restart-data dump at the beginning of the transient. Subroutine TRANS then calls subroutine HOUT, which performs one or more outer iterations to solve the basic hydrodynamic equations. Each outer iteration is performed by module OUTER and corresponds to one iteration of a Newton-method solution procedure for the fully coupled difference equations of the flow network. The outer-iteration loop ends when the outer-iteration convergence criterion (EPSO on Main-Data Card 5) is met. This criterion requires that the maximum fractional change in the pressure throughout the system during the last iteration be less than or equal to EPSO.

Alternatively, the outer-iteration loop may terminate when the number of outer iterations reaches a user-specified limit (OITMAX on Main-Data Card 6). When this happens, TRAC-M restores the thermal-hydraulic state of all components to what it was at the beginning of the timestep, reduces the DELT timestep size (with the constraint that DELT be greater than or equal to DTMIN), and continues the timestep calculation with the new reduced timestep size. This represents a backup situation and is discussed in greater detail in Sec. 3.5.

When the outer iteration converges, TRANS calls module POST to perform a postpass evaluation of the stabilizer mass and energy equations and the heat-transfer calculation. Then the NSTEP timestep number is incremented by 1, and the TIMET problem time is increased by DELT. The calculation is finished when TIMET reaches the last TEND time specified by the timestep data.

The transient calculation is controlled by a sequence of time domains input specified by the timestep data. During each of these time domains, the minimum and maximum timestep sizes and the edit, dump, and graphics time intervals are defined. When the EDIT and DUMP modules are invoked, they calculate the time when the next output of the associated type is to occur by incrementing the current time by its time interval. When TRANS later finds that TIMET has reached or exceeded the indicated time, the corresponding output modules are invoked again. Whenever TIMET equals or exceeds the TEND ending time for a timestep data domain, the next timestep data domain is read in. The output indicators are set to the current time plus the new values of the appropriate time intervals.

Subroutine STEADY directs steady-state calculations using the structure shown in Fig. 3. The calculation sequence of subroutine STEADY is similar to that of the transient driver subroutine TRANS. The same sequence of evaluations used for a transient calculation also is used for a steady-state calculation. The main difference in STEADY is the addition of a steady-state convergence test, logic to turn on the steady-state power level, an optional evaluation of constrained steady-state controllers, and an optional hydraulic-path steady-state initialization of the initial hydraulic state estimate. To provide output results, STEADY like TRANS invokes the EDIT and DUMP modules by calling subroutine PSTEPQ.

Subroutine STEADY is called by the TRAC main program whether or not a steady-state calculation has been requested by STDYST. If no steady-state calculation is to be done because STDYST = 0, STEADY returns to the TRAC main program. The TRAC main program then calls TRANS and performs a transient calculation if ITRANS = 1.

Timestep control in STEADY is identical to that implemented in TRANS. This includes the selection of the timestep size, the timing for output, and the backup of a timestep if the outer-iteration limit is exceeded. In STEADY, the input variable SITMAX (from Main-Data Card 6) is the maximum number of outer iterations used in place of OITMAX. The maximum fractional rates of change per second of seven thermal-hydraulic parameters are calculated by subroutines TF1DS3 and FF3D. These rates and their locations in the system model are passed to STEADY through the array variables FMAX and LOK in COMMON block SSCON of include file SSCON.H. Tests for steady-state convergence are performed every 5 timesteps and before every large edit. The maximum fractional rates of change per second and their locations are written to the TRCMSG and TRCOUT files, as well as the terminal. The minimum value of the flow velocity, MINVEL, and its maximum fractional rate of change, FMXLVZ, in the hydraulic channels coupled to powered heat structures determine when the steady-state power should be set on. Once MINVEL exceeds 0.5 m/s and FMXLVZ falls below 0.5, the steady-state power is set to its input steady-state value RPOWRI for each powered heat structure. The steady-state calculation is completed when all maximum fractional rates of change per second are below the user-specified convergence criterion EPSS (from Main-Data Card 5) or when STIME reaches the TEND end time of the last time domain specified in the steady-state calculation timestep data.

There are three types of steady-state calculations: generalized (as described above), constrained, and static check. A constrained steady-state calculation (CSS) is

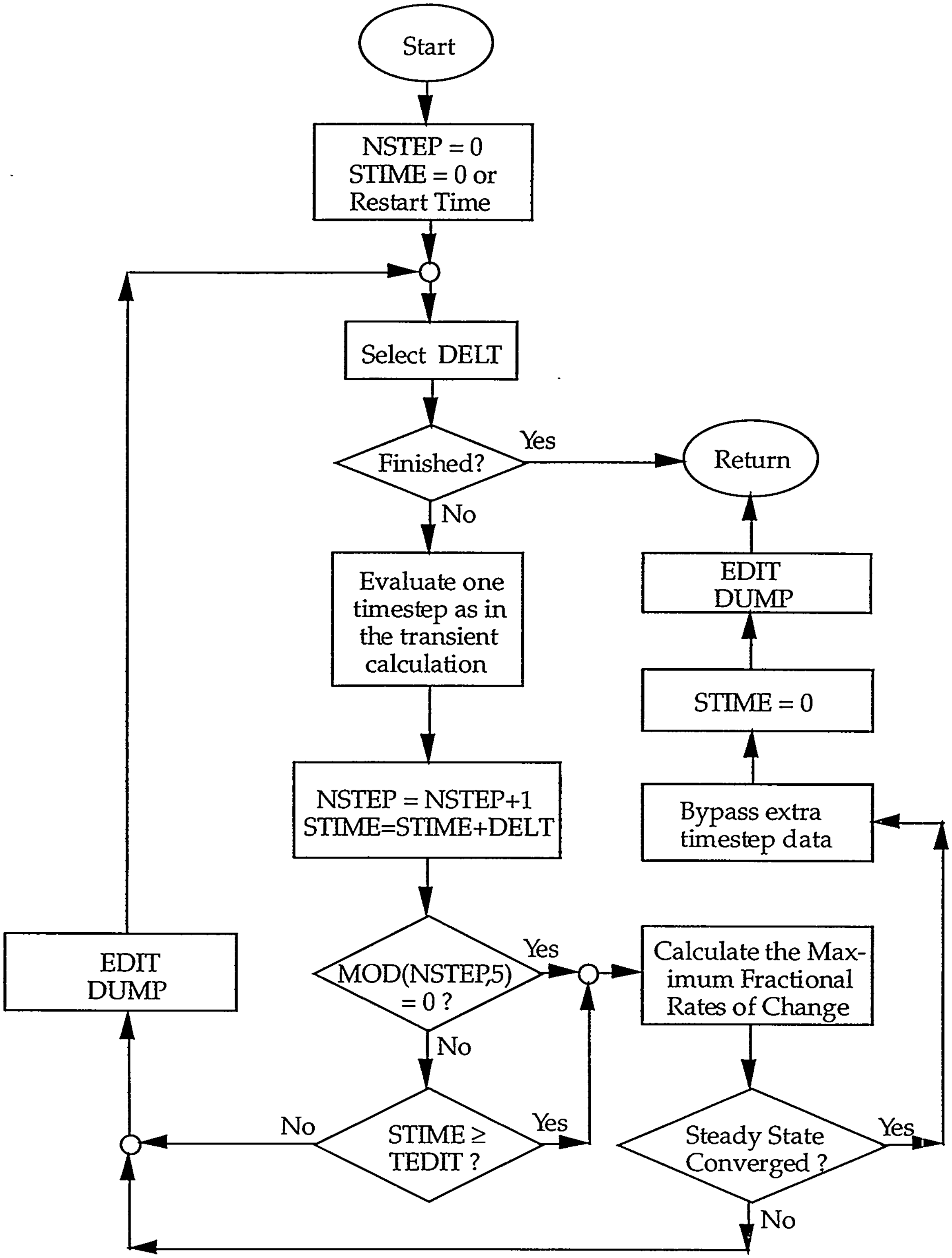

Fig. 3.
Steady-state calculation flow diagram.

a generalized steady-state calculation (GSS) with input-specified controllers adjusting uncertain component-hardware actions to achieve known or desired steady-state, thermal-hydraulic conditions. A static-check, steady-state calculation (SSS) checks for erroneous momentum and energy sources in the system model by having TRAC-M internally set the PUMP momentum source to zero and not evaluate heat transfer with the expectation that any input fluid flow becomes stagnant and temperatures don't change.

Both steady-state and transient calculations may be performed during one computer run. The end of the steady-state timestep cards is signified by a single card containing a –1.0 in columns 4 to 14. The transient timestep cards should follow immediately. If the steady-state calculation converges before reaching the end of its last time domain, the remaining steady-state timestep data are read in but are not used so that the transient calculation proceeds as planned with its own timestep data.

3.2. Input Processing

The processing of all TRAC-M input data (except for the timestep data) is performed by the INPUT module and its sub-modules RDIN1, RDIN3, and RDRES. The data are of two types: input data retrieved from the input-data file TRACIN and restart data retrieved from the dump-restart file TRCRST. In addition to reading the input data, these modules also organize the component data in memory, assign the array pointer variables for each component, analyze the system-model loop structure, and allocate the initial container-A-array space for part of the global data. The remainder of the space necessary within the A array for the global variables is allocated by subroutine INIT in module INIT

Subroutine INPUT (the INPUT-module driver) reads the Namelist, main-data, and countercurrent-flow-limitation (CCFL) model input data from the TRACIN file. The initial A-array global array variable space is allocated using main-data parameter information. Hydraulic-path, steady-state initialization and CSS-controller data are then input if these options are selected in the main data. The signal-variable, control-block, and trip control-parameter data are read and processed by calling subroutine RCNTL. Subroutine RDCOMP (module RDIN1) reads and processes the 1D hydraulic-component data, and subroutine RDCOM3 (module RDIN3) performs a similar function for the VESSEL-component data from the TRACIN file. Any control-parameter and component data not provided by the TRACIN file are retrieved from the restart-data TRCRST file by subroutine RDREST (module RDRES). Finally, subroutine INPUT calls subroutine SRTLP to establish hydraulic loops and pointers for the network solver and calls subroutine ASIGN to define the component data-block pointer array COMPTR.

Subroutine RDCOMP calls component input-processing subroutines to read and process each component type. These routines have names that begin with the letter R followed by the letters of the component-type name. For example, the PIPE-component input-processing subroutine is named RPIPE. In addition to reading hydraulic and heat-structure component data from the TRACIN file, these component input-processing routines also initialize the generic-component, specific-component, and pointer tables and define the JUN array with component-junction connective information. Each 1D hydraulic-component input-processing subroutine calls subroutine RCOMP to process input data common to 1D hydraulic components. All input data are echoed as output to the TRCOUT file.

Pointer variables common to 1D hydraulic components are initialized with a call to subroutine S1DPTR. Any additional pointers special to a component type are initialized within that component's input-processing subroutine. An example of specialized pointer variables are those for array variables defining component actions for the component type. When adding a new array variable to a 1D hydraulic component, it is necessary to initialize its new pointer in S1DPTR or in the component input-processing routine, in addition to performing several other steps. The step-by-step procedure involved is discussed in Sec. 2.2.1.1, and an example update that adds five new array variables and their pointers to all 1D hydraulic components in TRAC-M is presented in Appendix E.

The JUN array, defined by the component input-processing routines, is a doubly subscripted array, JUN(4, 2*NJUN). The four values of the first index are defined in Table 2. The second index indicates the order in which the component junctions were encountered during input processing.

Subroutine RDCOM3 calls the VESSEL-component input-processing subroutine RVSSL. In addition to reading VESSEL input-data parameters from the TRACIN file, this routine also initializes the generic-component tables, specific-component tables, and pointer tables and reads VESSEL level data and performs input-data testing.

Subroutine RDREST opens file TRCRST and obtains restart data from the data-dump edit corresponding to the requested timestep number of a previous calculation (as specified by variable DSTEP on Main-Data Card 3 of file TRACIN). If

TABLE 2

FIRST INDEX OF THE COMPONENT-JUNCTION ARRAY, JUN

Index	Description
1	Junction number.
2	Component number.
3	Component-type number.
4	Junction direction flag. 0 = positive flow is into the component at this junction (a JUN1 junction); 1 = positive flow is out of the component at this junction (a JUN2 or JUN3 junction).

the requested timestep number is negative, RDREST uses the last data-dump edit available. If the requested timestep number is –99, the problem time from the last data-dump is replaced by TIMET read from file TRACIN. The restart data initialize the signal-variable, control-block, trip, and component data that were not provided by the TRACIN file. Component data are read from the TRCRST file by calls to component restart-processing subroutines. These subroutines have names that begin with the letters RE followed by the letters of the component-type name. They function in much the same way as the component input-processing subroutines that begin with the letter R. For example, the PIPE-component restart-processing subroutine is called REPIPE. The restart data common to 1D hydraulic components are processed from the restart data using a call to subroutine RECOMP. Details on the structure of the dump-restart TRCRST file are given in Sec. 3.6. All restart data are echoed in their input-data form as output to the TRCOUT file.

Subroutine SRTLP sorts through the 1D hydraulic components of the system model and groups them by loops that are isolated from one another by VESSEL components or TEE-component internal junctions. The IORDER array is rearranged to reflect this grouping and to provide a convenient order within each group for the network solution procedure. The I^{th} element of the array IORDER is the number of the component that is processed after the $I{-}1^{th}$ component but before the $I{+}1^{th}$ component.

Subroutine ASIGN defines the component pointer array, COMPTR, according to the order of the IORDER array. The I^{th} element of array COMPTR is the starting location in the container A array of the component IORDER(I) data block containing its generic-component table, specific-component table, pointer table, and array data.

If the input-data file TRACIN is in FREE format (rather than in TRAC format), TRAC-M creates an additional file TRCINP. The TRACIN data are written into file TRCINP in the TRAC-format form that can be read by the TRAC-M input routines. File TRCINP (in TRAC format) is used as the input-data file rather than file TRACIN (in FREE format).

The user has the option of creating an echo file of the input data contained in file TRACIN by defining Namelist variable INLAB = 3. With this option, a file named INLAB (INput LABeled) is created during input-data processing and has all the input data from file TRACIN output to it along with variable-name comments contained between asterisks. This provides a useful means of labeling an otherwise difficult-to-interpret TRACIN file. It also allows the user to verify the input data being read by TRAC-M. Comments between asterisks in the TRACIN file are not output to the INLAB file.

All input data from files TRACIN and TRCRST are echoed to the TRCOUT file by subroutines READI, READR, REECHO, WARRAY, and WIARN that are called by the component input (Rcomp) and restart (REcomp) processing subroutines. The input and output echo of all input data has been consolidated in these five subroutines. SI- or English-unit symbols for real-valued input-data variables are output echoed to the TRCOUT file when Namelist variable IUNOUT = 1 (dafault value).

3.3. Initialization

During the initialization stage performed in module INIT, subroutine ICOMP performs the initialization of arrays and variables for each component that is required by TRAC-M but is not read in directly from files TRACIN and TRCRST.

The overall component-initialization subroutine ICOMP first defines the junction sequence array JSEQ and the velocity sign indicator array VSI and then initializes the data for heat-structure, 1D hydraulic, and 3D VESSEL components. The array JSEQ contains junction numbers in the order that they are processed as determined by the component order-of-evaluation array IORDER. The I^{th} element of the array VSI is the junction flow-reversal indicator for junction JSEQ(I). Using a call to subroutine SETNET, the array IOU is initialized to contain network junction numbers for the junctions of all components excluding BREAKs and FILLs. Finally, VESSEL source connections to 1D hydraulic components are checked to ensure that all connections for a particular loop are in the same coordinate direction as the VESSEL component/s they are coupled to. This is necessary to ensure that the predictor and stabilizer velocities solved by subroutines FEMOMX, FEMOMY, and FEMOMZ remain independent of one another for numerical stability at high fluid flows.

Subroutine CIHTST controls the initialization of all heat-structure components with calls to subroutines IRODL and IROD. Subroutine IRODL initializes arrays that provide information on the location of hydrodynamic data for heat-transfer coupling. Subroutine IROD initializes various power-related arrays that are not input.

The 1D hydraulic-component initialization routines have names that begin typically with the letter I followed by the letters of the component-type name. For example, the PIPE component initialization subroutine is called IPIPE. After determining the junction connection and component sequencing, these routines call subroutine VOLFA to calculate volume-averaged cell flow areas and to perform several input-data tests on valid flow-area configurations between cells and cell interfaces. Subroutine COMPI is called to initialize several variable arrays (e.g., tilde velocities). Thermodynamic properties, transport properties, and stabilizer quantities are initialized by calling subroutine IPROP. A call to subroutine SETBD initializes the boundary-array data. Junction-data consistency is checked using a call to subroutine CHKBD. Finally, subroutine ELGR is called to compute FRICs and GRAVs from input form losses and elevations if these particular input options are selected using the Namelist options IKFAC and IELV, respectively.

Component boundary data are stored in the doubly dimensioned array BD#(72,NJUN). The data define the current solution state of the adjacent component across the junction # and are evaluated at one of three possible space points: the edge of the mesh cell at the junction, the midpoint of that mesh cell, or the opposite-side edge of that mesh cell. Junction # = 1 corresponds to junctions JUN1 and JUN4 (the internal junction of a TEE component); junction # = 2 corresponds to junctions JUN2 or JUN3 (the external junction of the TEE-component side channel). The first dimension index indicates the parameter variable that is defined in subroutine J1D for 1D hydraulic components, subroutine BDPLEN for one-cell PLENUM components, and subroutine J3D for 3D VESSEL components. The second dimension index indicates the order in which the junction numbers are processed.

Subroutine CIVSSL controls the initialization of all 3D VESSEL components by calling subroutine IVSSL. Subroutine IVSSL performs analogous initializations for the VESSEL component as does subroutine IPIPE for the PIPE component. Obviously, using many of the same low-level subroutines for initializing both component types is not possible because of the differences in the 1- and 3D databases.

Before the above component initialization by subroutines ICOMP and CIVSSL begins, an optional hydraulic-path steady-state initialization procedure may be performed for a steady-state calculation by subroutine ICOMP calling subroutine IHPSS1 and subroutine CIVSSL calling subroutine IHPSS3. The procedure replaces the phasic temperature and velocity (and possibly pressure) values input for the 1D hydraulic and 3D VESSEL components with fluid mass-conserving and energy-conserving values based on input-specified known or estimated thermal-hydraulic flow conditions along 1D-flow hydraulic paths of the system model. The purpose of this procedure is to provide a better initial estimate of the thermal-hydraulic solution so that steady-state solution convergence is satisfied in fewer timesteps and less computational effort. This saves the TRAC-M user the effort of inputting such detail in the solution estimate defined by the component data so that the steady-state solution can converge quicker with a better initial-solution estimate.

3.4. Prepass, Outer-Iteration, and Postpass Calculations

One complete timestep calculation consists of a prepass, outer-iteration, and postpass stage. Each of these stages of the timestep calculation is described below.

3.4.1. Prepass Calculation. To evaluate numerous quantities to be used during the outer-iteration-stage and postpass-stage calculations, the prepass calculation uses the modeled-system solution state defined initially from input and later from the completion of the previous timestep (the beginning of the present timestep). The prepass stage begins by evaluating signal variables and control blocks and determining the set status of all trips of the control procedure. Each component begins the prepass by moving its end-of-timestep values (its new-time values) from the previous timestep into the variable storage for its old-time values for the present timestep. Next, wall and interfacial friction coefficients are evaluated. The predictor stabilizer velocities, as well as the forward elimination of the corrector stabilizer motion equations, are evaluated. The prepass evaluates material properties and heat-transfer coefficients (HTCs) for components that require heat-transfer calculations. A second pass through all 1D hydraulic components evaluates the backward substitution of the corrector stabilizer motion equations for the corrector stabilizer tilde velocities. The prepass for heat-structure components can be more complex. Besides calculating material properties and HTCs for both average and supplemental rods, the prepass evaluates quench-front positions and fine-mesh properties if the reflood model has been activated.

The prepass calculation is controlled by module PREP, whose entry-point subroutine is of the same name. Subroutine TRIPS (not to be confused with subroutine TRIP that interrogates a trip's set status to decide on initiating specific consequences, such as a component action, controlled by the trip) calls for the evaluation of signal variables, control blocks, and trips. Then subroutine PREP performs the IBKS = 1 first pass of the PREP stage for all 1D hydraulic components by calling PREP1D. All heat-structure components are processed by calling HTSTR1. If the SETS3D method has been selected for all VESSEL components (Namelist variable NOSETS = 0 or NOSETS = 2 and NSTAB=1), overlay PREP3D is called to evaluate the predictor velocities and the forward elimination of the corrector stabilizer motion equations. The IBKS = 2 second pass through the PREP stage performs the backward-substitution evaluation for the 1D corrector stabilizer tilde velocities by again calling PREP1D and the 3D corrector stabilizer tilde velocities by again calling PREP3D. If the SETS3D method is not selected (Namelist variable NOSETS = 1 or NSTAB = 0), the prepass is completed with a call to PREP3D to define all tilde velocities by their beginning-of-timestep basic velocities for the 3D VESSEL components.

Subroutine TRIPS calls subroutines SVSET, CBSET, and TRPSET. Subroutine SVSET uses beginning-of-timestep values of system-state variables to define the signal variables. Subroutines CBSET and CONBLK, which is called by subroutine CBSET, evaluates control-block function operators. Subroutine TRPSET uses the current signal-variable and control-block values to determine the set status of trips.

The prepass driver subroutine PREP1D calls 1D hydraulic-component prepass routines to perform both passes of the prepass for each 1D hydraulic-component type. The names of the prepass component driver routines end with the numeral 1, as shown in Table 3. For example, the PIPE component prepass subroutine is called PIPE1. On the IBKS = 1 first pass through the PREP stage, during which the predictor stabilizer velocities are evaluated and the corrector stabilizer motion equations are

TABLE 3

COMPONENT-DRIVER SUBROUTINES

Component Type	Prepass	Outer	Postpass
BREAK	BREAK1	BREAK2	BREAK3
FILL	FILL1	FILL2	FILL3
PIPE	PIPE1	PIPE2	PIPE3
PLENUM	PLENUM1	PLENUM2	PLENUM3
PRIZER	PRIZER1	PRIZER2	PRIZER3
PUMP	PUMP1	PUMP2	PUMP3
ROD or SLAB	HTSTR1		HTSTR3
SEPD or TEE	TEE1	TEE2	TEE3
TURB	TURB1	TURB2	TURB3
VALVE	VLVE1	VLVE2	VLVE3
VESSEL	VSSL1	VSSL2	VSSL3

forward eliminated, the 1D component subroutines utilize the common low-level routines SAVBD, PREPER, and SETBD to avoid redundant coding in the component-driver subroutines. On the IBKS = 2 second pass, during which the stabilizer motion equations are backward substituted and solved for the corrector stabilizer tilde velocities, the common low-level routine BKMOM is called by the component-driver subroutines.

Subroutine SAVBD retrieves BD-array boundary data from adjacent components, stores it in appropriate array locations, and moves data from the last completed timestep into the old-time arrays. Subroutine PREPER evaluates wall friction by calling FWALL, material properties by calling MPROP, HTCs by calling HTPIPE, and interfacial-shear coefficients by calling FEMOM, and begins the forward-elimination solution of the stabilizer motion equations by calling FEMOM. For a specific component, any or all steps may occur during a call to PREPER by its component prepass driver routine. Subroutine SETBD uses the information in the component data arrays to reset the BD-array boundary data for both ends of the component. Subroutine BKMOM solves the stabilizer equations of motion by backward substitution for the stabilizer corrector tilde velocities for 1D hydraulic components.

Subroutine HTSTR1 calls subroutine FLTOM to transfer hydrodynamic data into the necessary heat-structure arrays; subroutine CORE1 to evaluate HTCs, fine-mesh properties, and quench-front positions; and subroutine FLTOM again to transfer heat-transfer information back into the hydrodynamic database. From subroutine CORE1, subroutine RFDBK is called to evaluate reactivity feedback, and subroutine RKIN is called to evaluate the point-reactor kinetics model.

Subroutine VSSL1 controls the prepass evaluation of each VESSEL component. A new-time to old-time variable update is performed by calling subroutine TIMUPD. Donor-cell weighting factors are initialized, vent-valve calculations are performed, and momentum source terms are defined. Subroutine CIF3 is called to evaluate the interfacial shear coefficients. Subroutine PREFWD is called to evaluate the wall-shear coefficients. Subroutines FEMOMX, FEMOMY, and FEMOMZ are called to evaluate the 3D predictor and corrector stabilizer tilde velocities. Finally, subroutine J3D is called to update the BD-array boundary information.

3.4.2. Outer-Iteration Calculation. The hydrodynamic state of the modeled system is analyzed in TRAC-M by a sequence of Newton iterations that use direct inversion of the linearized equations for all 1D hydraulic-component loops and the VESSELs during each iteration. Throughout the sequence of iterations that constitute an outer calculation (each called an outer iteration within TRAC-M), the properties evaluated during the prepass stage and the previous-timestep postpass stage remain fixed. Such properties include wall (SLAB and ROD) temperatures, HTCs, wall- and interfacial-shear coefficients, stabilizer tilde velocities, and quench-front positions. The remaining fluid properties can vary to obtain a consistent hydrodynamic-model solution.

Each call to module OUTER completes a single outer (Newton) iteration. Subroutine HOUT, which is the entry-point routine of this module, controls the overall structure of an outer iteration, as shown in Fig. 4.

Both the forward-elimination and backward-substitution sweeps through the 1D hydraulic-component loops are performed by subroutine OUT1D and associated outer-iteration routines. The calculations that these routines perform are controlled by the variable IBKS, which is set by subroutine OUTER. Subroutine OUT3D solves the hydrodynamic equations for all VESSEL components (IBKS = 0) and updates boundary data (IBKS = 1).

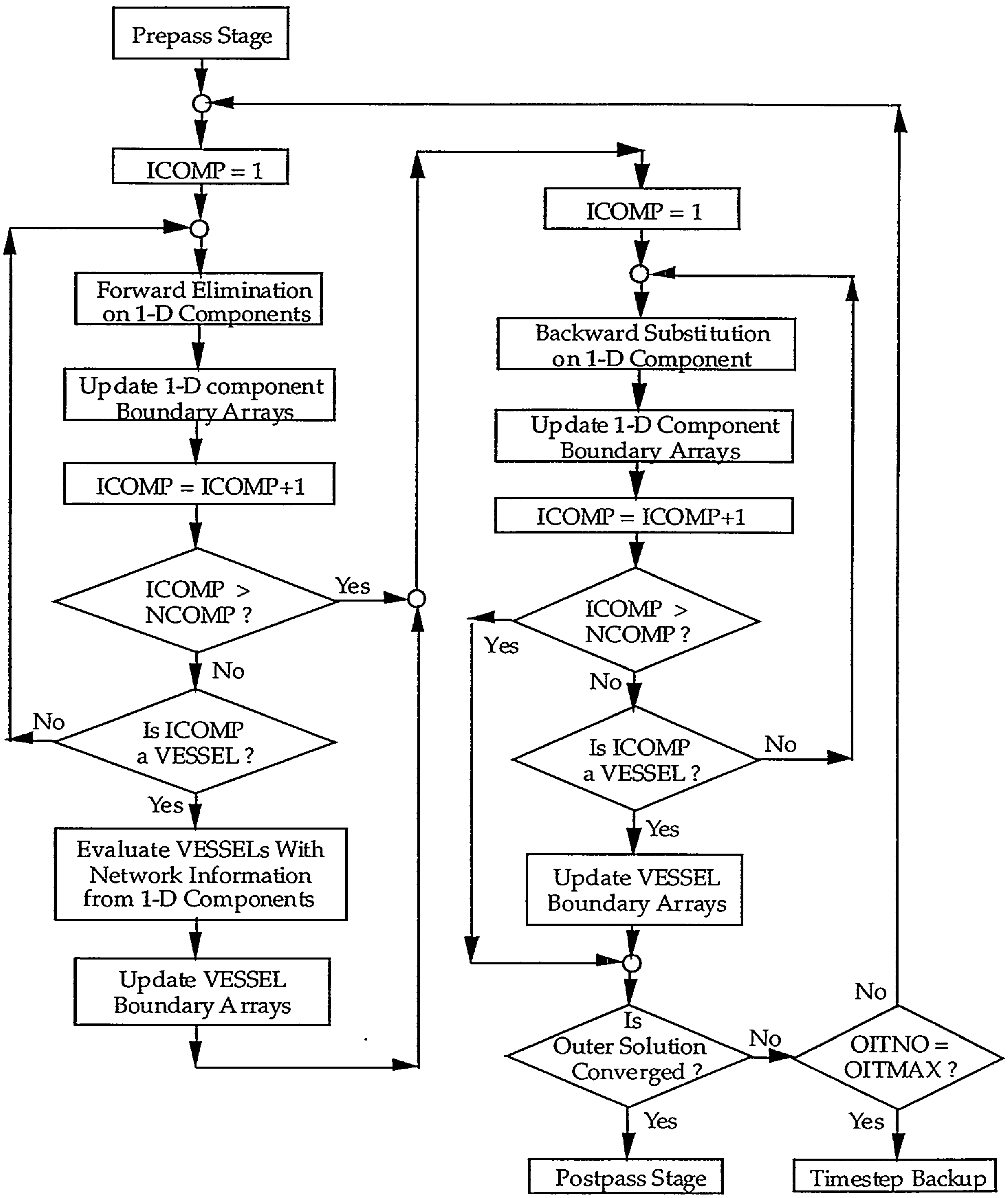

Fig. 4.
Outer-iteration calculation flow diagram.

All 1D hydraulic components in a particular loop are handled by a single call to subroutine OUT1D. This routine loads the data blocks for a component into memory and then calls the appropriate component outer-iteration subroutine. Component outer-iteration subroutines have names that end with the numeral 2, as shown in Table 3. For example, the PIPE component outer-iteration subroutine is called PIPE2. Subroutine OUT3D functions in a similar manner, except that subroutine VSSL2 is called for each 3D VESSEL component.

The outer-iteration subroutines for 1D hydraulic components call subroutine INNER to perform common functions. Subroutine INNER retrieves boundary information from the BD boundary array, tests other boundary information for consistency, calls subroutine TF1D to perform the appropriate hydrodynamic calculation, and resets the BD boundary array by calling subroutine J1D. Subroutine TF1D calls subroutines TF1DS1 (first outer-iteration only), TF1DS, and TF1DS3 to solve the basic semi-implicit finite-difference equations.

Subroutine VSSL2 solves the basic semi-implicit, finite-difference equations defined by the VESSEL-matrix equation (depending on the value of IBKS) for a single VESSEL component problem, whereas subroutine OUT3D does the same for a multi-VESSEL component problem. Subroutines TF3DS1 and TF3DS are called to linearize the hydrodynamic basic semi-implicit, finite-difference equations. Subroutine STDIR sets up the VESSEL-matrix equation for direct inversion. Subroutine MATSOL is called to solve the linear-system VESSEL-matrix equation using the capacitance-matrix method. Subroutine BACIT stores the new-time pressures that are evaluated.

3.4.3. Postpass Calculation. After the modeled-system hydrodynamic state has been evaluated by a sequence of outer iterations that have converged, TRAC-M performs the postpass stage to solve the stabilizer mass and energy equations and to evaluate both fluid mixture properties and component wall temperatures. Module POST performs this postpass stage. The same module also implements the timestep backup procedure, which is explained in detail in the next section.

Subroutine POST, as the controlling subroutine for this module, first processes all 1D hydraulic components by calling the appropriate 1D hydraulic-component postpass subroutine, whose name ends with the numeral 3, as shown in Table 3. For example, the PIPE-component postpass subroutine is called PIPE3. Subroutine POST3D is called to process all 3D VESSEL components, and subroutine HTSTR3 is called to process all heat-structure components.

The 1D hydraulic-component postpass subroutines use the low-level routines SAVBD, POSTER, and SETBD to retrieve BD-array boundary conditions; to evaluate the stabilizer mass and energy equations, component wall temperatures, fluid mixture properties, and fluid transport properties; and to reset the BD boundary array, respectively.

The VESSEL postpass routine, VSSL3, is called by POST3D for each VESSEL component. Within subroutine VSSL3, stabilizer mass and energy quantities are evaluated by subroutine BKSTB3 or defined by subroutine MIX3D, depending on the status of the VESSEL SETS3D-method flag NSTAB. Subroutines FF3D, FPROP, and J3D are used to complete the hydrodynamic calculation, to evaluate fluid transport properties, and to update BD-array boundary data, respectively.

Subroutine HTSTR3 controls the postpass by calling subroutine CORE3 for each heat-structure component. In subroutine CORE3, subroutine FROD is called to evaluate the temperature distribution and gap heat-transfer coefficients by calling subroutines RODHT and GAPHT, respectively.

3.5. Timestep Advancement and Backup

Upon the successful completion of a timestep calculation (evaluated by the prepass, outer-iteration, and postpass stages), the modeled-system solution state defines the end-of-timestep new-time conditions. At the start of the next timestep's PREP stage, the previous timestep's new-time condition defines the next timestep's old-time condition. This is handled on a component-by-component basis within the component "1" subroutines, i.e., PIPE1. During this step, all dual-time variables are updated by copying the values of the new-time array variables into the old-time array variables. The prepass, outer-iteration, and postpass stages that follow during the next timestep then attempt to evaluate new values for the new-time array variables for the end-of-timestep condition. This process is repeated as problem time advances with each timestep calculation.

Calculation of a new timestep size takes place just before the PREP stage and is controlled by subroutine TIMSTP. Two types of algorithms, inhibitive and promotional, are implemented in subroutine NEWDLT to evaluate the next timestep size. The inhibitive algorithms limit the new timestep size to ensure stability and to reduce finite-difference error. The promotional algorithm increases the timestep size to improve computational efficiency (by requiring fewer timesteps during a time interval). A new maximum timestep size is calculated based on each of the following conditions: the 1- and 3D material Courant limits; the VESSEL and total mass error limits; the outer-iteration count; the maximum allowable fractional change in gas volume fraction, temperature, and pressure; the diffusion number for heat transfer; and the maximum allowable fractional change in reactor-core power and adjustable-valve flow area. The new timestep size selected is the minimum imposed by the above conditions and the DTMAX maximum timestep size specified by the user in the timestep data. In subroutine NEWDLT, each conditional maximum timestep size is calculated, except for those based on the reactor-core power level and valve flow-area adjustment. The reactor-core power-change maximum timestep size is evaluated by subroutine RKIN, and the valve flow-area adjustment-change maximum timestep size is evaluated by subroutine VLVEX after evaluating subroutine NEWDLT. During the outer-iteration stage, subroutine HOUT applies the lesser of these two maximum timestep sizes to define DELT when it is less than the subroutine-NEWDLT defined timestep size.

In the event that a timestep solution is not completed successfully, TRAC-M will back up and try to reevaluate the modeled-system, new-time solution state. A backup occurs when the outer iteration does not converge (necessitating a reduction in the current timestep size) or when a flag indicating an extraordinary condition is activated. Either one will require the outer-iteration procedure to be reevaluated. It is important to understand that there are two types of backups, one corresponding to each scenario. When the outer iteration fails to converge during the OUTER overlay, the current timestep size is reduced and the calculation backs up to the start of the PREP stage after the control-parameter evaluation. This is necessary because any variable calculated during the prepass that is dependent on the timestep size was computed for the original timestep size and not for the newly-reduced timestep size. In addition, all new-time variables are reset to reflect their beginning-of-timestep values. This enables TRAC-M to begin again in the PREP stage in a manner no different than for any other timestep calculation except for having reduced the timestep size because of the backup. When the timestep requires one or more backups, the timestep size is halved for the first, second, and third backup; quartered for the fourth and fifth backup; and tenth for backups thereafter. This backup process continues until a small enough timestep size is reached to allow outer-iteration convergence to be satisfied or the timestep size needs to be reduced below the DTMIN minimum timestep size from the timestep data wherein TRAC-M stops the calculation.

The second type of backup is initiated by a flag being set signaling an extraordinary condition such as a water pack. This indicates that the outer iteration needs to be repeated to account for the extraordinary condition. TRAC-M resets any new-time variables that have been potentially evaluated incorrectly by the current attempt through subroutine OUTER with their old-time values, makes appropriate adjustments to prevent the extraordinary condition, and repeats the outer-iteration calculation. For this type of backup, the timestep size does not change, making it unnecessary to repeat the PREP-stage calculation.

The difference between the two types of backups is that for a backup to the start of the PREP stage, the timestep size is adjusted, all new-time variables are reset to their beginning-of-timestep values, and variables evaluated during the PREP stage are reevaluated using the newly adjusted timestep size. For a backup to the start of the outer iteration, no change occurs in the timestep size and only new-time variables calculated during the outer iteration are reset to reflect their beginning-of-timestep values.

3.6. Output Processing

The TRAC-M program normally produces five output files: TRCOUT, TRCMSG, TRCDMP, XTVGR.T, and XTVGR.B. TRAC-M also may produce a TRAC-format input-data file TRCINP and a labeled input-data file INLAB. The TRCDMP-, XTVGR.T-, and XTVGR.B-file real-valued variables have SI units. The TRCOUT- and TRCMSG-, TRACIN- and TRCINP-, and INLAB-file real-valued variables can have SI (0) or English (1) units based on the 0 (default value) or 1 value of Namelist variables IOOUT, IOINP, and IOLAB, respectively. SI- or English-units symbols can be output to the TRCOUT and TRCMSG files along with their real-valued variable values when Namelist variable IUNOUT = 1 (default value).

The TRCOUT file is in ASCII format and contains a user-oriented presentation of the calculation's input data and output results. During the input process, an echo of the input and restart data is output, and at selected times during the calculation, variable values of the current solution state of the modeled system are output. The TRCMSG file is in ASCII format and contains diagnostic messages concerning the progress of the calculation. The TRCDMP file is a binary file designed to provide solution-state data for problem restarts by TRAC-M. The XTVGR.T file is an ASCII-format file and the XTVGR.B file is a binary file, both of which provide data for XTV graphics. File TRCINP is output only when input-data file TRACIN is in FREE format, and file INLAB is output when Namelist variable INLAB = 3 is input as discussed in Sec. 3.2.

As the main driver routine of module EDIT, subroutine EDIT calls subroutine WCOMP to direct the output of a small and a large edit to the TRCOUT file at a specific time (timestep). The first edit written to the TRCOUT file occurs during the first timestep after the PREP stage, but all subsequent time edits are written after the POST stage. Subroutine WCOMP outputs general data first, then invokes lower-level routines to output the solution state of each component. The component-edit routines, which have names that begin with the letter W followed by the letters of the component-type name, output the solution-state variable data for that component to the TRCOUT file in an appropriate format for readability. For example, the PIPE-component-edit routine is called WPIPE, whereas the VESSEL-component-edit routine is called WVSSL. The 1D hydraulic-component-edit routines call subroutine ECOMP to output variable data that is common to 1D hydraulic components and then output any additional data special to that particular component.

The TRCDMP file is a structured binary file written with unformatted write statements. It contains sufficient data to restart the TRAC-M calculation at the problem time of a data-dump edit. This file is created by a sequence of calls to module DUMP. As the main driver routine of the DUMP module, subroutine DMPIT outputs the dump-header data and then calls the component data-dump subroutines. The names of the component data-dump subroutines begin with the letter D followed by the letters of the component-type name. For example, the PIPE component data-dump routine is called DPIPE, whereas the VESSEL component data-dump routine is called DVSSL. The 1D hydraulic-component data-dump routines call subroutine DCOMP to output to the TRCDMP file data common to 1D hydraulic components and then output any additional data special to that particular component using individual calls to subroutine BFOUT. The VESSEL-component data-dump routine DVSSL also calls subroutine BFOUT to output general VESSEL arrays and calls subroutine DLEVEL to output level arrays.

A time-edit data block is output at each dump edit time during a calculation. The number of time-edit blocks output to the TRCDMP file is determined by the dump-edit frequency specified by the timestep data. Each component has its own data block as a part of a time-edit data block. In subroutine DCOMP, the variable LCOMP, calculated for each 1D hydraulic component, is the total number of variable values output for the component to a time-edit block. The number of any additional variable values special to a particular component and output by the component data-dump routine is reflected in the variable LEXTRA. It is important to remember to increment either the variable LCOMP or LEXTRA accordingly when adding new component-variable values to the TRCDMP-file output.

The XTVGR.T and XTVGR.B files provide data for X-TRAC-VIEW (XTV), a phenomena visualization package. To assist in development and debugging, the graphics catalog is carried separately in the ASCII-format XTVGR.T file rather than at the beginning of the binary-format XTVGR.B file. Subroutines XTVINIT and XTVDR are called to create the XTV graphics catalog in file XTVGR.T. The XTV graphics catalog contains information for setting up the component and variable visualizations. This includes component name, type, connectivity and geometry, as well as a list of available variables with their types. File XTVGR.B is created by subroutine XTVDR and is called to generate a timestep-edit data block based on the graphics-edit frequency specified by the timestep data. It contains timestep-edit information as arrays of IEEE double-precision values, even in UNICOS. Each timestep-edit contains the problem time, followed by all the variables described in the graphics catalog, in the order listed. There is no compression or packing of variable values. This gives the XTV file cross platform compatibility.

XTV was designed as a phenomena visualization tool to replace the TRCGRF file and EXCON/TRAP that generated graphics for TRAC-P and earlier versions of TRAC. Presently most of the variables available in TRCGRF are, and eventually all of the variables available in TRCGRF will be, available in XTV. The timestep-data graphics-edit frequency GFINT determines the XTV graphics edit times. The maximum number of graphics edits depends on the modeled-system size and the XTVGR.B file internally defined file size (currently 500Mb). This internal limit can be overridden through the use of the optional XTVTIN input file, which contains the size in megabytes as an integer that the XTVGR.B data file is created. If the graphics data output to the XTVGR.B file reaches the XTVGR.B-file size limit, no further graphics edits will be output and an error message will be written to the TRCMSG file for each time edit that is not output. A complete description of the XTV file format is contained in Appendix C of the XTV Users Guide.

Subroutine INPUT opens the TRCINP file and calls subroutine PREINP to determine if the input-data TRACIN file is in FREE format or TRAC format. A FREE-format TRACIN file is read as ASCII data and parsed for numerical values to output the TRAC input data to the TRCINP file in TRAC format. Then either the TRACIN file (for TRACIN in TRAC format) or the TRCINP file in TRAC format (for TRACIN in FREE format) is read by the READI, READR, WARRAY, and WIARN subroutines to process the TRAC-M input data.

When Namelist variable INLAB = 3, the READI, READR, WARRAY, and WIARN subroutines output to file INLAB an input-data echo of the TRACIN-file data with variable-name label comments in FREE format. Outputting variable-name label comments between asterisks makes it a FREE-format file even though the input-data values are right-justified in 14-column fields. With a variable-name label above its scalar value or to the left of its array-element values, file INLAB provides input data whose parameter variables can be easily identified rather than require the input-data format description to define their parameter variables. This makes the input data infinitely more readable in a standard form so that all TRAC-M users can become familiar with it thereby reducing input-data defining errors. File INLAB is renamed TRACIN for subsequent use as the input-data file to TRAC-M. The file-INLAB option also is convenient for converting SI- or English-units input data in the TRACIN file to English- or SI-units input data. This is done with Namelist variables INLAB = 3, IOINP = 0 (SI) or 1 (English) for the TRACIN file , and IOLAB = 1 (English) or 0 (SI) for the INLAB file, respectively.

4. INPUT/OUTPUT IN SI OR ENGLISH UNITS

Real-valued variables in TRAC-M have SI (metric) units. Input to and output from TRAC-M can be in SI or English units as a user option. Namelist variables IOINP, IOLAB, and IOOUT define input-data file TRACIN, labeled input-data file INLAB, and output-data files TRCMSG and TRCOUT, respectively, to have real-valued variables with SI or English units when their values are 0 (default) for SI units or 1 for English units. The SI- or English-units symbols of real-valued variables are output with their values to the TRCMSG and TRCOUT files when Namelist variable IUNOUT is 1 (default). No units symbols are output when IUNOUT is 0. When programming the input and output routines of TRAC-M, code developers must consider the units of real-valued variables that are input and output and assure that they have SI units for internal use by TRAC-M.

The units of all real-valued variables involved in the input/output (I/O) of TRAC-M are defined by arrays LABELS(I), ITLS(I) = J, and LABUN(J) stored in COMMON block LABELV of include file LABELV.H. LABELS(I) is the left-justified CHARACTER*8 name of the Ith real-valued variable for I=1,...,777 (TRAC-M Version 1.10+) or for I=1,...,806 (Version 1.10+ when pending KAPL update changes are made). LABUN(J) is the left-justified CHARACTER*8 Jth units-name label beginning with the letters 'LU' for J=1,...,150. In Version 1.10+, 50 units-name labels are defined (see Table 6-2 in the TRAC-M Users Guide) and 100 are reserved for being defined by user input. LABUN(ITLS(I)) defines the units-name label of the Ith real-valued variable name LABELS(I). For example, the new-time liquid-temperature real-valued variable name LABELS(601) = `'TLN      '` has units-name label LABUN(3) = `'LUTEMP   '` based on ITLS(601) = 3. Appendix F, Secs. F.6 and F.7, provides a listing of COMMON block LABELV in the include file LABELV.H and a listing of subroutine BLOCK DATA BLKDAT2 in file BLKDAT2.F. The BLKDAT2 data initializes the variables stored in COMMON block LABELV. The variable values in the above example can be seen in this data.

Subroutine BLOCK DATA BLKDAT2 also initializes the values of arrays FACTOR(J), OFFSET(J), LABSV(L,K), LUNCB(L,J), LUPCB(L,J), and RUNCB(L,J) that are stored in COMMON block LABELV. FACTOR(J) and OFFSET(J) are the factor and shift values for converting a SI-units variable value with units-name label LABUN(J) to English units by multiplying the SI-units value by the factor FACTOR(J) and then adding the shift OFFSET(J) to obtain the English-units value. LABSV(L,K) is a left-justified CHARACTER*14 name label with units symbol in parentheses for the Kth signal-variable parameter in SI units (L=1) or English units (L=2). LUNCB(L,J), LUPCB(L,J), and RUNCB(L,J) are left-, (in parentheses) left-, and right-justified CHARACTER*13, *13, and *12 labels, respectively, for the units-name label LABUN(J) units symbol in SI (L=1) or English (L=2) units. LUPCB(L,J) has units symbols in parentheses, whereas LUNCB(L,J) and RUNCB(L,J) do not.

Subroutine UNCNVT with arguments (LABEL, VAR, LV, IV, IU) is called to determine the units and possibly convert the units of a real-valued variable's value/s from input or for output. The real-valued variable is VAR with LV values and stride IV. For a real-valued scalar variable VAR, LV = 1 and IV = 1. For a real-valued array variable VAR with values VAR(1), VAR(3), VAR(5), . . . , VAR(LV), LV is an odd value and IV = 2. LABEL is the CHARACTER*(*) variable name of real-valued variable VAR. In the above example, `'TLN'`, `'TLN      '`, `'   TLN   '`, or `'      TLN'` are all valid definitions for LABEL. IU = IUIN – IUOUT defines the type of units conversion: 1 converts an English-units value/s to SI units, 0 does no units conversion, and –1 converts a SI-units value/s to English units. IUIN defines the units of VAR input to UNCNVT (0 = SI, 1 = English) and IUOUT defines the units of VAR output from UNCNVT (0 = SI, 1 = English). For example, if VAR has English units that were input to TRAC-M and UNCNVT is to convert VAR to have SI units, IU = 1 – 0 = 1. If the parameter value/s of VAR are internal to TRAC-M and used in the calculation in SI units, VAR should be a temporary variable so that any possible units conversion will not effect the internal value/s of the parameter variable. Even converting the value of VAR from SI to English units for output and then back from English to SI units for internal use will change the internal value/s in TRAC-M because of numerical roundoff. This will affect the calculation. The frequency of output edits in English units must not affect the calculative results in this way.

In subroutine UNCNVT, an SI-units value is converted to English units by VAR(L) = VAR(L)*FACTOR(J)+OFFSET(J), whereas an English-units value is converted to SI units by VAR(L) = (VAR(L)–OFFSET(J))/FACTOR(J) for L=1,IL,IV. The units-name index J = ITLS(I) is determine in UNCNVT by left-justifying the characters of the LABEL variable name and finding the Ith element of LABELS(I) that matches it. The value of I is saved in variable IOLD that is stored in COMMON block LABELV so that its value can be used outside of subroutine UNCNVT as well.

An example of programming the output of three new arrays: DNEWN, HYNEW, and HTNEW in subroutine ECOMP is presented in the example update of Appendix E. The two DO loops over N are needed because the temporary array TMP(10,24) for values of the new arrays handles only 10 cell or interface values at a time for units conversion and output. The calls to subroutine UNCNVT define IU = –IOOUT because IUIN = 0 for the SI-units internal new arrays that are output and IUOUT = IOOUT. SI- or English-units symbols are output in the tabular-data heading labels with LUNCB(IOOUT+1, ITLS(IOLD)).

When programming new real-valued variables in TRAC-M that are involved in I/O, code developers need to update their units information in arrays LABELS(I), ITLS(I), and LABUN(J), as well as their related arrays. The efficient search algorithm in subroutine UNCNVT for finding a match between LABEL and LABELS(I) requires that the LABELS(I) variable names be ordered alphabetically and be left justified. A lack of success in getting code developers to enter new variable names alphabetically in COMMON blocks (to make variable names easier to find visually) prompted the writing of FORTRAN 77 program LABPRG to do this updating automatically in COMMON block LABELV and subroutine BLOCK DATA BLKDAT2. Getting the Jth units-name value defined in the corresponding location of ITLS(I) would have been even more of a challenge to code developers. Appendix F describes how to prepare the input data of file LABNEW for program LABPRG to add or delete units-name labels in LABUN(J) (which also effects FACTOR(J), OFFSET(J), LUNCB(L,J), LUPCB(L,J), and RUNCB(L,J)); signal-variable name labels in LABSV(L,K); and real-valued variable names in LABELS(I) (which also effects ITLS(I)). The procedure is straight forward and allows LABNEW input data from one or more situations of programming changes made to TRAC-M to be processed together by program LABPRG. The output result is the include file LABELV.H that defines COMMON block LABELV and file BLKDAT2.F that defines subroutine BLOCK DATA BLKDAT2. After executing program LABPRG, these two output files need to be committed into the CVS source-file repository as described on page E-4 of App. E.

5. MEMORY MANAGEMENT

To understand the data storage in TRAC-M, it is necessary to consider the memory-management requirements for a large code. First, any program that uses a large amount of memory must allocate that memory flexibly and dynamically during execution. Static dimensioning, i.e., dimensioning at compile time to accommodate the largest possible problem, is at best wasteful of memory and at worst infeasible. The alternative strategy of preprocessing the input to determine array sizes before compilation would be extremely cumbersome for a code as complex as TRAC-M. Static-memory allocation schemes of all types also have the disadvantage that there is no possibility of increasing or decreasing memory requirements during a calculation when the evaluation path changes or when temporary arrays are no longer required.

Second, because standard FORTRAN does not support dynamic-memory allocation, it is necessary to accomplish dynamic-memory allocation by using variable offsets into a single container array. Obviously, any implementation based on this concept will have some degree of awkwardness. On some operating systems, the size of the container array can be changed dynamically. On others, it must be fixed in advance. Although the latter implementation is not, technically speaking, dynamic, it is flexible, and fixing the size of the container array makes a trivial difference in the coding. The bulk of the memory-management implementation in TRAC-M arises in the computation and management of the offset or pointer variables.

As an example of using a container array for dynamic-memory management, consider the container array, A(*), where the actual dimensioned size of the A array is sufficient for the problem at hand. Now assume that we wish to store two arrays, X(20) and Y(20), starting with the 14th element of the container array. There are a number of ways of doing this. One option is to define offset pointers as in this example:

```
IFREE   = 14
NCELLS  = 20
LX      = IFREE
LY      = LX + NCELLS
IFREE   = LY + NCELLS
```

These pointer variables are defined in a manner that establishes mesh-wise storage. In this example, the arrays X and Y occupy locations A(14) through A(33) and A(34) through A(53), respectively. With the use of these pointer variables, X(N) can be referenced as A(LX+N–1) and Y(N) as A(LY+N–1). The referencing can be made more readable by passing A(LX) and A(LY) as actual arguments to a subroutine that uses X and Y as the names for the corresponding local arrays.

Two drawbacks of the pointer methodology are the large amount of coding needed to define the pointer variables and the need to use subroutine arguments for readability. Another drawback arises when using pointer variables in the context of multidimensioned array variables, i.e., the dimensions must be treated as variable. This complicates the coding and makes dynamic debugging more difficult.

Another option for storing in a container array is to use EQUIVALENCE statements. This has the advantage that the variables can appear in COMMON. In our previous example, we could achieve the same data storage and data structure by

```
PARAMETER    (LX = 14, LY = 34)
EQUIVALENCE (A(LX), X(1)), (A(LY), Y(1))   .
```

Equivalencing that creates mesh-wise storage, as in this example, cannot be used for dynamic-memory allocation because knowledge of the array sizes, as well as their actual memory locations, is built into the EQUIVALENCE statement. The answer to using equivalencing for dynamic-memory allocation is to equivalence the arrays according to the cell-wise storage scheme, i.e.,

```
EQUIVALENCE (A(1), X(1)), (A(2), Y(1))   .
```

The establishment of a cell-wise storage scheme using EQUIVALENCE statements is useful for dynamic-memory allocation because the EQUIVALENCE statements can be treated as determining the relative order of the variables, rather than their actual locations in memory. The location in memory, or offset into the container array, is then defined dynamically in terms of loop limits. Using loop limits NB = 14 and NE = 52 with a stride of NV = 2 in referencing arrays X and Y in the last example would establish a mesh-wise storage occupying the same memory locations in the A array as in the two previous examples but with X and Y elements interspersed.

One drawback to a cell-wise scheme is the necessity for including the stride in the coding. Another drawback that can arise on certain hardware is inefficiency in referencing vectors with non-unit stride. Finally, this methodology can be cumbersome when combined with the use of temporary arrays that have mesh-wise storage. Nonetheless, our experience with this methodology in TRAC-P has been positive in terms of eliminating coding errors resulting from maintenance of pointers and long subroutine argument lists in earlier versions of TRAC.

TRAC-M eliminates the use of EQUIVALENCE statements for the VESSEL-component 2- and 3D parameter arrays by storing these arrays in labeled COMMON blocks vssWhat and vssArCOM with PARAMETER-constant dimensions NI, NJ, and NK rather than storing them in the container A array. The use of these EQUIVALENCE statements is not standard FORTRAN 77 coding. They also prevented optimization by some computing-platform compilers. Doing this eliminated the NV=291 stride of cell-wise storage that made programming and debugging more complicated. Now the NV=1 stride of mesh-wise storage makes the (I,J,K) subscript the subscript of the 3D cell. For multi-VESSEL problems, this is complicated by the I-dimension range having a subrange for each VESSEL, with the sum of each VESSEL's number of I-direction cells needing to be no more than the NI dimension. On the other hand, the NJ and NK dimensions are maximum dimensions for each of the VESSELs.

6. TRAC-M FOR VARIOUS COMPUTER SYSTEMS

The source coding of TRAC-M Version 1.10+ is the coding of TRAC-P Version 5.4.29 reprogrammed in standard FORTRAN 77 so that it can be compiled on different computing platforms without change. TRAC-P required UPDATE/HISTORIAN conditional directives (*DEFINEs) in its program library to configure the TRAC-P source coding generated by HISTORIAN for a specific computing platform. This complicated the programming of TRAC-P. By reprogramming TRAC-M in standard FORTRAN 77, with low level I/O routines programmed in the C language, source-coding changes are no longer needed for different computing-platform compilers. Now the TRAC-M source coding is stored under the revision control system (RCS) supervised by the concurrent version system (CVS). This provides version control for making programming changes to TRAC-M. See App. E for a description and example of making programming changes to TRAC-M under CVS.

Reprogramming of TRAC-P for TRAC-M involved passing the source coding through the FOR_STRUCT restructuring tool with the following options selected: set to unify indentions (for IF and DO statements), capitalize all FORTRAN keywords (IF, THEN, ELSE, ENDIF, DO, ENDO, CALL, WRITE, READ, FORMAT, etc.), and untangle programmed loops. Statement identifiers in columns 82–94 of the first line of each statement were added by a Perl script. The continuation of a statement on the next line was reprogrammed with a trailing & in column 74 [for use in Fortran 90 (F90) free-format mode] and an & in column 6 of the continued line. Perl scripts were programmed to do additional FORTRAN 77 coding changes automatically. Much of this involved eliminating EQUIVALENCE statements involving the container A array, moving or changing to REAL the CHARACTER and INTEGER data in the container A array, eliminating GOTO statements wherever possible, and reprogramming the control-logic bit numbers to the first 32 bits of the BIT and BITN array variables. The newly defined bit numbers and their definitions are described in App. G.

The details of converting TRAC-P Version 5.4.25 to TRAC-M Version 1.10 are described in the report "Transitioning between TRAC-P Version 5.4.25 and TRAC-M Version 1.10." The TRAC-P update changes from Version 5.4.25 to 5.4.29 and coding corrections found during later F-90 modernization work on TRAC-M were added to TRAC-M Version 1.10 under CVS control. This created TRAC-M Version 1.10+. Future changes to TRAC-M need to follow the FORTRAN 77 (and eventually F90) programming standard that has been implemented.

APPENDIX A

TRAC-M SUBPROGRAMS

A.1. C-Language Routines

Name	Function
BTESTC	Obtains (accesses) a bit's 0 or 1 values for a cell or interface in the BIT or BITN array.
CEPSILON	Returns precision of REAL*8 arithmetic.
CGCLOSE	Not used.
CGOPENA	Not used.
CGOPENR	Not used.
CGOPENW	Not used.
CGREAD	Not used.
CGWRITE	Not used.
CRSTIME	Returns the date and time as an ASCII string.
CUSRTIME	Returns the cpu time in seconds.
CXTVBW	Writes binary data to the XTV datafile.
CXTVCL	Closes the XTV datafile.
CXTVIN	Sets the maximum XTV datafile size from optional file XTVTIN.
CXTVOA	Opens the XTV datafile for appending if less than the maximum size.
CXTVOW	Creates a new XTV datafile.
EXIT_PROCESS	Error-handling routine for C-language routines.
IBCLRC	Clears a bit's value to 0 for a cell or interface in the BIT or BITN array.

IBSETC	Sets a bit's value to 1 for a cell or interface in the BIT or BITN array.
LOC4	Calculates the differential offset from the beginning of a common block with 4-byte values; e.g., LOC4(LQP3RF) – LOC4(LALP).
LOCF	Calculates the differential offset from the beginning of the container A array or a common block with 8-byte values; e.g., LOCF(ALPN) – LOCF(A11111).
OF1123C	Clears the values of bits 11, 12, and 13 to 0 for all the cells in the BITN array.
ON1123C	Clears the values of all bits except bits 2, 11, 12, 13, 30, and 32 for all cells and interfaces in the BITN array.
RS_TIME	Returns the current time as an ASCII string; e.g., Fri Nov 21 17:45:34: MST 1997.

A.2. FORTRAN-Language Routines

Name	**Function**
AICOMP	Determines the A-array index for a 1D-component parameter.
ALLBLK	Tests for all blanks in specified substring of string.
ASIGN	Assigns the component pointers according to the internal order (IORDER) array.
ASTPLN	Evaluates mass and energy fluxes at the PLENUM junctions during postpass.
AUXPLN	Evaluates mass and energy fluxes at the PLENUM junctions during the outer iteration.
BACIT	Initiates backward substitution after direct vessel matrix inversion.
BAKUP	Overwrites end-of-timestep variables with start-of-timestep values for one vessel level.

BALANCT	Support subroutine for SGEEV that balances a real matrix and isolates eigenvalues whenever possible.
BALBAKT	Support subroutine for SGEEV that forms the eigenvectors of a real matrix.
BANSOL	Solves linear matrix equation.
BDPLEN	Fills the PLENUM boundary array.
BFALOC	Allocates files and buffers for buffered I/O.
BFCLOS	Empties buffers and closes file.
BFIN	Initiates binary input subroutine.
BFOUT	Initiates binary output subroutine.
BKMOM	Initiates backward substitution for stabilizing momentum equations.
BKSMOM	Performs backward substitution for stabilizing momentum equations.
BKSPLN	Initiates backward substitution for stabilizing mass and energy equations for the plenum component.
BKSSTB	Initiates backward substitution for stabilizing mass and energy equations.
BKSTB3	Initiates backward substitution for stabilizing mass and energy equations for the VESSEL component.
BLKDAT	Initializes common variables in a block data statement.
BLKDAT2	Initializes SI/English-conversion common variables in a block data statement.
BREAK1	Controls BREAK prepass.
BREAK2	Controls BREAK outer iteration.
BREAK3	Controls BREAK postpass.
BREAKX	Evaluates BREAK pressure, temperature, and void fraction.

C2R	Converts CHARACTER*8 data to REAL*8 data for the container A array.
CBEDIT	Edits the first 10 control-block parameter values along with their variable-name labels and a control-block schematic.
CBSET	Evaluates control-block function output parameters.
CDTHEX	Evaluates the diametral thermal expansion of Zircaloy as a function of temperature.
CELLA3	Evaluates cell-averaged quantities that are required for the interphasic heat-transfer calculation for the VESSEL component.
CELLAV	Evaluates cell-averaged quantities that are required for the interphasic heat-transfer calculation for 1D components.
CHBD	Checks boundary data.
CHBSAV	Transfers selected BD-array data into the A array required for the accumulator phase-separation model.
CHBSET	Stores data in the BD array temporarily to check for consistency in the junction data.
CHECKSIZE	Checks for adequate dimensioned space in an array.
CHEN	Uses Chen correlation to evaluate the forced convection nucleate boiling heat-transfer coefficient.
CHF	Evaluates CHF based on a local-conditions formulation.
CHF1	Applies Biasi CHF correlation.
CHKBD	Checks for the consistency in the boundary-array data during initialization.
CHKSR	Checks VESSEL component source locations.
CHOKE	Evaluates the critical-flow phasic velocities and their derivatives with respect to the donor-cell total pressure.
CIF3	Evaluates interfacial shear for VESSEL component.
CIHTST	Sets up arrays for heat-structure component.

CIRAD	Completes initialization of enclosures for the radiation model.
CIRADH	Gets the A-array index of the hydrodynamic-cell data needed by an enclosure of the radiation model.
CIRADR	Gets the A-array index of the heat-structure node-row data needed by an enclosure of the radiation model.
CIVSSL	Transfers vessel data from large-core memory (LCM) to small-core memory (SCM) so that the remaining data can be initialized.
CLEAN	Closes TRAC output files.
CLEAR	Sets the elements of a real array to a constant value.
CLEARI	Sets the elements of an integer array to a constant value.
CLRBRVLT	Initializes all values of the BREAK-component specific-component-table specTableCom common block to 0 or 0.0e0.
CLRFIVLT	Initializes all values of the FILL-component specific-component-table specTableCom common block to 0 or 0.0e0.
CLRPIVLT	Initializes all values of the PIPE-component specific-component-table specTableCom common block to 0 or 0.0e0.
CLRPLVLT	Initializes all values of the PLENUM-component specific-component-table specTableCom common block to 0 or 0.0e0.
CLRPRVLT	Initializes all values of the PRIZER-component specific-component-table specTableCom common block to 0 or 0.0e0.
CLRPUVLT	Initializes all values of the PUMP-component specific-component-table specTableCom common block to 0 or 0.0e0.
CLRRDVLT	Initializes all values of the heat-structure ROD- or SLAB-component specific-component-table specTableCom common block to 0 or 0.0e0.
CLRTEVLT	Initializes all values of the TEE- or SEPD-component specific-component-table specTableCom common block to 0 or 0.0e0.

CLRVAVLT	Initializes all values of the VALVE-component specific-component-table specTableCom common block to 0 or 0.0e0.
CLRVSVLT	Initializes all values of the VESSEL-component specific-component-table specTableCom common block to 0 or 0.0e0.
COMPI	Performs various A-array loading tasks common to most 1D components.
CONBLK	Computes all 61 types of control-block outputs that do not require tabular storage or PI/PID controllers.
CONCF	Returns maximum solubility (kg solute/kg liquid, lb_m solute/lb_m liquid) for species ISPEC at pressure P and liquid temperature TL.
CONSTB	Drives subroutine STBME.
COPYA	Copies value of variable SRCVAL into variable SNKVAL.
CORE1	Evaluates rod heat-transfer coefficients and tracks quench fronts.
CORE3	Evaluates rod temperature distributions.
COURNO	Defines the maximum material Courant number for the VESSEL component.
CPLL	Determines the specific heat of D2O or H2O liquid as a function of enthalpy and pressure by calling CPLLD or CPLLH.
CPLLD	Evaluates the specific heat of D2O liquid as a function of enthalpy and pressure.
CPLLH	Evaluates the specific heat of H2O liquid as a function of enthalpy and pressure.
CPVV1	Determines the specific heat of D2O or H2O vapor as a function of temperature and pressure by calling CPVV1D or CPVV1H.
CPVV1D	Evaluates the specific heat of D2O vapor as a function of temperature and pressure.

CPVV1H	Evaluates the specific heat of H2O vapor as a function of temperature and pressure.
CVMGT	A logical test function.
CWVSSL	Transfers VESSEL data from LCM to SCM so that they can be printed.
CYLHT	Evaluates temperature fields in a cylinder.
DATER	Date routine.
DAXPY	Function that evaluates a constant times a vector plus a vector.
DBRK	Generates BREAK data dump.
DCODF	Evaluates a numeric code based on data types.
DCOMP	Dumps 1D component data.
DDOT	Evaluates the dot product of two vectors.
DECAYS	Initializes the decay-heat constants to be consistent with the ANS5.1 1979 standard.
DELAY	Provides a time-delay function for the input variable (XIN). The output (XOUT) is played back with the value that the input had TAU seconds previously. Linear interpolation is used for playback when (TIMET minus TAU) falls between two stored time values. The user specifies the number of table storage pairs (NINT) to be saved. Both the time and the value of the input are stored in the table array as pairs of points.
DELTAR	Evaluates transient fuel-cladding gap spacing (only if NFCI = 1).
DFILL	Generates FILL data dump.
DGBFA	Factors a double precision band matrix by elimination.
DGBSL	Solves double precision band system A * X = B or TRANS(A) * X = B using factors computed by subroutine DGBFA.

DHTSTR	Determines the size of the data dump and writes the restart input data for a heat-structure component to the dump file.
DLEVEL	Generates VESSEL level data dump.
DMPBRVLT	Stores the BREAK-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPFIVLT	Stores the FILL-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPFLT	Stores the generic-component-table common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPIT	Main module for generating a dump-restart data file.
DMPPIVLT	Stores the PIPE-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPPLVLT	Stores the PLENUM-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPPRVLT	Stores the PRIZER-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPPUVLT	Stores the PUMP-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPRDVLT	Stores the heat-structure ROD- or SLAB-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPTBVLT	Stores the TURB-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.

DMPTEVLT	Stores the TEE- or SEPD-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPVAVLT	Stores the VALVE-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DMPVLT	Calls the component-specific subroutine DMPxxVLT for outputting specTableCom common-block data to the TRCDMP file.
DMPVSVLT	Stores the VESSEL-component specific-component-table specTableCom common-block data in an array that is written to the TRCDMP file by subroutine BFOUT.
DPIPE	Generates PIPE data dump.
DPLEN	Generates PLENUM data dump.
DPUMP	Generates PUMP data dump.
DRAD	Generates the radiation-model data dump.
DROD1	Writes the restart input data arrays for a subset of the heat-structure component data to the TRCDMP file.
DSCAL	Scales a vector by a constant factor.
DTDIAG	Outputs timestep diagnostic information.
DTEE	Generates TEE data dump.
DTURB	Generates TURB (turbine) data dump.
DVLVE	Generates VALVE data dump.
DVPSCL	Initializes scale factors on derivative of velocities with respect to pressure for a VESSEL level.
DVSSL	Generates VESSEL-component data dump.
ECOMP	Writes hydrodynamic and heat-transfer information for 1D components to output file.
EDIT	Writes a large edit to the TRCOUT file.

ELGR	Converts cell elevations to the slope between cells and converts K-factors to additive friction-loss coefficients.
ENDDMP	Empties dump buffers and closes dump file.
ERROR	Processes different kinds of error conditions.
ETEE	Evaluates TEE parameters on explicit pass.
EVALDF	Evaluates the absolute difference between XOLD and XNEW.
EVFXXX	Evaluates the XXX component-action function.
EVLTAB	Interpolates the function value F from the tabular data based on the value of the table's independent variable: a signal variable (NVAR.GT.0), a control block (NVAR. LT.0), or a trip-signal difference DELSV (NVAR.EQ.0).
EXPAND	Adds rows of conduction nodes within the vessel rods during reflood.
FAXPOS	Evaluates the flow-area fraction, FA, or valve-stem fractional position, XPOS, for the VALVE.
FBRCSS	Identifies break components that are coupled through a fluid-flow path to the secondary side of a steam generator.
FCEINF1	Finds the radiation enclosure number, face number, hydro-level number, and radiation-level number associated with a given M number.
FCEINFO	Finds an array of radiation enclosure numbers, face numbers, hydro-level numbers, and radiation-level numbers associated with a given array of M numbers.
FEMOM	Sets up stabilizing momentum equations.
FEMOMX	Performs forward elimination on radial motion equation.
FEMOMY	Performs forward elimination on azimuthal motion equation.
FEMOMZ	Performs forward elimination on axial motion equation.
FF3D	Makes final pass update for all variables in 3D VESSEL.

FILL1	Controls FILL prepass.
FILL2	Controls FILL outer iteration.
FILL3	Controls FILL postpass.
FILLX	Evaluates postpass FILL velocity.
FIND	Obtains the A-array index for a hydrodynamic-component parameter.
FINDER	Locates array data for a given component.
FINDH	Obtains the A-array index for a heat-structure parameter.
FINDNUM	Obtains the A-array index for fixed-length table data of a component.
FLTOM	Controls transfer of data between hydro and heat-structure databases.
FLUX	Evaluates mass flow at the boundary of a 1D component for use in mass inventory.
FLUXES	Defines explicit portion of mass and energy flux terms.
FNMESH	Initializes the supplemental user-specified rows of conduction nodes within the vessel rods at the start of reflood.
FPROP	Determines the D2O or H2O fluid enthalpy, transport properties, and surface tension by calling FPROPD or FPROPH.
FPROPD	Evaluates the D2O fluid enthalpy, transport properties, and surface tension.
FPROPH	Evaluates the H2O fluid enthalpy, transport properties, and surface tension.
FROD	Evaluates temperature profiles in nuclear or electrically heated fuel rods.
FTHEX	Evaluates the fuel linear thermal-expansion coefficient for uranium dioxide and mixed-oxide fuels.

FWALL	Computes a two-phase friction factor.
FWKF	Evaluates form-loss K-factors for an abrupt contraction or expansion.
GAPHT	Evaluates fuel-cladding gap heat-transfer coefficient.
GETBIT	Returns value of bit N of word B.
GETCENC	Gets TRAC's internal radiation enclosure number corresponding to an input radiation enclosure number.
GETCRV	Gets appropriate pump curves from database.
GETGEN	Returns a selected variable value from the generic-component-table genTableCom common block.
GETPUMP	Returns a selected variable value from a PUMP-component specific-component-table specTableCom common block.
GETRADM	Gets running index for an enclosure radiation level face.
GETRDM1	Gets an array of running indices for an array of enclosure radiation level faces.
GETROD	Returns a selected variable value from a heat-structure ROD- or SLAB-component specific-component-table specTableCom common block.
GETTEE	Returns a selected variable value from a TEE- or SEPD-component specific-component-table specTableCom common block.
GETTURB	Returns a selected variable value from a TURB-component specific-component-table specTableCom common block.
GETTYPE	Defines the CHARACTER*8 name of a component type given the REAL*8 internal-code value for the component type.
GETVALVE	Returns a selected variable value from a VALVE-component specific-component-table specTableCom common block.
GETVSAR	Returns a selected variable value from a VESSEL-component specific-component-table specTableCom common block when

its calling subroutine doesn't know the structure of the VESSEL-component database.

GETVSS	Returns a selected variable value from a VESSEL-component specific-component-table specTableCom common block.
GVSSL1	Evaluates integrated vessel parameters for graphics purposes.
GVSSL2	Evaluates average values for vessel graphics (integrated values calculated in subroutine GVSSL1).
HASH	Determines the first array index for each alphabet letter that is the first letter of the character-string label names.
HEV	Determines the heat of evaporation of D2O or H2O liquid corresponding to a given temperature at low pressure.
HEVD	Evaluates the heat of evaporation of D2O liquid corresponding to a given temperature at low pressure.
HEVH	Evaluates the heat of evaporation of H2O liquid corresponding to a given temperature at low pressure.
HLFILM	Evaluates wall-to-liquid heat-transfer coefficient in transition and film boiling.
HLFLMR	Evaluates wall-to-liquid heat-transfer coefficient in reflood transition and film boiling.
HOUT	Controls the outer-iteration logic for a complete timestep.
HQR2T	Support subroutine for SGEEV that finds the eigenvalues of a real upper Hessenberg matrix by the QR method.
HQRT	Support subroutine for SGEEV that finds the eigenvalues and eigenvectors of a real upper Hessenberg matrix by the QR method.
HTCOR	Computes heat-transfer coefficients.
HTIF	Evaluates the interphasic heat-transfer for the zero- and 1D components.
HTPIPE	Averages velocities and generates heat-transfer coefficients for 1D components.

HTSTR1	Controls heat-structure prepass.
HTSTR3	Controls heat-structure postpass.
HTSTRP	Evaluates the heat-structure instantaneous power and total energy in each ROD or SLAB element of the heat structure.
HTSTRV	Initializes to zero some VESSEL-component hydro-cell arrays used to store heat-structure information.
HTVSSL	Averages velocities and generates heat-transfer coefficients for the vessel.
HUNTS	Searches character string for specified search string.
HVFILM	Evaluates the vapor heat-transfer coefficient that is the maximum of the Bromley, natural-convection, and the Dougall-Rohsenow coefficients.
HVNB	Evaluates vapor heat-transfer coefficient for nucleate boiling.
HVWEBB	Evaluates vapor heat-transfer coefficient for dispersed vapor flow.
I42R	Copies the values of an INTEGER*4 array into a REAL*8 array.
IBRK	Initializes the BREAK data arrays that are not input.
ICOMP	Controls the routines that initialize component data.
IDAMAX	Finds the index of the vector element that has the maximum absolute value.
IDEL	Searches specified substring of string for any one character in a set of specified characters.
IFILL	Initializes the FILL data arrays that are not input from cards.
IFSET	Initializes 3D interfacial shear at the start of each VESSEL prepass.
IHPSS1	Evaluates hydraulic-path steady-state initialization for the 1D hydraulic components.

IHPSS3	Evaluates hydraulic-path steady-state initialization for the 3D VESSEL component.
INDEL	Searches specified substring of string for first nonoccurrence of any one character in a set of specified characters.
INIT	Entry routine for subroutine INIT.
INITBC	Initializes VESSEL component phantom cells and sets some boundary conditions.
INNER	Performs an inner iteration for a 1D component.
INPUT	Entry routine for subroutine INPUT.
IPIPE	Initializes the PIPE data arrays that are not input.
IPLEN	Loads the PLENUM arrays that are needed, but not input, to start a problem.
IPRIZR	Initializes the PRIZER (pressurizer) data arrays that are not input.
IPROP	Calls subroutines THERMO, FPROP, and MIXPRP for most 1D components.
IPUMP	Initializes the PUMP data arrays that are not input.
IROD	Initializes rod component parameters that are not user-input.
IRODL	Initializes heat-structure arrays that provide information on the location of hydro data.
ISORT	Sorts a list of integers in ascending order.
ITEE	Initializes TEE data arrays that are not input from cards.
ITURB	Loads the arrays that are not input but that are needed to start a problem.
IVLVE	Initializes the VALVE data arrays that are not input.
IVSSL	Initializes the VESSEL data arrays that are not input.
IWALL3	Divides input friction factor by hydraulic diameter.

J1D	Fills boundary array at component junctions.
J3D	Fills boundary array at VESSEL source-connection junctions.
JBD4	Fills boundary array with JCELL parameters for the TEE-component internal junction.
JFIND	Locates junctions in junction sequence array.
JUNSOL	Determines junction parameters for connecting and sequencing components.
JUSTLR	Left or right justifies the letters of a character string.
JVALUE	Converts one character of a string to a binary number: 0-9 returned as binary mode; blank, as binary 0; all others, as <0.
LABELD	Outputs the D2O properties comment.
LABELH	Outputs the H2O properties comment.
LABELP	Outputs the D2O or H2O properties comment be calling LABELD or LABELH.
LCHPIP	Defines the pointer to the hydro array data for a 1D component.
LCHVSS	Defines the pointer to the hydro array data for a VESSEL component.
LCMTRN	Transfers data to LCM.
LEVEL	Uses a curve fit to obtain the water level in a cylindrical pipe as a function of the void fraction.
LEVELI	Transfers data for axial level IZ from inverted form to stacked form.
LEVELR	Transfers data for axial level IZ from stacked form to inverted form.
LININT	Performs linear interpolation on array tabular data.
LININT0	Performs linear interpolation on array tabular data without a derivative evaluation.

LINT4D	Linearly interpolates a function table with zero to four independent variables.
LOADN	Reads in an array of input real or integer values in the load format and stores their values in a REAL*8 array.
LOCTRB	Evaluates the required relative variable location in a common block for the TURB.
LTOPP	Determines if velocities at opposite faces of a TEE-component JCELL are both directed into the JCELL.
MANAGE	Performs all level and rod-data management operations for the VESSEL and heat-structure components.
MATSOL	Solves the vessel-matrix equation A * X = C using the capacitance method.
MBN	Evaluates values for electrically heated nuclear fuel-rod insulator properties.
MFROD	Orders fuel-rod property selection and evaluates an average temperature for property evaluation.
MFUEL	Evaluates uranium-dioxide and uranium-plutonium dioxide properties.
MGAP	Evaluates values for the thermal conductivity of the gap-gas mixture.
MHTR	Evaluates values for electrically heated fuel-rod heater coil properties.
MIX3D	Initializes stabilizer quantities at start of problem and equivalences stabilizer quantities to basic values when two-step method is not being used.
MIXPRP	Evaluates mixture properties from those of separate phases.
MOVINFO	Reorders radiation enclosure information.
MPROP	Orders structure property selection and evaluates an average temperature for property evaluation.
MSTRCT	Evaluates properties for certain types of steel.

MWRX	Evaluates the Zircaloy steam reaction in the cladding at high temperatures.
MZIRC	Evaluates properties for Zircaloy-4.
NAMLST	Performs input-data check on all namelist variables.
NEWDLT	Evaluates prospective new-time increment.
NXTCMP	Finds the beginning of data for the next component.
OFFTKE	Evaluates exit void fraction for TEE component offtake model.
ORDER	Rearranges the signal-variable, control-block, and trip ID numbers in ascending order based on their absolute value and searches for the do-loop index values for each control-parameter evaluation pass through the signal variables, control blocks, and trips.
ORTHEST	Support subroutine for SGEEV that does a orthogonal similarity transformation of a real matrix.
ORTRANT	Support subroutine for SGEEV that accumulates the orthogonal similarity transformation used in the reduction of a real matrix.
OUT1D	Controls outer calculation for 1D components.
OUT3D	Controls outer calculation for a VESSEL.
OUTER	Controls outer calculation for one timestep.
PIPE1	Controls PIPE prepass.
PIPE1X	Evaluates liquid volume discharged (q_{out}), collapsed liquid level (z), and volumetric flow rate (v_{flow}); assumes vertical component with low-numbered cell at top.
PIPE2	Controls PIPE outer iteration.
PIPE3	Controls PIPE postpass.
PIPROD	Moves hydro data for a 1D component to and from the heat-structure database.

PLEN1	Performs the prep stage calculation for the PLENUM timestep initialization.
PLEN2	Controls PLENUM outer iteration.
PLEN3	Controls PLENUM postpass.
PNTROD	Initializes HTSTR pointers.
PNTVSS	Initializes VESSEL pointers.
POST	Controls postpass calculation for one timestep.
POST3D	Controls postpass calculation for the VESSEL.
POSTER	Performs postpass calculation for 1D components.
POWINT	Evaluates the integral power (energy) into the pipe wall.
PREFWD	Prepares for evaluation of the 3D wall shear coefficients.
PREINP	Converts free-format TRACIN deck to format used by TRAC input subroutine.
PREP	Controls prepass calculation for one timestep.
PREP1D	Controls the prepass calculation for 1D components.
PREP3D	Controls prepass calculation for 3D components.
PREPER	Performs prepass calculation for 1D components.
PRIZR1	Controls PRIZER (pressurizer) prepass.
PRIZR2	Controls PRIZER (pressurizer) outer iteration.
PRIZR3	Controls PRIZER (pressurizer) postpass.
PRZR1X	Evaluates pressurizer mass change during steady-state calculation.
PSTEPQ	Controls printing, dumping, and graphing of data at the completion of a timestep.
PTRSPL	Initializes PLENUM pointers for use by signal variables and graphics.

PUMP1	Controls PUMP prepass.
PUMP2	Controls PUMP outer iteration.
PUMP3	Controls PUMP postpass.
PUMPD	Evaluates head and torque from PUMP curves.
PUMPI	Supplies built-in PUMP characteristics.
PUMPSR	Evaluates PUMP momentum and energy source.
PUMPX	Evaluates PUMP head and torque.
PUTRADM	Stores the running index for an enclosure radiation-level face.
PUTRDM1	Stores the M number for a radiation-level face for a given enclosure number.
R2C	Converts REAL*8 data in the container A array to CHARACTER*8 data.
R2C32	Converts four REAL*8 array elements defining the component title to one CHARACTER*32 variable.
R2I4	Copies the values of a REAL*8 array into an INTEGER*4 array.
R2II	Copies integer values stored in a REAL*8 array into a generic INTEGER array.
RADCHTS	Combines the radiative surface heat fluxes for the radiation levels to obtain the heat flux for the heat-structure node rows.
RADCHYD	Combines the phasic heat fluxes due to radiation to determine the energy transferred to each hydro cell.
RADEMS	Evaluates the emissivity of radiation-level faces.
RADFP	Evaluates the gas and liquid radiative properties.
RADMAP	Defines the flow-regime map for evaluating gas and liquid radiative properties.

RADMOD1	Controls the radiation-model calculation.
RADPT	Defines pointers for radiation model arrays.
RADSOL	Solves for radiative surface heat fluxes and phasic energy to the fluid.
RBREAK	Reads BREAK data from the input file and creates a pointer table for these data.
RCNTL	Reads in signal-variable, trip, and controller input data.
RCOMP	Reads data common to most 1D components from input files and writes these data to output file.
RDBRVLT	Reads BREAK-component data from the specific-component-table specTableCom common block and stores that data in the breakCom common block.
RDCOM3	Controls reading of 3D VESSEL data from input file.
RDCOMP	Controls reading of component data from input file.
RDCRDS	Reads timestep cards until DTMIN <0 is encountered.
RDCRVS	Reads PUMP curves from input file.
RDDIM	Reads number of points on PUMP curves from input file.
RDFIVLT	Reads FILL-component data from the specific-component-table specTableCom common block and stores that data in the fillCom common block.
RDFLT	Reads component-specific data from the generic-component-table genTableCom common block and stores that data in common block FLTAB.
RDPIVLT	Reads PIPE-component data from the specific-component-table specTableCom common block and stores that data in the pipeCom common block.
RDPLVLT	Reads PLENUM-component data from the specific-component-table specTableCom common block and stores that data in the plenumCom common block

RDPRVLT	Reads PRIZER-component data from the specific-component-table specTableCom common block and stores that data in the prizerCom common block.
RDPTR	Reads REAL*8 array values from the container A array and converts them to INTEGER*4 array values.
RDPUVLT	Reads PUMP-component data from the specific-component-table specTableCom common block and stores that data in the pumpCom common block.
RDRDVLT	Reads heat-structure ROD- or SLAB-component data from the specific-component-table specTableCom common block and stores that data in the teeCom common block.
RDREST	Controls reading of component data from a restart dump file.
RDTBVLT	Reads TURB-component data from the specific-component-table specTableCom common block and stores that data in the turbCom common block.
RDTEVLT	Reads TEE- or SEPD-component data from the specific-component-table specTableCom common block and stores that data in the teeCom common block.
RDVAVLT	Reads VALVE-component data from the specific-component-table specTableCom common block and stores that data in the valveCom common block.
RDVLT	Calls the component-specific subroutine RDxxVLT for reading specific-component-table specTableCom common-block data and storing that data in the component-specific xxxxCom common block.
RDVSVLT	Reads VESSEL-component data from the specific-component-table specTableCom common block and stores that data in the vesselCom common block.
RDZMOM	Defines momentum cell reciprocal lengths and weighting factors.
READI	Reads integer data in I14 format.
READR	Reads real data in E14.6 format.

REBRK	Reads BREAK data from a restart dump and creates a pointer table for these data.
RECNTL	Reads the signal-variable, trip, and controller data from the restart file.
RECOMP	Reads data from a restart dump common to most 1D components.
REECHO	Outputs real-valued scalar input data read from the TRCRST file to the TRCOUT file.
REFILL	Reads FILL data from a restart dump and creates a pointer table for these data.
REHTST	Reads heat-structure scalar input data from a restart dump and creates a pointer table for these data.
RENC	Reads radiation-model enclosure input data.
RENC1	Reads radiation-model enclosure input data.
REPIPE	Reads PIPE data from a restart dump and creates a pointer table for these data.
REPLEN	Reads PLENUM data from a restart dump and creates a pointer table for these data.
REPRZR	Reads PRIZER (pressurizer) data from a restart dump and creates a pointer table for these data.
REPUMP	Reads PUMP data from a restart dump and creates a pointer table for these data.
RERAD	Reads radiation-model enclosure data from a restart dump.
REROD1	Reads heat-structure input-data arrays from a restart dump.
RETEE	Reads TEE data from a restart dump and creates a pointer table for these data.
RETURB	Reads TURB (turbine) stage data from a restart dump and creates a pointer table for these data.
REVLVE	Reads VALVE data from a restart dump and creates a pointer table for these data.

REVSSL	Reads VESSEL data from a restart dump and creates a pointer table for these data.
RFDBK	Evaluates the reactor core reactivity feedback caused by changes in the fuel temperature, coolant temperature, and coolant void from the beginning of the previous timestep.
RFILL	Reads FILL data from input file and creates a pointer table for these data.
RHOLID	Evaluates the D2O liquid density and its derivatives.
RHOLIH	Evaluates the H2O liquid density and its derivatives.
RHOLIQ	Determines the D2O or H2O liquid density and its derivatives by calling RHOLID or RHOLIH.
RHTSTR	Reads ROD or SLAB heat-structure data from the input file and creates a pointer table for these data.
RKIN	Solves the neutron point-reactor kinetics differential equations.
RLEVEL	Writes real VESSEL level array to output file TRCOUT.
RODHT	Evaluates the fuel-rod temperature field.
RPIPE	Reads PIPE data from the input file and creates a pointer table for these data.
RPLEN	Reads PLENUM data from the input file and creates a pointer table for these data.
RPRIZR	Reads PRIZER (pressurizer) data from input file and creates a pointer table for these data.
RPUMP	Reads PUMP data from input file and creates a pointer table for these data.
RRDLCM	Reads rod data from LCM.
RROD1	Reads basic ROD input parameters.
RROD2	Reads and checks array data for powered heat structures.

RSTBRVLT	Reads BREAK-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTFIVLT	Reads FILL-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTFLT	Reads generic-component-table genTableCom common-block data from the TRCRST file.
RSTPIVLT	Reads PIPE-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTPLVLT	Reads PLENUM-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTPRVLT	Reads PRIZER-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTPUVLT	Reads PUMP-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTRDVLT	Reads heat-structure ROD- or SLAB-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTTBVLT	Reads TURB-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTTEVLT	Reads TEE- or SEPD-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTVAVLT	Reads VALVE-component specific-component-table specTableCom common-block data from the TRCRST file.
RSTVLT	Calls component-specific subroutine RSTxxvlt to read specific-component-table specTableCom common-block data from the TRCRST file.
RSTVSVLT	Reads VESSEL-component specific-component-table specTableCom common-block data from the TRCRST file.
RTEE	Reads TEE data from input file and creates a pointer table for these data.

RTTR	Determines coefficient for momentum convection across the TEE internal junction.
RTURB	Reads TURB (turbine) stage data from input file and creates a pointer table for these data.
RVLVE	Reads VALVE data from input file and creates a pointer table for these data.
RVSLCM	Reads VESSEL data from LCM.
RVSSL	Reads VESSEL data from input file and creates a pointer table for these data.
S1DPTR	Sets pointers for 1D components.
SASUMT	Evaluates the sum of the magnitudes of vector elements.
SATDED	Evaluates the derivative of the saturation temperature with respect to pressure for D2O vapor.
SATDEH	Evaluates the derivative of the saturation temperature with respect to pressure for H2O vapor.
SATDER	Determines the derivative of the saturation temperature with respect to pressure for D2O or H2O vapor by calling SATDED or SATDEH.
SATPRD	Evaluates the saturation pressure of D2O vapor at a given vapor temperature.
SATPRH	Evaluates the saturation pressure of H2O vapor at a given vapor temperature.
SATPRS	Determines the saturation pressure of D2O or H2O vapor at a given temperature by calling SATPRD or SATPRH.
SATTMD	Evaluates the saturation temperature of D2O vapor at a given pressure.
SATTMH	Evaluates the saturation temperature of H2O vapor at a given pressure.
SATTMP	Determines the saturation temperature of D2O or H2O vapor at a given pressure by calling SATTMD or SATTMH.

SAVBD	Moves boundary information into component arrays.
SAXPYT	Performs single precision computation of Y = A * X + Y.
SCLMOM	Sets up geometric scale factors for velocities to improve momentum conservation.
SCLTBL	Scales input table according to scale factor passed by input routine.
SCMLCM	Checks for overflow. Transfers fixed-length, variable-length, and pointer tables to LCM. Adjusts pointers.
SCOPYM	Support subroutine for SGEEV that copies one vector into another vector.
SCOPYT	Support subroutine for SGEEV that copies the negative of one vector into another vector.
SDOTT	Computes single precision inner product of single precision vectors.
SEDIT	Writes a small edit to the TRCOUT file.
SEPDI	Computes separator side-arm void fraction and mixture velocity.
SEPDX	Computes mechanistic separator carryover and carryunder quantities.
SETBD	Stores component information in boundary arrays.
SETBDT	Sets values for boundary to first theta cell equal to values for last theta cell and sets values for boundary to last theta cell equal to values for first theta cell.
SETEOD	Defines the equation-of-state constants for D2O fluid.
SETEOH	Defines the equation-of-state constants for H2O fluid.
SETEOS	Defines the equation-of-state constants for D2O or H2O fluid by calling SETEOD or SETEOH.
SETLCM	Monitors use of LCM dynamic area.

SETNET	Provides the information needed to set up the network solution matrices.
SETROD	Sets the value of a selected variable from a heat-structure ROD- or SLAB-component specific-component-table specTableCom common block.
SETTYPE	Defines the component-type internal-code REAL*8 value given its CHARACTER*8 component-type name.
SETVA	Sets value of variable VAR to VAL for one level of VESSEL data.
SFA22V	Hardwired version of SGEFAT for 2 x 2 matrices evaluated as a NMAT-element vector.
SFA33V	Hardwired version of SGEFAT for 3 x 3 matrices evaluated as a NMAT-element vector.
SFA44	Hardwired version of SGEFAT for a 4 x 4 matrix.
SFA44V	Hardwired version of SGEFAT for 4 x 4 matrices evaluated as a NMAT-element vector.
SFA55	Handwired version of SGEFAT for a 5 x 5 matrix.
SFA55V	Hardwired version of SGEFAT for 5 x 5 matrices evaluated as a NMAT-element vector.
SGECOT	Factors a real matrix by Gaussian elimination and estimates the condition of the matrix.
SGEDIT	Computes the determinant of a matrix using the factors computed by SGEFAT.
SGEEV	Computes the eigenvalues and eigenvectors of a general real matrix. SGEFAT Factors a real matrix by Gaussian elimination.
SGEFAT	Factors a real matrix by Gaussian elimination.
SGEFST	Solves a N x N system of linear equations by calling SGECOT and SGESLT.
SGESLT	Solves the real system A * X = B or TRANS(A) * X = B using the factors computed by SGEFAT.

SHIFTB	Translates the table's abscissa-coordinate values so that the function value F in the table corresponds to an abscissa-coordinate value of 0.0.
SHRINK	Removes rows of conduction nodes within the heat-structure rods or slabs during reflood.
SIGMA	Returns surface tension of water as a function of pressure.
SOUND	Performs a homogeneous-equilibrium sound-speed calculation.
SPLIT	Reads appropriate data from PUMP curves.
SRTLP	Sorts components into loops and reorders them for the network solution.
SSCALT	Performs single precision vector scale X = A * X.
SSEPOR	Performs detailed calculation of a steam-water separator.
SSL22V	Hardwired version of SGESLT for 2 x 2 matrices evaluated as a NMAT-element vector.
SSL33V	Hardwired version of SGESLT for 3 x 3 matrices evaluated as a NMAT-element vector.
SSL44	Hardwired version of SGESLT for a 4 x 4 matrix.
SSL44V	Hardwired version of SGESLT for 4 x 4 matrices evaluated as a NMAT-element vector.
SSL55	Hardwired version of SGESLT for a 5 x 5 matrix.
SSL55V	Hardwired version of SGESLT for 5 x 5 matrices evaluated as a NMAT-element vector.
STBME	Sets up the stabilizing mass and energy equations.
STBME3	Sets up stabilizer mass and energy equations for the VESSEL component.
STBMPL	Sets up the stabilizing mass and energy equations for the PLENUM component.

STDIR	Defines the pressure-variation matrix equation for the current outer iteration.
STEADY	Generates a steady-state solution.
SVSET	Calls SVSET1, SVSET3, and SVSETH to determine location-dependent signal-variable parameters.
SVSET1	Evaluates signal-variable parameters with locations defined in 1D hydraulic components.
SVSET3	Evaluates signal-variable parameters with locations defined in a 3D VESSEL component.
SVSETH	Evaluates signal-variable parameters defined in a HTSTR component.
TBC1	Stores the TEE internal-junction momentum term and set flag when a JCELL main-channel interface is a TEE external junction.
TEE1	Controls TEE prepass.
TEE1X	Evaluates source for TEE side-leg hydrodynamics.
TEE2	Controls TEE outer iteration.
TEE3	Controls TEE postpass.
TEEMET	Evaluates the explicit third term of TEE internal-junction momentum convection.
TEEMF1	Evaluates the coefficient of the implicit first term of TEE internal-junction momentum convection.
TEEMF2	Evaluates the coefficient of the implicit second term of TEE internal-junction momentum convection.
TEEMOM	Evaluates the TEE internal-junction three momentum-convection terms by calling TEEMET, TEEMF1, and TEEMF2.
TEEX	Evaluates coefficients for flow-coupling at the TEE internal junction.
TF1D	Controls 1D hydrodynamics routines.

TF1DS	Solves the hydrodynamic equations for the 1D two-fluid pipe model.
TF1DS1	Sets up initial velocity approximations and their pressure derivatives for the 1D two-fluid pipe model.
TF1DS3	Performs the backward-substitution for the 1D two-fluid pipe model.
TF3DS	Sets up basic mass and energy equations for 3D VESSEL component.
TF3DS1	Estimates new-time velocities from motion equation and evaluates variation of velocities with respect to pressure for 3D VESSEL component.
TF3DS3	Performs back-substitution for 3D VESSEL component.
TFPLBK	Performs the backward-substitution for the basic difference equations for the PLENUM (similar to TF1DS3 for the other 1D components).
TFPLN	Solves the basic hydrodynamic equations for the PLENUM (similar to TF1DS for the other 1D components).
THCL	Determines the thermal conductivity of D2O or H2O as a function of pressure and enthalpy by calling THCLD or THCLH.
THCLD	Evaluates the thermal conductivity of D2O as a function of pressure and enthalpy.
THCLH	Evaluates the thermal conductivity of H2O as a function of pressure and enthalpy.
THCV	Evaluates thermal conductivity of steam as a function of pressure and enthalpy.
THERMD	Evaluates the thermodynamic properties of D2O.
THERMH	Evaluates the thermodynamic properties of H2O.
THERMO	Determines the thermodynamic properties of D2O or H2O by calling THERMD or THERMH.

TIMCHK	Checks elapsed time to see whether certain functions should be performed.
TIMED	Determines the time and date for the IBM Risc/6000 and SUN computer platforms.
TIMSTP	Sets up timestep and time-edit interval times.
TIMUPD	Updates start-of-timestep values with end-of-timestep values for one VESSEL level.
TMPPTR	Sets up temporary pointers for subroutines PREIFD and PREFWD.
TMSFB	Evaluates the minimum stable film-boiling temperature (T_{min}).
TRAC	TRAC-M main program.
TRANS	Controls overall calculation for each timestep.
TRBPOW	Evaluates the efficiency and power output of a turbine stage.
TRBPRE	Evaluates the data pertaining to the entire turbine-generator set (common/sum all stages) during the prep stage.
TRBPST	Evaluates the data pertaining to the entire turbine-generator set (common/sum all stages) during the post stage.
TRIP	Returns status of a trip.
TRIPS	Evaluates the control parameters for the beginning of the timestep system state.
TRISLV	Solves linear system of the form A * X = B where A is tridiagonal.
TRPSET	Sets up trip status flags.
TURB1	Performs the prep stage calculation for the turbine stage component timestep initialization.
TURB2	Controls turbine stage outer iteration.
TURB3	Controls turbine stage postpass.

UNCNVT	Converts a parameter's value from SI to English units or from English to SI units.
UNNUMB	Assigns the units-label number to a parameter name in array LABELS.
UNSVCB	Determines the units label and units-label number of a signal variable or control block.
VALUE	Converts an ASCII string to its binary value.
VELBC	Sets velocities at internal FILL boundaries for a vessel.
VFWALL3	Evaluates 3D wall shear coefficients.
VISCL	Determines the viscosity of D2O or H2O liquid as a function of pressure and enthalpy by calling VISCLD or VISCLH.
VISCLD	Evaluates the viscosity of D2O liquid as a function of pressure and enthalpy.
VISCLH	Evaluates the viscosity of H2O liquid as a function of pressure and enthalpy.
VISCV	Determines the viscosity of D2O or H2O vapor as a function of pressure and enthalpy by calling VISCVD or VISCVH.
VISCVD	Evaluates the viscosity of D2O vapor as a function of pressure and enthalpy.
VISCVH	Evaluates the viscosity of H2O vapor as a function of pressure and enthalpy.
VLVE1	Controls VALVE prepass.
VLVE2	Controls VALVE outer iteration.
VLVE3	Controls VALVE postpass.
VLVEX	Evaluates the value of the flow-area change action for a VALVE.
VMCELL	Converts a VESSEL cell number to a VESSEL-matrix cell number.
VOLFA	Evaluates cell volume flow areas.

VOLV	Evaluates cell-averaged phase velocities for 1D components.
VRBD	Defines VESSEL velocities in the upstream radial direction for the inner ring (not currently used).
VSSL1	Performs prepass calculations for VESSEL dynamics.
VSSL2	Performs inner iterations for VESSEL dynamics.
VSSL3	Performs postpass calculations for VESSEL dynamics.
VSSROD	Transfers data between hydro and heat-structure databases.
VSSSSR	Performs steady-state change ratio calculations for vessel.
WARRAY	Writes a real array to output file TRCOUT.
WBREAK	Writes selected BREAK data to output file TRCOUT.
WCOMP	Controls the writing of selected component data to output file TRCOUT.
WDRAG	Evaluates coefficient of friction for liquid and vapor at the wall.
WFILL	Writes selected FILL data to output file TRCOUT.
WHTSTR	Writes selected heat-structure data to output file TRCOUT.
WIARN	Converts REAL*8-array values to integer-array values and then writes the integer-array values to output file TRCOUT.
WIR	Writes one to five real or integer variable values to a character string.
WJCELL	Evaluates the JCELL width seen by the adjacent side-channel cell from which the pressure gradient across the internal junction is defined.
WLABI	Edits labeled integer-valued input data that is to be read by the LOAD subroutine.
WLABIN	Writes labeled integer-array values input with the load format to output file TRCOUT and converts the integer-array values to REAL*8-array values.

WLABR	Edits labeled real-valued input data that is to be read by the LOAD subroutine.
WLEVEL	Writes real VESSEL level array to output file TRCOUT.
WMXYTB	Converts the units of input-array tabular data with 1 to 4 independent variable parameters for output to the INLAB or TRCOUT files and to SI units for the TRAC calculation.
WPIPE	Writes selected PIPE data to output file TRCOUT.
WPLEN	Writes selected PLENUM quantities to the output file TRCOUT.
WPRIZR	Writes selected PRIZER (pressurizer) data to output file TRCOUT.
WPUMP	Writes selected PUMP data to output file TRCOUT.
WRBRVLT	Writes breakCom common block data to the BREAK-component specific-component-table specTableCom common block.
WRCOMP	Writes data common to 1D components to output files.
WRFIVLT	Writes fillCom common block data to the FILL-component specific-component-table specTableCom common block.
WRFLT	Writes component-specific data from common block FLTAB to the generic-component-table genTableCom common block.
WRPIVLT	Writes pipeCom common block data to the PIPE-component specific-component-table specTableCom common block.
WRPLVLT	Writes plenumCom common block data to the PLENUM-component specific-component-table specTableCom common block.
WRPRVLT	Writes prizerCom common block data to the PRIZER-component specific-component-table specTableCom common block.
WRPTR	Converts INTEGER*4 array values to REAL*8 array values and writes them to the container A array.

WRPUVLT	Writes pumpCom common block data to the PUMP-component specific-component-table specTableCom common block.
WRRDVLT	Writes rodCom common block data to the heat-structure ROD- or SLAB-component specific-component-table specTableCom common block.
WRTBVLT	Writes turbCom common block data to the TURB-component specific-component-table specTableCom common block.
WRTEVLT	Writes teeCom common block data to the TEE- or SEPD-component specific-component-table specTableCom common block.
WRVAVLT	Writes valveCom common block data to the VALVE-component specific-component-table specTableCom common block.
WRVLT	Calls the component-specific subroutine WRxxVLT to write xxxxCom common block data to the specific-component-table specTableCom common block.
WRVSVLT	Writes vesselCom common block data to the VESSEL-component specific-component-table specTableCom common block.
WTEE	Writes selected TEE data to output file TRCOUT.
WTURB	Writes selected quantities to the printer for a TURB (turbine) stage component.
WVLVE	Writes selected VALVE data to output file TRCOUT.
WVSSL	Writes selected VESSEL data to output file TRCOUT.
XTV1D	Writes index and data for generic variables of 1D components for XTV graphics.
XTVBI3E	Converts values to IEEE format under UNICOS for XTV graphics.
XTVBUF	Buffers data to be sent to a C binary write routine and converts to IEEE format under UNICOS for XTV graphics.

XTVCB	Writes index and data for control-block output-parameter values for XTV graphics.
XTVDR	Main XTV driver routine that calls appropriate component specific routine to perform a function.
XTVHT	Writes index and data for heat-structure component variables for XTV graphics.
XTVINIT	Defines names for all output variables, opens header file, and calls CXTVTIN to set the maximum datafile size for XTV graphics.
XTVPIPE	Writes index and data for PIPE variables and calls XTV1D for generic 1D variables for XTV graphics.
XTVPLEN	Writes index and data for PLENUM variables for XTV graphics.
XTVPRZR	Writes index and data for PRIZER variables for XTV graphics.
XTVPUMP	Writes index and data for PUMP variables and calls XTV1D for generic 1D variables for XTV graphics.
XTVSIG	Writes index and data for signal-variables parameter values for XTV graphics.
XTVTEE	Writes index and data for TEE variables and calls XTV1D for generic 1D variables for XTV graphics.
XTVVALV	Writes index and data for VALVE variables and calls XTV1D for generic 1D variables for XTV graphics.
XTVVSL	Writes index and data for VESSEL variables for XTV graphics.
ZCORE	Evaluates axial locations for CHF and transition boiling within the core and computes associated void fractions.
ZEROV	Zeroes velocities at zero flow areas.
ZPWHCI	Evaluates axial power shape based on user input.
ZPWNRM	Normalizes the 1D or 2D axial-power distribution to a spatially averaged value of unity.

ZPWRCI Interpolates the r- or x-direction power shapes from ZPWF at the axial locations of the node rows.

APPENDIX B

TRAC-M SUBROUTINE CALLING SEQUENCE

TRAC						
	Calls	DMPIT	INIT	CUSRTIM	ERROR	INPUT
		SETLCM	BLKDAT	STEADY	BLKDAT2	TRANS
		CLEAN	C2R			
AICOMP						
	Calls	ERROR	RDLCM			
	Called by	FIND				
ALLBLK						
	Calls	INDEL				
	Called by	PREINP				
ASIGN						
	Called by	INPUT				
ASTPLN						
	Called by	PLEN3				
AUXPLN						
	Calls	BTESTC				
	Called by	PLEN2				
BACIT						
	Called by	VSSL2				
BAKUP						
	Called by	VSSL2	VSSL3			
BALANCT						
	Called by	SGEEV				
BALBAKT						
	Called by	SGEEV				
BANSOL						
	Called by	RODHT				
BDPLEN						
	Calls	IBSETC				
	Called by	IPLEN	PLEN2	PLEN3	PLEN1	

BFALOC

	Called by	DMPIT	RDREST			

BFCLOS

	Calls	C2R				
	Called by	ENDDMP				

BFIN

	Calls	ERROR	R2C			
	Called by	RDREST	REROD1	RSTPLVLT	REBRK	RETEE
		RSTPRVLT	RECOMP	RETURB	RSTPUVLT	REFILL
		REVLVE	RSTRDVLT	REHTST	REVSSL	RSTTBVLT
		REPIPE	RSTBRVLT	RSTTEVLT	REPLEN	RSTFIVLT
		RSTVAVLT	REPUMP	RSTFLT	RSTVSVLT	RERAD
		RSTPIVLT				

BFOUT

	Calls	ERROR				
	Called by	DBRK	DMPPIVLT	DPIPW	DCOMP	DMPPLVLT
		DPLEN	DFILL	DMPPRVLT	DPUMP	DHTSTR
		DMPPUVLT	DRAD	DLEVEL	DMPRDVLT	DROD1
		DMPBRVLT	DMPTBVLT	DTEE	DMPFIVLT	DMPTEVLT
		DTURB	DMPFLT	DMPVAVLT	DVLVE	DMPIT
		DMPVSVLT	DVSSL			

BKMOM

	Calls	BKSMOM				
	Called by	PIPE1	PUMP1	TURB1	PRIZR1	TEE1
		VLVE1				

BKSMOM

	Called by	BKMOM				

BKSPLN

	Calls	SFA55	CONCF	SSL55		
	Called by	PLEN3				

BKSSTB

	Calls	SSL55	CONCF	SFA55		
	Called by	POSTER				

BKSTB3

	Calls	IBSETC	SFA55	BTESTC	SSL55	CONCF
	Called by	VSSL3				

BLKDAT

	Called by	TRAC				

BLKDAT2

	Called by	TRAC				

BREAK1

	Calls	BREAKX	J1D	SHIFTB	BTESTC	
	Called by	PREP1D				

BREAK2

	Calls	J1D				
	Called by	OUT1D				

BREAK3

	Calls	FPROP	THERMO	J1D		
	Called by	POST				

BREAKX

	Calls	SHIFTB	ERROR	LININT0	THERMO	EVLTAB
		MIXPRP	TRIP	FPROP	SATTMP	
	Called by	BREAK1				

BTESTC

	Called by	AUXPLN	BKSTB3	BREAK1	CHKBD	CIF3
		ECOMP	FEMOM	FEMOMX	FEMOMY	FEMOMZ
		FILL1	FLUX	HTIF	INITBC	IVSSL
		J3D	PLEN3	POSTER	PREPER	STBME
		STBME3	TEE3	TF1DS	TF1DS1	TF1DS3
		TF3DS	TF3DS1	TF3DS3	TFPLBK	TFPLN
		VELBC	VSSL2	VSSL3	WRCOMP	

C2R

	Called by	BFCLOS	RCNTL	RPUMP	CBSET	RFILL
		RVLVE	DMPIT	RHTSTR	TRAC	INPUT

CBEDIT

	Called by	RCNTL	RECNTL			

CBSET

	Calls	ERROR	LINT4D	C2R	CONBLK	LININT0
		DELAY				
	Called by	TRIPS				

CDTHEX

	Calls	LININT0				
	Called by	DELTAR				

CELLA3

	Called by	VSSL2				

CELLAV

Called by	TF1D				

CEPSILON

Called by	HQR2T	HQRT	SGEFST		

CGCLOSE

No Callers

CGOPENA

No Callers

CGOPENR

No Callers

CGOPENW

No Callers

CGREAD

No Callers

CGWRITE

No Callers

CHBD

Calls	ERROR	GETTYPE			
Called by	CHKBD				

CHBSAV

Called by	IPIPE	IPUMP	ITURB	IPRIZR	ITEE
	IVLVE				

CHBSET

Called by	IPIPE	IPUMP	ITURB	IPRIZR	ITEE
	IVLVE				

CHECKSIZE

Calls	ERROR				
Called by	ICOMP	POST3D	RDREST	IHPSS3	PREP1D
	REBRK	INPUT	PREP3D	REFILL	LCMTRN
	RADSOL	RFILL	OUT3D	RBREAK	SCMLCM
	OUTER	RCNTL	VSSL2	POST	

CHEN

Calls	SATPRS				
Called by	HTCOR	HTVSSL			

CHF

Calls	CHF1	SATPRS	ERROR		
Called by	HTCOR	HTVSSL			

CHF1						
	Called by	CHF	HTCOR	HTVSSL		
CHKBD						
	Calls	CHBD	BTESTC			
	Called by	IPIPE IVLVE	IPUMP	ITURB	IPRIZR	ITEE
CHKSR						
	Calls	ERROR				
	Called by	RVSSL				
CHOKE						
	Calls	SGESLT SGEEV	ERROR THERMO	SATPRS SGEFAT	SOUND	SGEDIT
	Called by	TF1DS1				
CIF3						
	Calls	BTESTC				
	Called by	VSSL1				
CIHTST						
	Calls	ERROR IROD	IRODL	RRDLCM	RDFLT	WRVLT
	Called by	ICOMP				
CIRAD						
	Calls	CIRADH	CIRADR			
	Called by	ICOMP				
CIRADH						
	Calls	ERROR	FIND			
	Called by	CIRAD				
CIRADR						
	Calls	ERROR	FINDH			
	Called by	CIRAD				
CIVSSL						
	Calls	IVSSL IHPSS3	RVSLCM WRVLT	ERROR	JFIND	WRPTR
	Called by	ICOMP				

CLEAN

	Calls	ENDDMP				
	Called by	ERROR	STEADY	TRAC		

CLEAR

	Called by	CORE1	OUT1D	RBREAK	HOUT	OUT3D
		RCOMP	HTSTRP	OUTER	REROD1	ICOMP
		PLEN1	REVSSL	IHPSS3	PLEN2	RFILL
		INPUT	PLEN3	RHTSTR	IPLEN	POST
		RPLEN	IPUMP	POST3D	RROD2	IVSSL
		PREP1D	RVSSL	LCMTRN	PREP3D	SCMLCM
		LOADN	PREPER			

CLEARI

	Called by	INPUT	PNTVSS	S1DPTR	LOADN	RCNTL
		SEDIT	OUT3D	RDDIM	SRTLP	PNTROD

CLRBRVLT

	Called by	RBREAK

CLRFIVLT

	Called by	RFILL

CLRPIVLT

	Called by	RPIPE

CLRPLVLT

	Called by	RPLEN

CLRPRVLT

	Called by	RPRIZR

CLRPUVLT

	Called by	RPUMP

CLRRDVLT

	Called by	RHTSTR

CLRTEVLT

	Called by	RTEE

CLRVAVLT

	Called by	RVLVE

CLRVSVLT

	Called by	RVSSL

CMPLX

	Called by	CONBLK	HQR2T

COMPI

	Called by	IPIPE	IPUMP	ITURB	IPRIZR	ITEE
		IVLVE				

CONBLK						
	Calls	ERROR				
	Called by	CBSET				
CONCF						
	Called by	BKSPLN	BKSTB3	FF3D	BKSSTB	
CONSTB						
	Calls	STBME	J1D			
	Called by	PIPE3	PUMP3	TURB3	PRIZR3	TEE3
		VLVE3				
COPYA						
	Called by	MIX3D				
CORE1						
	Calls	SHRINK	CLEAR	MANAGE	ERROR	TRIP
		EVFXXX	MFROD	UNCNVT	EXPAND	VISCV
		FNMESH	ZCORE	GETROD	ZPWHCI	HTCOR
		RFDBK	ZPWNRM	HTVSSL	RKIN	ZPWRCI
		SETROD				
	Called by	HTSTR1				
CORE3						
	Calls	FROD	ERROR	EVALDF	MANAGE	
	Called by	HTSTR3				
COURNO						
	Called by	INPUT	NEWDLT			
CPLL						
	Calls	CPLLD	CPLLH			
	Called by	FPROPD	FPROPH			
CPLLD						
	Called by	CPLL				
CPLLH						
	Called by	CPLL				
CPVV1						
	Calls	CPVV1D	CPVV1H			
	Called by	FPROPD	HTCOR	HVWEBB	FPROPH	HTVSSL
CPVV1D						
	Called by	CPVV1				

CPVV1H

Called by CPVV1

CRSTIME

Called by DATER TIMED

CUSRTIME

No Callers

CVMGT

Called by FEMOMX FEMOMZ TF3DS1 FEMOMY TF3DS

CWVSSL

Calls RVSLCM WVSSL

Called by WCOMP

CXTVBW

Called by XTVBI3E

CXTVCL

Called by INIT XTVDR

CXTVIN

Called by XTVINIT

CXTVOA

Called by XTVDR

CXTVOW

Called by INIT

CYLHT

Called by POSTER

DATER

Calls CRSTIME

Called by INPUT

DAXPY

Called by DGBFA DGBSL

DBRK

Calls BFOUT DMPVLT DMPFLT RDVLT

Called by DMPIT

DCODF

Called by LOADN

DCOMP

Calls BFOUT DMPVLT RDVLT DMPFLT RDPTR

	Called by	DMPIT	DPUMP	DTURB	DPIPE	DTEE
		DVLVE				
DDOT						
	Called by	DGBSL				
DELAY						
	Calls	ERROR	LININT0			
	Called by	CBSET				
DELTAR						
	Calls	CDTHEX	FTHEX			
	Called by	GAPHT				
DFILL						
	Calls	BFOUT	DMPVLT	DMPFLT	RDVLT	
	Called by	DMPIT				
DGBFA						
	Calls	DAXPY	IDAMAX	DSCAL		
	Called by	MATSOL				
DGBSL						
	Calls	DAXPY	DDOT			
	Called by	MATSOL				
DHTSTR						
	Calls	BFOUT	RDFLT	DMPFLT	DMPVLT	RRDLCM
		DROD1				
	Called by	DMPIT				
DLEVEL						
	Calls	BFOUT	LEVELI			
	Called by	DVSSL				
DMPBRVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPFIVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPFLT						
	Calls	BFOUT				

	Called by	DBRK	DFILL	DPLEN	DCOMP	DHTSTR
		DVSSL				
DMPIT						
	Calls	BFALOC	DHTSTR	DVLVE	BFOUT	DPIPE
		DVSSL	C2R	DPLEN	ERROR	CUSRTIM
		DPUMP	GETTYPE	DBRK	DRAD	I42R
		DCOMP	DTEE	RDFLT	DFILL	DTURB
	Called by	ERROR	TIMCHK	TRANS	PSTEPQ	TRAC
DMPPIVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPPLVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPPRVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPPUVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPRDVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPTBVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPTEVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPVAVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DMPVLT						
	Calls	DMPBRVLT	DMPPRVLT	DMPTEVLT	DMPFIVLT	DMPPUVLT

		DMPVAVLT	DMPPIVLT	DMPRDVLT	DMPVSVLT	DMPPLVLT
		DMPTBVLT	ERROR			
	Called by	DBRK	DFILL	DPLEN	DCOMP	DHTSTR
		DVSSL				
DMPVSVLT						
	Calls	BFOUT				
	Called by	DMPVLT				
DPIPE						
	Calls	BFOUT	DCOMP			
	Called by	DMPIT				
DPLEN						
	Calls	BFOUT	DMPVLT	RDVLT	DMPFLT	RDPTR
	Called by	DMPIT				
DPUMP						
	Calls	BFOUT	DCOMP			
	Called by	DMPIT				
DRAD						
	Calls	BFOUT				
	Called by	DMPIT				
DROD1						
	Calls	BFOUT	MANAGE			
	Called by	DHTSTR				
DSCAL						
	Called by	DGBFA				
DTDIAG						
	Calls	FIND				
	Called by	TIMSTP				
DTEE						
	Calls	BFOUT	DCOMP			
	Called by	DMPIT				
DTURB						
	Calls	BFOUT	DCOMP			
	Called by	DMPIT				

DVLVE						
	Calls	BFOUT	DCOMP			
	Called by	DMPIT				
DVPSCL						
	Calls	SETVA				
	Called by	IVSSL	VSSL1			
DVSSL						
	Calls	BFOUT	DMPVLT	DLEVEL	MANAGE	RVSLCM
		DMPFLT				
	Called by	DMPIT				
ECOMP						
	Calls	BTESTC	UNCNVT			
	Called by	WBREAK	WPRIZR	WTURB	WFILL	WPUMP
		WVLVE	WPIPE	WTEE		
EDIT						
	Calls	SEDIT	UNCNVT	WCOMP		
	Called by	ERROR	PSTEPQ	TIMCHK	HOUT	STEADY
		TRANS				
ELGR						
	Calls	UNCNVT	ERROR	WARRAY	GETTYPE	
	Called by	IPIPE	IPUMP	ITURB	IPRIZR	ITEE
		IVLVE				
ENDDMP						
	Calls	BFCLOS	ERROR			
	Called by	CLEAN				
ERROR						
	Calls	CLEAN	EDIT	CUSRTIM	GETTYPE	DMPIT
	Called by	AICOMP	CORE3	GETROD	BFIN	DELAY
		GETTEE	BFOUT	DMPIT	GETTURB	BREAKX
		DMPVLT	GETVALVE	CBSET	ELGR	GETVSS
		CHBD	ENDDMP	HASH	EVALDF	CHECKSIZE
		HOUT	CHF	EVFXXX	HTSTR3	CHKSR
		EVLTAB	HTSTRP	CHOKE	FBRCSS	HVWEBB
		CIHTST	FEMOM	ICOMP	CIRADH	FILLX
		IHPSS1	CIRADR	FIND	IHPSS3	CIVSSL
		GETCRV	INIT	CONBLK	GETGEN	INPUT
		CORE1	GETPUMP	IROD	IRODL	READI
		SETROD	ITEE	READR	SETTYPE	IVLVE
		REBRK	SGEEV	JFIND	RECNTL	SGEFST

		JUNSOL	REFILL	SOUND	LOADN	REHTST
		SRTLP	LOCTRB	RENC	STEADY	LTOPP
		RENC1	SVSET	MANAGE	REPIPE	SVSET1
		MATSOL	REPLEN	SVSET3	MFROD	REPRZR
		SVSETH	MSTRCT	REPUMP	TEE1	NAMLST
		RETEE	TEEMET	NXTCMP	RETURB	TEEMF1
		OFFTKE	REVLVE	TEEMOM	OUT1D	REVSSL
		TF3DS	OUT3D	RFDBK	THERMD	OUTER
		RFILL	THERMH	POST	RHTSTR	TIMCHK
		POST3D	RKIN	TIMSTP	POSTER	RLEVEL
		TRAC	PREFWD	RODHT	TRANS	PREINP
		RPIPE	TRIP	PREP1D	RPLEN	TRIPS
		PREP3D	RPUMP	TRPSET	PTRSPL	RROD1
		UNCNVT	PUMPD	RROD2	UNNUMB	PUMPSR
		RSTVLT	UNSVCB	RBREAK	RTEE	VLVEX
		RCNTL	RTTR	VSSL1	RCOMP	RTURB
		VSSL2	RDCOMP	RVLVE	WIR	RDDIM
		RVSSL	WRVLT	RDREST	SCLMOM	XTVINIT
		RDVLT	SETLCM	ZPWRCI		
ETEE						
	Calls	TEEMET	TEEMF2			
	Called by	ITEE	TEE3			
EVALDF						
	Calls	ERROR				
	Called by	CORE3	PUMP3	VLVE3	PIPE3	TEE3
		VSSL3	PRIZR3	TURB3		
EVFXXX						
	Calls	ERROR	LININT0	TRIP	EVLTAB	
	Called by	CORE1	PUMP3	TEE3	PIPE1	RKIN
		TURB1	PIPE3	TEE1X	VLVE3	
EVLTAB						
	Calls	ERROR				
	Called by	BREAKX	FILLX	TRBPRE	EVFXXX	PUMPSR
		VLVEX				
EXIT_PROCESS						
	Called by	CXTVIN	CXTVBW			
EXPAND						
	Called by	CORE1				
FBRCSS						
	Calls	ERROR				
	Called by	INPUT				

FCEINF1

	Called by	FCEINFO				

FCEINFO

	Calls	FCEINF1				
	Called by	RENC				

FEMOM

	Calls	LEVEL	BTESTC	TEEMET	ERROR	TEEMF1
		TEEMF2				
	Called by	PREPER				

FEMOMX

	Calls	SATTMP	BTESTC			
	Called by	VSSL1				

FEMOMY

	Calls	SATTMP	BTESTC			
	Called by	VSSL1				

FEMOMZ

	Calls	SATTMP	BTESTC			
	Called by	VSSL1				

FF3D

	Calls	GVSSL1	CONCF	IBSETC		
	Called by	VSSL3				

FILL1

	Calls	BTESTC	FILLX	J1D		
	Called by	PREP1D				

FILL2

	Calls	J1D				
	Called by	OUT1D				

FILL3

	Calls	J1D				
	Called by	POST				

FILLX

	Calls	ERROR	LININT0	SHIFTB	EVLTAB	THERMO
		FPROP	MIXPRP	TRIP		
	Called by	FILL1				

FIND						
	Calls	AICOMP	ERROR	FINDER		
	Called by	CIRADH	IHPSS1	IHPSS3	DTDIAG	
FINDER						
	Calls	FINDNUM	RDFLT			
	Called by	FIND	FINDH			
FINDH						
	Calls	FINDER	LOC4	GETROD		
	Called by	CIRADR				
FINDNUM						
	Called by	FINDER				
FLTOM						
	Calls	PIPROD	VSSROD			
	Called by	HTSTR1	HTSTR3			
FLUX						
	Calls	BTESTC				
	Called by	PREPER				
FLUXES						
	Called by	VSSL2				
FNMESH						
	Called by	CORE1				
FPROP						
	Calls	FPROPD	FPROPH			
	Called by	BREAK3	IFILL	PLEN3	BREAKX	IPLEN
		POSTER	FILLX	IPROP	VSSL3	IBRK
		IVSSL				
FPROPD						
	Calls	CPLL	THCL	VISCL	CPVV1	THCV
		VISCV	SIGMA			
	Called by	FPROP				
FPROPH						
	Calls	CPLL	THCL	VISCL	CPVV1	THCV
		VISCV	SIGMA			
	Called by	FPROP				

FROD					
	Calls	GAPHT	MWRX	RODHT	
	Called by	CORE3			
FTHEX					
	Called by	DELTAR			
FWALL					
	Calls	FWKF			
	Called by	PREPER			
FWKF					
	Called by	FWALL	IWALL3		
GAPHT					
	Calls	DELTAR	MGAP		
	Called by	FROD			
GETCENC					
	Called by	RHTSTR			
GETCRV					
	Calls	ERROR	SPLIT		
	Called by	PUMPD			
GETGEN					
	Calls	ERROR			
	Called by	ICOMP	LCHPIP	PIPROD	IRODL
GETPUMP					
	Calls	ERROR			
	Called by	SVSET1			
GETRADM					
	Calls	GETRDM1			
	Called by	RHTSTR			
GETRDM1					
	Called by	GETRADM			
GETROD					
	Calls	ERROR			
	Called by	CORE1	FINDH		

GETTEE

	Calls	ERROR				
	Called by	ICOMP				

GETTURB

	Calls	ERROR				
	Called by	ICOMP				

GETTYPE

	Called by	CHBD	POST3D	WHTSTR	DMPIT	PREP1D
		XTV1D	ELGR	PREP3D	XTVHT	ERROR
		RDCOMP	XTVPLEN	IROD	RDREST	XTVVSL
		POST	SCMLCM			

GETVALVE

	Calls	ERROR				
	Called by	ICOMP	INPUT	SVSET1		

GETVSAR

	Called by	SVSET3				

GETVSS

	Calls	ERROR				
	Called by	LCHVSS				

GVSSL1

	Called by	FF3D				

GVSSL2

	Calls	SATTMP				
	Called by	VSSL3				

HASH

	Calls	ERROR				
	Called by	INPUT				

HEV

	Calls	HEVD	HEVH			
	Called by	SATDED	SATTMH	THERMD	SATDEH	SETEOD
		THERMH	SATTMD	SETEOH		

HEVD

	Called by	HEV				

HEVH

	Called by	HEV				

HLFILM

Called by	HTCOR				

HLFLMR

Called by	HTVSSL				

HOUT

Calls	ERROR	OUTER	CLEAR	POST	CUSRTIM
	UNCNVT	EDIT			
Called by	STEADY	TRANS			

HQR2T

Calls	CEPSILON				
Called by	SGEEV				

HQRT

Calls	CEPSILON				
Called by	SGEEV				

HTCOR

Calls	HLFILM	CHEN	HVFILM	CHF	HVNB
	CHF1	THCV	CPVV1	TMSFB	VISCV
Called by	CORE1	HTPIPE			

HTIF

Calls	IBSETC	BTESTC	SATPRS		
Called by	PLEN2	TF1D	VSSL2		

HTPIPE

Calls	HTCOR				
Called by	PREPER				

HTSTR1

Calls	MANAGE	RDFLT	CORE1	RRDLCM	FLTOM
	UNCNVT	HTSTRV	RADMODL	WRVLT	

HTSTR3

Calls	CORE3	MANAGE	RDFLT	ERROR	RRDLCM
	FLTOM	WRVLT	HTSTRP		
Called by	POST				

HTSTRP

Calls	ERROR	MANAGE	CLEAR		
Called by	HTSTR3				

HTSTRV

	Calls	MANAGE	RDFLT	SETVA	RVSLCM	
	Called by	HTSTR1				

HTVSSL

	Calls	HLFLMR	CHEN	HVNB	CHF	HVWEBB
		CHF1	THCV	CPVV1	VISCV	
	Called by	CORE1				

HUNTS

	Calls	IDEL				
	Called by	PREINP				

HVFILM

	Called by	HTCOR				

HVNB

	Called by	HTCOR	HTVSSL			

HVWEBB

	Calls	CPVV1	THCV	ERROR	VISCV	
	Called by	HTVSSL				

I42R

	Called by	DMPIT				

IBCLRC

	Called by	TF1DS	TF1DS3	TF3DS	TFPLN	

IBRK

	Calls	FPROP	MIXPRP	THERMO	J1D	WRFLT
		JFIND	WRVLT			
	Called by	ICOMP				

IBSETC

	Called by	BDPLEN	BKSTB3	FF3D	HTIF	INITBC
		J3D	PLEN3	POSTER	PREPER	RCOMP
		TF1DS	TF1DS1	TF1DS3	TF3DS	TF3DS3
		TFPLBK	TFPLN			

ICOMP

	Calls	IBRK	IVLVE	CIHTST	ITURB	CHECKSIZE
		IFILL	LOCTRB	CIRAD	IHPSS1	CIVSSL
		RDFLT	CLEAR	IPIPE	RDPTR	ERROR
		IPLEN	RDVLT	GETGEN	IPRIZR	SETLCM
		GETTEE	IPUMP	SETNET	GETTURB	ITEE
		WRVLT	GETVALVE			
	Called by	INIT				

IDAMAX						
	Called by	DGBFA				
IDEL						
	Called by	HUNTS	INPUT	PREINP		
IFILL						
	Calls	FPROP	THERMO	J1D	WRFLT	JFIND
		WRVLT	MIXPRP			
	Called by	ICOMP				
IFSET						
	Calls	SETVA				
	Called by	VSSL1				
IHPSS1						
	Calls	ERROR	FIND	THERMO	SATTMP	UNCNVT
	Called by	ICOMP				
IHPSS3						
	Calls	MANAGE	SATTMP	CLEAR	MATSOL	ERROR
		THERMO	FIND	CHECKSIZE		
	Called by	CIVSSL				
INDEL						
	Called by	ALLBLK	PREINP			
INIT						
	Calls	CXTVCL	ERROR	XTVDR	CXTVOW	ICOMP
		XTVINIT				
	Called by	TRAC				
INITBC						
	Calls	BTESTC	IBSETC	SETVA		
	Called by	IVSSL				
INNER						
	Calls	J1D	TF1D	ON1123C		
	Called by	PIPE2	PUMP2	TURB2	PRIZR2	TEE2
		VLVE2				
INPUT						
	Calls	LABELP	READR	ASIGN	LOADN	REECHO
		C2R	RENC	SETEOS	CLEAR	CHECKSIZE
		NAMLST	SETLCM	CLEARI	NXTCMP	SETTYPE
		COURNO	ORDER	SRTLP	DATER	PREINP

		ERROR	R2II	TIMED	FBRCSS	RCNTL
		UNCNVT	GETVALVE	RDCOM3	UNNUMB	HASH
		RDCOMP	VMCELL	RDREST	WARRAY	IDEL
		READI	WLABI	ISORT		
	Called by	TRAC				
IPIPE						
	Calls	CHBSAV	CHBSET	IPROP	SETBD	CHKBD
		JFIND	VOLFA	COMPI	JUNSOL	WRFLT
		ELGR	WRVLT			
	Called by	ICOMP				
IPLEN						
	Calls	BDPLEN	CLEAR	THERMO	FPROP	MIXPRP
		WRFLT	WRVLT	JFIND		
	Called by	ICOMP				
IPRIZR						
	Calls	ELGR	CHBSAV	IPROP	SETBD	CHBSET
		JFIND	VOLFA	CHKBD	JUNSOL	WRFLT
		COMPI	WRVLT			
	Called by	ICOMP				
IPROP						
	Calls	FPROP	MIXPRP	THERMO		
	Called by	IPIPE	IPUMP	ITURB	IPRIZR	ITEE
		IVLVE				
IPUMP						
	Calls	CHBSAV	CHBSET	IPROP	SETBD	CHKBD
		JFIND	VOLFA	CLEAR	JUNSOL	WRFLT
		COMPI	ELGR			
	Called by	ICOMP				
IROD						
	Calls	ERROR	LININT0	GETTYPE	MANAGE	UNCNVT
		ZPWHCI	ZPWRCI			
	Called by	CIHTST				
IRODL						
	Calls	ERROR	LCHVSS	GETGEN	LCHPIP	
	Called by	CIHTST				
ISORT						
	Called by	INPUT				

ITEE

Calls	ETEE	SETBD	CHBSAV	CHBSET	IPROP
	CHKBD	JFIND	VOLFA	COMPI	JUNSOL
	WJCELL	ELGR	WRFLT	ERROR	WRVLT
Called by	ICOMP				

ITURB

Calls	CHBSAV	IPROP	SETBD	CHBSET	JFIND
	TRBPOW	CHKBD	JUNSOL	VOLFA	COMPI
	WRFLT	ELGR	WRVLT		
Called by	ICOMP				

IVLVE

Calls	CHBSAV	CHBSET	IPROP	SETBD	CHKBD
	JFIND	VOLFA	COMPI	JUNSOL	WRFLT
	ELGR	WRVLT	ERROR		
Called by	ICOMP				

IVSSL

Calls	IWALL3	BTESTC	J3D	SCLMOM	CLEAR
	MANAGE	SETBDT	DVPSCL	FPROP	THERMO
	MIX3D	UNCNVT	INITBC	WLEVEL	RDZMOM
Called by	CIVSSL				

IWALL3

Calls	SETVA	FWKF			
Called by	IVSSL				

J1D

Called by	BREAK1	FILL1	IFILL	BREAK2	FILL2
	INNER	BREAK3	FILL3	JBD4	CONSTB
	IBRK	SETBD			

J3D

Calls	IBSETC	BTESTC	OF1123C	MANAGE	
Called by	IVSSL	VSSL1	VSSL3	POST3D	VSSL2

JBD4

Calls	J1D	WJCELL			
Called by	TEE1				

JFIND

Calls	ERROR				
Called by	CIVSSL	IPLEN	ITEE	IBRK	IPRIZR
	ITURB	IFILL	IPUMP	IVLVE	IPIPE

JUNSOL

	Calls	ERROR				
	Called by	IPIPE IVLVE	IPUMP	ITURB	IPRIZR	ITEE

JUSTLR

	Called by	RCNTL WMXYTB	RECNTL READR	WIARN WARRAY	READI	REECHO

JVALUE

	Called by	PREINP	VALUE

LABELD

	Called by	LABELP

LABELH

	Called by	LABELP

LABELP

	Calls	LABELD	LABELH
	Called by	INPUT	

LCHPIP

	Calls	GETGEN
	Called by	IRODL

LCHVSS

	Calls	GETVSS
	Called by	IRODL

LCMTRN

	Calls	SETLCM WRVLT	WRPTR	CLEAR	WRFLT	CHECKSIZE
	Called by	REHTST	RHTSTR	RVSSL	REVSSL	

LEVEL

	Called by	FEMOM	OFFTKE

LEVELI

	Called by	DLEVEL	WLEVEL

LEVELR

	Called by	REVSSL	RLEVEL	RVSSL

LININT

	Called by	PUMPD	PUMPX

LININT0

	Called by	BREAKX	FILLX	RROD2	CBSET	IROD
		RTEE	CDTHEX	MZIRC	RTURB	DELAY
		RFILL	RVLVE	EVFXXX	RPIPE	VSSL1
		EVLTAB	RPUMP			

LINT4D

	Called by	CBSET	RFDBK

LOADN

	Calls	CLEAR	DCODF	CLEARI	ERROR	
	Called by	INPUT	RFILL	RROD2	RBREAK	RHTSTR
		RTEE	RCNTL	RPIPE	RTURB	RCOMP
		RPLEN	RVLVE	RDCRVS	RPUMP	RVSSL

LOC4

	Called by	FINDH	LOCTRB	PIPROD	PNTVSS	REPLEN
		RETURB	REVSSL	RPLEN	RTURB	RVSSL

LOCF

	Called by	OUT1D	PREP1D

LOCTRB

	Calls	ERROR	LOC4
	Called by	ICOMP	

LTOPP

	Calls	ERROR		
	Called by	TEEMET	TEEMF1	TEEMF2

MANAGE

	Calls	ERROR				
	Called by	CORE1	IHPSS3	SVSETH	CORE3	IROD
		VSSL1	DROD1	IVSSL	VSSL2	DVSSL
		J3D	VSSL3	HTSTR1	POST3D	WHTSTR
		HTSTR3	RFDBK	WVSSL	HTSTRP	
		SVSET3	XTVHT	HTSTRV		

MATSOL

	Calls	DGBFA	ERROR	SGEFAT	DGBSL	SGESLT
	Called by	IHPSS3	POST3D	VSSL2	OUT3D	PREP3D

MBN

	Called by	MFROD

MFROD						
	Calls	ERROR MBN	MFUEL	MSTRCT	MHTR	MZIRC
	Called by	CORE1				
MFUEL						
	Called by	MFROD				
MGAP						
	Called by	GAPHT				
MHTR						
	Called by	MFROD				
MIX3D						
	Calls	COPYA				
	Called by	IVSSL	VSSL3			
MIXPRP						
	Called by	BREAKX IPROP	IBRK	IPLEN	FILLX	IFILL
MOVINFO						
	Called by	RENC1				
MPROP						
	Calls	MSTRCT				
	Called by	PREPER				
MSTRCT						
	Calls	ERROR				
	Called by	MFROD	MPROP			
MWRX						
	Called by	FROD				
MZIRC						
	Calls	LININT0				
	Called by	MFROD				
NAMLST						
	Calls	UNCNVT	ERROR			
	Called by	INPUT				
NEWDLT						
	Calls	COURNO	SEDIT			
	Called by	TIMSTP				

NXTCMP						
	Calls	ERROR				
	Called by	INPUT				
OF1123C						
	Called by	J3D	POSTER			
OFFTKE						
	Calls	LEVEL	ERROR			
	Called by	TEE3				
ON1123C						
	Called by	INNER				
ORDER						
	Called by	INPUT				
ORTHEST						
	Called by	SGEEV				
ORTRANT						
	Called by	SGEEV				
OUT1D						
	Calls	BREAK2	PIPE2	RDVLT	CLEAR	PLEN2
		SETLCM	ERROR	PRIZR2	TEE2	FILL2
		PUMP2	TURB2	LOCF	RDFLT	VLVE2
		RDPTR	WRVLT			
	Called by	OUTER				
OUT3D						
	Calls	ERROR	RVSLCM	CLEAR	MATSOL	CHECKSIZE
		VSSL2	CLEARI	RDFLT	WRVLT	
	Called by	OUTER				
OUTER						
	Calls	OUT1D	SGEFAT	CLEAR	OUT3D	CHECKSIZE
		SGESLT	ERROR			
	Called by	HOUT				
PIPE1						
	Calls	BKMOM	PIPE1X	SAVBD	EVFXXX	PREPER
		SETBD				
	Called by	PREP1D				
PIPE1X						
	Called by	PIPE1				

PIPE2						
	Calls	INNER				
	Called by	OUT1D				
PIPE3						
	Calls	CONSTB	EVFXXX	SAVBD	EVALDF	POSTER
		SETBD				
	Called by	POST				
PIPROD						
	Calls	GETGEN	LOC4	RDPTR		
	Called by	FLTOM				
PLEN1						
	Calls	BDPLEN	CLEAR			
	Called by	PREP1D				
PLEN2						
	Calls	HTIF	TFPLBK	AUXPLN	TFPLN	BDPLEN
		THERMO	CLEAR			
	Called by	OUT1D				
PLEN3						
	Calls	ASTPLN	CLEAR	BDPLEN	FPROP	STBMPL
		BKSPLN	IBSETC	THERMO	BTESTC	
	Called by	POST				
PNTROD						
	Calls	CLEARI				
	Called by	REHTST	RHTSTR			
PNTVSS						
	Calls	CLEARI	LOC4			
	Called by	REVSSL	RVSSL			
POST						
	Calls	BREAK3	PIPE3	SGEFAT	PLEN3	CHECKSIZE
		SGESLT	CLEAR	POST3D	TEE3	ERROR
		PRIZR3	TRBPST	FILL3	PUMP3	TURB3
		GETTYPE	RDFLT	VLVE3	HTSTR3	RDPTR
		WRFLT	RDVLT	WRVLT	SETLCM	
	Called by	HOUT	STEADY	TRANS		

POST3D

	Calls	RDFLT	CLEAR	J3D	RVSLCM	CHECKSIZE
		ERROR	MANAGE	VSSL3	GETTYPE	MATSOL
		WRVLT				
	Called by	POST				

POSTER

	Calls	FPROP	OF1123C	BKSSTB	POWINT	BTESTC
		IBSETC	CYLHT	THERMo	ERROR	
	Called by	PIPE3	PUMP3	TURB3	PRIZR3	TEE3
		VLVE3				

POWINT

	Called by	POSTER				

PREFWD

	Calls	ERROR	TMPPTR	SETLCM	VFWALL3	
	Called by	VSSL1				

PREINP

	Calls	ALLBLK	JVALUE	ERROR	IDEL	HUNTS
		INDEL	VALUE			
	Called by	INPUT				

PREP

	Calls	HTSTR1	PREP3D	TRIPS	PREP1D	
	Called by	STEADY	TRANS			

PREP1D

	Calls	BREAK1	PIPE1	SGEFAT	PLEN1	CHECKSIZE
		SGESLT	CLEAR	PRIZR1	TEE1	ERROR
		PUMP1	TRBPRE	FILL1	RDFLT	TURB1
		GETTYPE	RDPTR	VLVE1	LOCF	RDVLT
		WRVLT	SETLCM			
	Called by	PREP				

PREP3D

	Calls	GETTYPE	RVSLCM	CLEAR	MATSOL	CHECKSIZE
		VSSL1	ERROR	RDFLT	WRVLT	
	Called by	PREP				

PREPER

	Calls	HTPIPE	MPROP	BTESTC	PUMPSR	CLEAR
		IBSETC	FEMOM	FLUX	VOLV	FWALL
	Called by	PIPE1	PUMP1	TURB1	PRIZR1	TEE1
		VLVE1				

PRIZR1						
	Calls	BKMOM	PRZR1X	SETBD	PREPER	SAVBD
	Called by	PREP1D				
PRIZR2						
	Calls	INNER				
	Called by	OUT1D				
PRIZR3						
	Calls	CONSTB	POSTER	SETBD	EVALDF	SAVBD
	Called by	POST				
PRZR1X						
	Called by	PRIZR1				
PSTEPQ						
	Calls	CUSRTIM	EDIT	XTVDR	DMPIT	SEDIT
	Called by	STEADY	TRANS			
PTRSPL						
	Calls	ERROR	SETLCM			
	Called by	REPLEN	RPLEN			
PUMP1						
	Calls	BKMOM	SAVBD	SETBD	PREPER	
	Called by	PREP1D				
PUMP2						
	Calls	INNER				
	Called by	OUT1D				
PUMP3						
	Calls	CONSTB SETBD	EVFXXX	SAVBD	EVALDF	POSTER
	Called by	POST				
PUMPD						
	Calls	GETCRV	LININT	ERROR		
	Called by	PUMPX				
PUMPI						
	Called by	RDCRVS				

PUMPSR						
	Calls	PUMPX	ERROR	SHIFTB	EVLTAB	TRIP
	Called by	PREPER				
PUMPX						
	Calls	LININT	PUMPD	WARRAY		
	Called by	PUMPSR				
PUTRADM						
	Calls	PUTRDM1				
	Called by	RENC				
PUTRDM1						
	Called by	PUTRADM				
R2C						
	Called by	BFIN WCOMP	RECNTL	UNSVCB	R2C32	TRPSET
R2C32						
	Calls	R2C				
	Called by	RDREST XTVVSL	XTV1D WHTSTR	XTVPLEN	WCOMP	XTVHT
R2I4						
	Called by	RDREST				
R2II						
	Called by	INPUT				
RADCHTS						
	Called by	RADMODL				
RADCHYD						
	Called by	RADMODL				
RADEMS						
	Called by	RADMODL				
RADFP						
	Called by	RADMODL				
RADMAP						
	Called by	RADMODL				
RADMODL						
	Calls	RADCHYD RADFP	RADMAP	RADEMS	RADSOL	RADCHTS
	Called by	HTSTR1				

RADPT

	Called by	RENC				

RADSOL

	Calls	SFA44V	SSL33V	SFA55V	SSL44V	CHECKSIZE
		SFA22V	SGEFST	SSL55V	SFA33V	SSL22V
	Called by	RADMODL				

RBREAK

	Calls	SCLTBL	CLEAR	SETLCM	CLRBRVLT	CHECKSIZE
		READI	WARRAY	ERROR	READR	WRFLT
		S1DPTR	WRPTR	SATTMP	WRVLT	LOADN
	Called by	RDCOMP				

RCNTL

	Calls	READR	C2R	JUSTLR	UNCNVT	CBEDIT
		LOADN	UNNUMB	UNSVCB	CLEARI	CHECKSIZE
		WARRAY	ERROR	WLABI	READI	WMXYTB
	Called by	INPUT				

RCOMP

	Calls	CLEAR	LOADN	THERMO	ERROR	UNCNVT
		WARRAY	IBSETC	SATTMP	WIARN	
	Called by	RPIPE	RPUMP	RTURB	RPRIZR	RTEE
		RVLVE				

RDBRVLT

	Called by	RDVLT				

RDCOM3

	Calls	RVSSL				
	Called by	INPUT				

RDCOMP

	Calls	ERROR	RPIPE	RTEE	GETTYPE	RPLEN
		RTURB	RBREAK	RPRIZR	RVLVE	RFILL
		RPUMP	WRFLT	RHTSTR		
	Called by	INPUT				

RDCRDS

	Calls	READR				
	Called by	STEADY				

RDCRVS

Calls	LOADN	PUMPI	WARRAY		
Called by	RPUMP				

RDDIM

Calls	CLEARI	ERROR	READI		
Called by	RPUMP				

RDFIVLT

Called by	RDVLT				

RDFLT

Called by	CIHTST	OUT1D	SVSETH	DHTSTR	OUT3D
	TRBPRE	DMPIT	POST	TRBPST	FINDER
	POST3D	WCOMP	HTSTR1	PREP1D	WHTSTR
	HTSTR3	PREP3D	XTVDR	HTSTRV	SVSET
	XTVHT	ICOMP			

RDPIVLT

Called by	RDVLT				

RDPLVLT

Called by	RDVLT				

RDPRVLT

Called by	RDVLT				

RDPTR

Called by	AICOMP	PREP1D	XTVPIPE	DCOMP	RRDLCM
	XTVPLEN	DPLEN	RVSLCM	XTVPRZR	ICOMP
	SVSET	XTVPUMP	OUT1D	TRBPRE	XTVTEE
	PIPROD	TRBPST	XTVVALV	POST	WCOMP

RDPUVLT

Called by	RDVLT				

RDRDVLT

Called by	RDVLT				

RDREST

Calls	BFALOC	R2C32	REPUMP	BFIN	R2I4
	RERAD	REBRK	RETEE	ERROR	CHECKSIZE
	RECNTL	RETURB	GETTYPE	REFILL	REVLVE
	REHTST	REVSSL	REPIPE	RSTFLT	REPLEN
	SETLCM	REPRZR			
Called by	INPUT				

RDTBVLT

Called by	RDVLT				

RDTEVLT

	Called by	RDVLT				

RDVAVLT

	Called by	RDVLT				

RDVLT

	Calls	ERROR	RDPLVLT	RDTBVLT	RDBRVLT	RDPRVLT
		RDTEVLT	RDFIVLT	RDPUVLT	RDVAVLT	RDPIVLT
		RDRDVLT	RDVSVLT			
	Called by	DBRK	PREP1D	XTVPIPE	DCOMP	RRDLCM
		XTVPLEN	DFILL	RVSLCM	XTVPRZR	DPLEN
		SVSET	XTVPUMP	ICOMP	TRBPRE	XTVTEE
		OUT1D	TRBPST	XTVVALV	POST	WCOMP

RDVSVLT

	Called by	RDVLT				

RDZMOM

	Calls	SETVA				
	Called by	IVSSL				

READI

	Calls	ERROR	JUSTLR			
	Called by	INPUT	RHTSTR	RROD1	RBREAK	RPIPE
		RTEE	RCNTL	RPLEN	RTURB	RDDIM
		RPRIZR	RVLVE	RENC1	RPUMP	RVSSL
		RFILL				

READR

	Calls	ERROR	WIR	JUSTLR	UNCNVT	
	Called by	INPUT	RHTSTR	RTEE	RBREAK	RPIPE
		RTURB	RCNTL	RPRIZR	RVLVE	RDCRDS
		RPUMP	RVSSL	RFILL	RROD1	TIMSTP

REBRK

	Calls	BFIN	WARRAY	REECHO	WRFLT	CHECKSIZE
		ERROR	RSTVLT	WRPTR	S1DPTR	WRVLT
		SETLCM				
	Called by	RDREST				

RECNTL

	Calls	UNNUMB	CBEDIT	UNSVCB	ERROR	WARRAY
		R2C	WMXYTB	JUSTLR	REECHO	
	Called by	RDREST				

RECOMP

	Calls	BFIN				

	Called by	REPIPE	REPUMP	RETURB	REPRZR	RETEE
		REVLVE				
REECHO						
	Calls	JUSTLR	UNCNVT	WIR		
	Called by	INPUT	REPIPE	RETEE	REBRK	REPLEN
		RETURB	RECNTL	REPRZR	REVLVE	REFILL
		REPUMP	REVSSL	REHTST		
REFILL						
	Calls	BFIN	SETLCM	WARRAY	ERROR	CHECKSIZE
		REECHO	WRFLT	RSTVLT	WRPTR	S1DPTR
		WRVLT				
	Called by	RDREST				
REHTST						
	Calls	REROD1	BFIN	RSTVLT	ERROR	PNTROD
		UNNUMB	REECHO	UNSVCB	LCMTRN	
	Called by	RDREST				
RENC						
	Calls	ERROR	PUTRADM	RENC1	FCEINFO	RADPT
		SETLCM				
	Called by	INPUT				
RENC1						
	Calls	ERROR	READI	MOVINFO		
	Called by	RENC				
REPIPE						
	Calls	BFIN	REECHO	UNSVCB	ERROR	RSTVLT
		WARRAY	S1DPTR	WMXYTB	SCMLCM	WRCOMP
		RECOMP				
	Called by	RDREST				
REPLEN						
	Calls	BFIN	PTRSPL	SCMLCM	ERROR	REECHO
		WARRAY	RSTVLT	WIARN	LOC4	
	Called by	RDREST				
REPRZR						
	Calls	ERROR	REECHO	SCMLCM	RSTVLT	WRCOMP
		RECOMP	S1DPTR			
	Called by	RDREST				

REPUMP

	Calls	BFIN	REECHO	UNSVCB	ERROR	RSTVLT
		WARRAY	S1DPTR	WMXYTB	SCMLCM	WRCOMP
		RECOMP				
	Called by	RDREST				

RERAD

	Calls	BFIN	SETLCM
	Called by	RDREST	

REROD1

	Calls	BFIN	UNSVCB	CLEAR	WARRAY	UNCNVT
		WIARN	UNNUMB	WMXYTB		
	Called by	REHTST				

RETEE

	Calls	RECOMP	UNSVCB	BFIN	REECHO	WARRAY
		ERROR	RSTVLT	WMXYTB	S1DPTR	WRCOMP
		SCMLCM				
	Called by	RDREST				

RETURB

	Calls	BFIN	RECOMP	SCMLCM	ERROR	REECHO
		WARRAY	RSTVLT	WIARN	S1DPTR	WRCOMP
		LOC4				
	Called by	RDREST				

REVLVE

	Calls	BFIN	REECHO	UNSVCB	ERROR	RSTVLT
		WARRAY	S1DPTR	WMXYTB	SCMLCM	WRCOMP
		RECOMP				
	Called by	RDREST				

REVSSL

	Calls	BFIN	LEVELR	RSTVLT	CLEAR	LOC4
		WARRAY	ERROR	WIARN	PNTVSS	WRVLT
		LCMTRN	REECHO			
	Called by	RDREST				

RFDBK

	Calls	ERROR	LINT4D	MANAGE
	Called by	CORE1		

RFILL

	Calls	LININT0	S1DPTR	C2R	LOADN	CHECKSIZE
		SCLTBL	SETLCM	CLEAR	WARRAY	CLRFIVLT

		WRFLT	ERROR	READI	WRPTR	READR
		WRVLT				
	Called by	RDCOMP				
RHOLID						
	Called by	RHOLIQ				
RHOLIH						
	Called by	RHOLIQ				
RHOLIQ						
	Calls	RHOLID	RHOLIH			
	Called by	THERMD	THERMH			
RHTSTR						
	Calls	C2R	LCMTRN	RROD1	CLEAR	LOADN
		RROD2	CLRRDVLT	UNCNVT	ERROR	WARRAY
		GETCENC	PNTROD	WIARN	GETRADM	READI
		WRVLT	READR			
	Called by	RDCOMP				
RHVGET						
	No Callers					
RKIN						
	Calls	ERROR	TRIP	EVFXXX		
	Called by	CORE1				
RLEVEL						
	Calls	ERROR	LEVELR	WARRAY		
	Called by	RVSSL				
RODHT						
	Calls	BANSOL	TRISLV	ERROR		
	Called by	FROD				
RPIPE						
	Calls	LOADN	SCMLCM	CLRPIVLT	RCOMP	UNCNVT
		ERROR	READI	UNSVCB	READR	WARRAY
		S1DPTR	WMXYTB	LININT0	SCLTBL	WRVLT
	Called by	RDCOMP				

RPLEN

	Calls	CLEAR	LOADN	SCMLCM	CLRPLVLT	LOC4
		WARRAY	ERROR	PTRSPL	WIARN	READI
	Called by	RDCOMP				

RPRIZR

	Calls	CLRPRVLT	READI	S1DPTR	READR	SCMLCM
		RCOMP				
	Called by	RDCOMP				

RPUMP

	Calls	SCLTBL	C2R	RCOMP	SCMLCM	CLRPUVLT
		RDCRVS	THERMO	ERROR	RDDIM	UNCNVT
		READI	UNSVCB	READR	WARRAY	LININT0
		S1DPTR	WMXYTB	LOADN		
	Called by	RDCOMP				

RRDLCM

	Calls	RDPTR	RDVLT			
	Called by	CIHTST	HTSTR3	WHTSTR	DHTSTR	SVSETH
		XTVHT	HTSTR1			

RROD1

	Calls	ERROR	READI	UNNUMB	READR	UNSVCB
	Called by	RHTSTR				

RROD2

	Calls	LININT0	UNNUMB	CLEAR	LOADN	UNSVCB
		DECAYS	WARRAY	ERROR	WLABR	SCLTBL
		WMXYTB	UNCNVT	ZPWNRM		
	Called by	RHTSTR				

RSTBRVLT

	Calls	BFIN
	Called by	RSTVLT

RSTFIVLT

	Calls	BFIN
	Called by	RSTVLT

RSTFLT

	Calls	BFIN
	Called by	RDREST

RS_TIME
    No Callers

RSTPIVLT
    Calls BFIN

    Called by RSTVLT

RSTPLVLT
    Calls BFIN

    Called by RSTVLT

RSTPRVLT
    Calls BFIN

    Called by RSTVLT

RSTPUVLT
    Calls BFIN

    Called by RSTVLT

RSTRDVLT
    Calls BFIN

    Called by RSTVLT

RSTTBVLT
    Calls BFIN

    Called by RSTVLT

RSTTEVLT
    Calls BFIN

    Called by RSTVLT

RSTVAVLT
    Calls BFIN

    Called by RSTVLT

RSTVLT
    Calls ERROR RSTPLVLT RSTTBVLT RSTBRVLT RSTPRVLT RSTTEVLT RSTFIVLT RSTPUVLT RSTVAVLT RSTPIVLT RSTRDVLT RSTVSVLT

    Called by REBRK REPLEN RETURB REFILL REPRZR REVLVE REHTST REPUMP REVSSL REPIPE RETEE

RSTVSVLT

Calls	BFIN				
Called by	RSTVLT				

RTEE

Calls	LOADN	SCMLCM	CLRTEVLT	RCOMP	UNCNVT
	ERROR	READI	UNSVCB	READR	WARRAY
	S1DPTR	WMXYTB	LININT0	SCLTBL	WRVLT
Called by	RDCOMP				

RTTR

Calls	ERROR				
Called by	TEEMET	TEEMF1	TEEMF2		

RTURB

Calls	LOC4	SCMLCM	RCOMP	ERROR	READI
	UNCNVT	READR	WARRAY	WIARN	LININT0
	S1DPTR	WRVLT	LOADN	SCLTBL	
Called by	RDCOMP				

RVLVE

Calls	LOADN	SCMLCM	C2R	CLRVAVLT	ERROR
	RCOMP	THERMO	FAXPOS	READI	UNCNVT
	READR	UNSVCB	S1DPTR	WARRAY	LININT0
	SCLTBL	WMXYTB			

RVSLCM

Calls	RDPTR	RDVLT			
Called by	CIVSSL	HTSTRV	PREP3D	CWVSSL	OUT3D
	SVSET3	DVSSL	POST3D	XTVVSL	

RVSSL

Calls	READR	LCMTRN	CHKSR	LEVELR	RLEVEL
	CLEAR	LOADN	CLRVSVLT	LOC4	UNCNVT
	WARRAY	ERROR	PNTVSS	WIARN	READI
	WRVLT				
Called by	RDCOM3				

RVVGET

No Callers

S1DPTR

Calls	CLEARI				
Called by	RBREAK	RETEE	RPRIZR	REBRK	RETURB
	RPUMP	REFILL	REVLVE	RTEE	REPIPE
	RFILL	RTURB	REPRZR	RPIPE	RVLVE
	REPUMP				

SASUMT

Called by	SGECOT				

SATDED

Calls	HEV				
Called by	SATDER				

SATDEH

Calls	HEV				
Called by	SATDER				

SATDER

Calls	SATDED	SATDEH			
Called by	TF1DS	TFPLN	THERMH	TF3DS	THERMD

SATPRD

Called by	SATPRS				

SATPRH

Called by	SATPRS				

SATPRS

Calls	SATPRD	SATPRH			
Called by	CHEN	SATTMH	TF3DS3	CHF	SOUND
	TFPLBK	CHOKE	TF1DS	TFPLN	HTIF
	TF1DS3	THERMD	SATTMD	TF3DS	THERMH

SATTMD

Calls	SATPRS	HEV			
Called by	SATTMP				

SATTMH

Calls	SATPRS	HEV			
Called by	SATTMP				

SATTMP

Calls	SATTMD	SATTMH			
Called by	BREAKX	IHPSS3	TF3DS1	FEMOMX	RBREAK
	TF3DS3	FEMOMY	RCOMP	TFPLBK	FEMOMZ
	SOUND	THERMD	GVSSL2	TF1DS3	THERMH
	IHPSS1	TF3DS	TRBPOW		

SAVBD

Called by	PIPE1	PUMP1	TURB1	PIPE3	PUMP3
	TURB3	PRIZR1	TEE1	VLVE1	PRIZR3
	TEE3	VLVE3			

SAXPYT

	Called by	SGECOT				

SCLMOM

	Calls	ERROR				
	Called by	IVSSL				

SCLTBL

	Calls	WARRAY	WMXYTB	UNSVCB		
	Called by	RBREAK	RPUMP	RTURB	RFILL	RROD2
		RVLVE	RPIPE	RTEE		

SCMLCM

	Calls	WRPTR	CLEAR	SETLCM	WRVLT	CHECKSIZE
		GETTYPE	WRFLT			
	Called by	REPIPE	RETURB	RPUMP	REPLEN	REVLVE
		RTEE	REPRZR	RPIPE	RTURB	REPUMP
		RPLEN	RVLVE	RETEE	RPRIZR	

SCOPYM

	Called by	SGEEV				

SCOPYT

	Called by	SGEEV				

SDOTT

	Called by	SGECOT				

SEDIT

	Calls	CLEARI	UNCNVT	CUSRTIM		
	Called by	EDIT	NEWDLT	PSTEPQ		

SEPDI

	Called by	TEE2				

SEPDX

	Calls	SSEPOR				
	Called by	TEE1				

SETBD

	Calls	J1D				
	Called by	IPIPE	PIPE1	TEE1	IPRIZR	PIPE3
		TEE3	IPUMP	PRIZR1	TURB1	ITEE
		PRIZR3	TURB3	ITURB	PUMP1	VLVE1
		IVLVE	PUMP3	VLVE3		

SETBDT

	Calls	CUSRTIM				
	Called by	IVSSL	VSSL1	VSSL2		

SETEOD

	Calls	HEV				
	Called by	SETEOS				

SETEOH

	Calls	HEV				
	Called by	SETEOS				

SETEOS

	Calls	SETEOD	SETEOH			
	Called by	INPUT				

SETLCM

	Calls	ERROR				
	Called by	ICOMP	PTRSPL	RERAD	INPUT	RBREAK
		RFILL	LCMTRN	RDREST	SCMLCM	OUT1D
		REBRK	TRAC	POST	REFILL	WARRAY
		PREFWD	RENC	WMXYTB	PREP1D	

SETNET

	Called by	ICOMP				

SETROD

	Calls	ERROR				
	Called by	CORE1				

SETTYPE

	Calls	ERROR				
	Called by	INPUT				

SETVA

	Called by	DVPSCL	INITBC	RDZMOM	HTSTRV	IWALL3
		VSSL1	IFSET			

SFA22V

	Called by	RADSOL				

SFA33V

	Called by	RADSOL				

SFA44

	Called by	TF1DS	TFPLN			

SFA44V

	Called by	RADSOL				

SFA55

	Called by	BKSPLN	BKSTB3	TF3DS	BKSSTB	TF1DS
		TFPLN				

SFA55V

	Called by	RADSOL				

SGECOT

	Calls	SAXPYT	SDOTT	SSCALT	SASUMT	SGEFAT
	Called by	SGEFST				

SGEDIT

	Called by	CHOKE				

SGEEV

	Calls	BALANCT	HQRT	ORTRANT	BALBAKT	SCOPYM
		ERROR	ORTHEST	SCOPYT	HQR2T	
	Called by	CHOKE				

SGEFAT

	Called by	CHOKE	OUTER	PREP1D	MATSOL	POST
		SGECOT				

SGEFST

	Calls	CEPSILON	SGESLT	ERROR	SGECOT	
	Called by	RADSOL				

SGESLT

	Called by	CHOKE	OUTER	PREP1D	MATSOL	POST
		SGEFST				

SHIFTB

	Called by	BREAK1	FILLX	VLVEX	BREAKX	PUMPSR

SHRINK

	Called by	CORE1				

SIGMA

	Called by	FPROPD	FPROPH			

SOUND

	Calls	ERROR	SATPRS	THERMO	SATTMP	
	Called by	CHOKE				

SPLIT

	Called by	GETCRV				

SRTLP

	Calls	CREALI	ERROR			
	Called by	INPUT				
SSCALT						
	Called by	SGECOT				
SSEPOR						
	Called by	SEPDX				
SSL22V						
	Called by	RADSOL				
SSL33V						
	Called by	RADSOL				
SSL44						
	Called by	TF1DS	TFPLN			
SSL44V						
	Called by	RADSOL				
SSL55						
	Called by	BKSPLN	BKSTB3	TF3DS	BKSSTB	TF1DS
		TFPLN				
SSL55V						
	Called by	RADSOL				
STBME						
	Calls	BTESTC				
	Called by	CONSTB				
STBME3						
	Calls	BTESTC				
	Called by	VSSL3				
STBMPL						
	Called by	PLEN3				
STDIR						
	Called by	VSSL2				
STEADY						
	Calls	RDCRDS	CLEAN	TIMCHK	EDIT	POST
		TIMSTP	ERROR	PREP	XTVDR	HOUT
		PSTEPQ				
	Called by	TRAC				

SVSET

	Calls	ERROR	RDPTR	SVSET3	RDVLT	SVSETH
		RDFLT	SVSET1			
	Called by	TRIPS				
SVSET1						
	Calls	GETVALVE	ERROR	GETPUMP		
	Called by	SVSET				
SVSET3						
	Calls	ERROR	GETVSAR	MANAGE	RVSLCM	
	Called by	SVSET				
SVSETH						
	Calls	RDFLT	ERROR	MANAGE	RRDLCM	
	Called by	SVSET				
TBC1						
	Calls	TEEMOM				
	Called by	TEE1				
TEE1						
	Calls	PREPER	TBC1	BKMOM	SAVBD	TEE1X
		ERROR	SEPDX	TEEX	JBD4	SETBD
	Called by	PREP1D				
TEE1X						
	Calls	EVFXXX				
	Called by	TEE1	TEE2			
TEE2						
	Calls	INNER	SEPDI	TEE1X		
	Called by	OUT1D				
TEE3						
	Calls	BTESTC	EVFXXX	SAVBD	CONSTB	SETBD
		ETEE	OFFTKE	EVALDF	POSTER	
	Called by	POST				
TEEMET						
	Calls	ERROR	LTOPP	RTTR		
	Called by	ETEE	FEMOM	TEEMOM		

TEEMF1

	Calls	ERROR	LTOPP	RTTR		
	Called by	FEMOM	TEEMOM			

TEEMF2

	Calls	LTOPP	RTTR			
	Called by	ETEE	FEMOM	TEEMOM		

TEEMOM

	Calls	ERROR	TEEMF1	TEEMF2	TEEMET	
	Called by	TBC1	TF1DS1			

TEEX

	Called by	TEE1				

TF1D

	Calls	CELLAV	TF1DS	TF1DS3	HTIF	TF1DS1
		THERMO				
	Called by	INNER				

TF1DS

	Calls	SFA44	BTESTC	SFA55	IBCLRC	SATDER
		SSL44	IBSETC	SATPRS	SSL55	
	Called by	TF1D				

TF1DS1

	Calls	BTESTC	CHOKE	TEEMOM	IBSETC	
	Called by	TF1D				

TF1DS3

	Calls	SATTMP	BTESTC	IBCLRC	IBSETC	SATPRS
		THERMO				
	Called by	TF1D				

TF3DS

	Calls	SATTMP	BTESTC	SFA55	ERROR	SSL55
		IBCLRC	SATDER	THERMO	IBSETC	SATPRS
	Called by	VSSL2				

TF3DS1

	Calls	BTESTC	VELBC	SATTMP	ZEROV	
	Called by	VSSL2				

TF3DS3						
	Calls	SATTMP	BTESTC	IBSETC	SATPRS	THERMO
	Called by	VSSL2				
TFPLBK						
	Calls	SATPRS	BTESTC	SATTMP	IBSETC	THERMO
	Called by	PLEN2				
TFPLN						
	Calls	SFA44	BTESTC	SFA55	IBCLRC	SATDER
		SSL44	IBSETC	SATPRS	SSL55	
	Called by	PLEN2				
THCL						
	Calls	THCLD	THCLH			
	Called by	FPROPD	FPROPH			
THCLD						
	Called by	THCL				
THCLH						
	Called b	THCL				
THCV						
	Called by	FPROPD	HTCOR	HVWEBB	FPROPH	HTVSSL
THERMD						
	Calls	ERROR	SATPRS	RHOLIQ	SATTMP	HEV
		SATDER				
	Called by	THERMO				
THERMH						
	Calls	ERROR	SATPRS	SATTMP	HEV	RHOLIQ
		SATDER				
	Called by	THERMO				
THERMO						
	Calls	THERMD	THERMH			
	Called by	BREAK3	IPROP	SOUND	BREAKX	IVSSL
		TF1D	CHOKE	PLEN2	TF1DS3	FILLX
		PLEN3	TF3DS	IBRK	POSTER	TF3DS3
		IFILL	RCOMP	TFPLBK	IHPSS1	RPUMP
		VSSL2	IHPSS3	RVLVE	VSSL3	IPLEN

TIMCHK						
	Calls	CUSRTIM	EDIT	DMPIT	ERROR	
	Called by	STEADY	TRANS			
TIMED						
	Calls	CRSTIME				
	Called by	INPUT				
TIMSTP						
	Calls	READR	DTDIAG	TRIP	ERROR	NEWDLT
		UNCNVT				
	Called by	STEADY	TRANS			
TIMUPD						
	Called by	VSSL1				
TMPPTR						
	Called by	PREFWD				
TMSFB						
	Called by	HTCOR				
TRANS						
	Calls	PSTEPQ	DMPIT	TIMCHK	EDIT	POST
		TIMSTP	ERROR	PREP	XTVDR	HOUT
	Called by	TRAC				
TRBPOW						
	Calls	SATTMP				
	Called by	ITURB	TURB1			
TRBPRE						
	Calls	RDVLT	EVLTAB	RDFLT	TRIP	RDPTR
	Called by	PREP1D				
TRBPST						
	Calls	RDPTR	RDFLT	RDVLT		
	Called by	POST				
TRIP						
	Calls	ERROR				
	Called by	BREAKX	PUMPSR	TRBPRE	CORE1	RKIN
		VLVEX	EVFXXX	TIMSTP	WPUMP	FILLX

TRIPS						
	Calls	CBSET	SVSET	TRPSET	ERROR	
	Called by	PREP				
TRISLV						
	Called by	RODHT				
TRPSET						
	Calls	R2C	ERROR	UNCNVT	UNNUMB	
	Called by	TRIPS				
TURB1						
	Calls	PREPER	BKMOM	SAVBD	EVFXXX	SETBD
		TRBPOW				
	Called by	PREP1D				
TURB2						
	Calls	INNER				
	Called by	OUT1D				
TURB3						
	Calls	CONSTB	POSTER	SETBD	EVALDF	SAVBD
	Called by	POST				
UNCNVT						
	Calls	ERROR				
	Called by	CORE1	REECHO	WBREAK	ECOMP	REROD1
		WCOMP	EDIT	RHTSTR	WFILL	ELGR
		RPIPE	WHTSTR	HOUT	RPUMP	WLEVEL
		HTSTR1	RROD2	WMXYTB	IHPSS1	RTEE
		WPIPE	INPUT	RTURB	WPLEN	IROD
		RVLVE	WPRIZR	IVSSL	RVSSL	WPUMP
		NAMLST	SEDIT	WTEE	RCNTL	TIMSTP
		WTURB	RCOMP	TRPSET	WVLVE	READR
		WARRAY	WVSSL			
UNNUMB						
	Calls	ERROR				
	Called by	INPUT	REHTST	RROD2	RCNTL	REROD1
		TRPSET	RECNTL	RROD1		
UNSVCB						
	Calls	ERROR	R2C			

	Called by	RCNTL	RETEE	RTEE	RECNTL	REVLVE
		RVLVE	REHTST	RPIPE	SCLTBL	REPIPE
		RPUMP	WARRAY	REPUMP	RROD1	WTURB
		REROD1	RROD2			
VALUE						
	Calls	JVALUE				
	Called by	PREINP				
VELBC						
	Calls	BTESTC				
	Called by	TF3DS1				
VFWALL3						
	Calls	WDRAG				
	Called by	PREFWD				
VISCL						
	Calls	VISCLD	VISCLH			
	Called by	FPROPD	FPROPH			
VISCLD						
	Called by	VISCL				
VISCLH						
	Called by	VISCL				
VISCV						
	Calls	VISCVD	VISCVH			
	Called by	CORE1	FPROPH	HTVSSL	FPROPD	HTCOR
		HVWEBB				
VISCVD						
	Called by	VISCV				
VISCVH						
	Called by	VISCV				
VLVE1						
	Calls	BKMOM	SAVBD	VLVEX	PREPER	SETBD
	Called by	PREP1D				
VLVE2						
	Calls	INNER				
	Called by	OUT1D				

VLVE3						
	Calls	CONSTB	EVFXXX	SAVBD	EVALDF	POSTER
		SETBD				
	Called by	POST				
VLVEX						
	Calls	FAXPOS	ERROR	EVLTAB	SHIFTB	TRIP
	Called by	VLVE1				
VMCELL						
	Called by	INPUT				
VOLFA						
	Called by	IPIPE	IPUMP	ITURB	IPRIZR	ITEE
		IVLVE				
VOLV						
	Called by	PREPER				
VRBD						
	Called by	VSSL1				
VSSL1						
	Calls	CIF3	IFSET	DVPSCL	SETBDT	ERROR
		J3D	SETVA	FEMOMX	LININT0	FEMOMY
		MANAGE	FEMOMZ	TIMUPD	PREFWD	VRBD
	Called by	PREP3D				
VSSL2						
	Calls	BACIT	BAKUP	J3D	STDIR	BTESTC
		MANAGE	TF3DS	CELLA3	MATSOL	CHECKSIZE
		TF3DS1	TF3DS3	ERROR	THERMO	FLUXES
		SETBDT	VSSSSR	HTIF		
	Called by	OUT3D				
VSSL3						
	Calls	FF3D	MANAGE	BAKUP	FPROP	MIX3D
		BKSTB3	GVSSL2	BTESTC	STBME3	EVALDF
		J3D	THERMO			
	Called by	POST3D				
VSSROD						
	Calls	SQRT				
	Called by	FLTOM				
VSSSSR						
	Called by	VSSL2				

WARRAY						
	Calls	JUSTLR	SETLCM	UNSVCB	UNCNVT	WLABR
	Called by	ELGR	REPLEN	RPIPE	INPUT	REPUMP
		RPLEN	PUMPX	REROD1	RPUMP	RBREAK
		RETEE	RROD2	RCNTL	RETURB	RTEE
		RCOMP	REVLVE	RTURB	RDCRVS	REVSSL
		RVLVE	REBRK	RFILL	RVSSL	RECNTL
		RHTSTR	SCLTBL	REFILL	RLEVEL	WRCOMP
		REPIPE				
WBREAK						
	Calls	ECOMP	UNCNVT			
	Called by	WCOMP				
WCOMP						
	Calls	CWVSSL	RDPTR	WPLEN	RDVLT	WPRIZR
		UNCNVT	WPUMP	WBREAK	WRFLT	R2C
		WFILL	WTEE	R2C32	WHTSTR	WTURB
		RDFLT	WPIPE	WVLVE		
	Called by	EDIT				
WDRAG						
	Called by	VFWALL3				
WFILL						
	Calls	ECOMP	UNCNVT			
	Called by	WCOMP				
WHTSTR						
	Calls	GETTYPE	RRDLCM	R2C32	UNCNVT	MANAGE
		RDFLT	WRFLT			
	Called by	WCOMP				
WIARN						
	Calls	WLABIN	JUSTLR			
	Called by	RCOMP	REVSSL	RTURB	REPLEN	RHTSTR
		RVSSL	REROD1	RPLEN	WRCOMP	RETURB
WIR						
	Calls	ERROR				
	Called by	READR	REECHO			
WJCELL						
	Called by	ITEE	JBD4			
WLABI						
	Called by	INPUT	RCNTL			

WLABIN

	Called by	WIARN				

WLABR

	Called by	RROD2	WARRAY	WMXYTB		

WLEVEL

	Calls	LEVELI	UNCNVT			
	Called by	IVSSL	WVSSL			

WMXYTB

	Calls	JUSTLR	SETLCM	WLABR	UNCNVT	
	Called by	RCNTL	RETEE	RROD2	RECNTL	REVLVE
		RTEE	REPIPE	RPIPE	RVLVE	REPUMP
		RPUMP	SCLTBL	REROD1		

WPIPE

	Calls	ECOMP	UNCNVT			
	Called by	WCOMP				

WPLEN

	Calls	UNCNVT				
	Called by	WCOMP				

WPRIZR

	Calls	ECOMP	UNCNVT			
	Called by	WCOMP				

WPUMP

	Calls	ECOMP	TRIP	UNCNVT		
	Called by	WCOMP				

WRBRVLT

	Called by	WRVLT				

WRCOMP

	Calls	BTESTC	WARRAY	WIARN		
	Called by	REPIPE	REPUMP	RETURB	REPRZR	RETEE
		REVLVE				

WRFIVLT

	Called by	WRVLT				

WRFLT

	Called by	IBRK	ITURB	REBRK	IFILL	IVLVE
		REFILL	IPIPE	LCMTRN	RFILL	IPLEN

		POST	SCMLCM	IPRIZR	RBREAK	WCOMP
		IPUMP	RDCOMP	WHTSTR	ITEE	
WRPIVLT						
	Called by	WRVLT				
WRPLVLT						
	Called by	WRVLT				
WRPRVLT						
	Called by	WRVLT				
WRPTR						
	Called by	CIVSSL	REBRK	RFILL	LCMTRN	REFILL
		SCMLCM	RBREAK			
WRPUVLT						
	Called by	WRVLT				
WRRDVLT						
	Called by	WRVLT				
WRTBVLT						
	Called by	WRVLT				
WRTEVLT						
	Called by	WRVLT				
WRVAVLT						
	Called by	WRVLT				
WRVLT						
	Calls	ERROR	WRPLVLT	WRTBVLT	WRBRVLT	WRPRVLT
		WRTEVLT	WRFIVLT	WRPUVLT	WRVAVLT	WRPIVLT
		WRRDVLT	WRVSVLT			
	Called by	CIHTST	ITEE	REBRK	CIVSSL	ITURB
		REFILL	HTSTR1	IVLVE	REVSSL	HTSTR3
		LCMTRN	RFILL	IBRK	OUT1D	RHTSTR
		ICOMP	OUT3D	RPIPE	IFILL	POST
		RTEE	IPIPE	POST3D	RTURB	IPLEN
		PREP1D	RVSSL	IPRIZR	PREP3D	SCMLCM
		IPUMP	RBREAK			
WRVSVLT						
	Called by	WRVLT				
WTEE						
	Calls	ECOMP	UNCNVT			
	Called by	WCOMP				
WTURB						
	Calls	ECOMP	UNCNVT	UNSVCB		

	Called by	WCOMP				
WVLVE						
	Calls	ECOMP	UNCNVT			
	Called by	WCOMP				
WVSSL						
	Calls	UNCNVT	WLEVEL	MANAGE		
	Called by	CWVSSL				
XTV1D						
	Calls	GETTYPE	R2C32	XTVBUF		
	Called by	XTVPIPE	XTVPUMP	XTVVALV	XTVPRZR	XTVTEE
XTVBI3E						
	Calls	CXTVBW				
	Called by	XTVBUF				
XTVBUF						
	Calls	XTVBI3E				
	Called by	XTV1D	XTVHT	XTVSIG	XTVCB	XTVPLEN
		XTVVSL	XTVDR			
XTVCB						
	Calls	XTVBUF				
	Called by	XTVDR				
XTVDR						
	Calls	CXTVCL	XTVHT	XTVSIG	CXTVOA	XTVPIPE
		XTVTEE	RDFLT	XTVPLEN	XTVVALV	XTVBUF
		XTVPRZR	XTVVSL	XTVCB	XTVPUMP	
	Called by	INIT	STEADY	TRANS	PSTEPQ	
XTVHT						
	Calls	GETTYPE	R2C32	RRDLCM	MANAGE	RDFLT
		XTVBUF				
	Called by	XTVDR				
XTVINIT						
	Calls	CXTVIN	ERROR			
	Called by	INIT				
XTVPIPE						
	Calls	RDPTR	RDVLT	XTV1D		

	Called by	XTVDR				
XTVPLEN	Calls	GETTYPE	R2C32	RDVLT	RDPTR	XTVBUF
	Called by	XTVDR				
XTVPRZR	Calls	RDPTR	RDVLT	XTV1D		
	Called by	XTVDR				
XTVPUMP	Calls	RDPTR	RDVLT	XTV1D		
	Called by	XTVDR				
XTVSIG	Calls	XTVBUF				
	Called by	XTVDR				
XTVTEE	Calls	RDPTR	RDVLT	XTV1D		
	Called by	XTVDR				
XTVVALV	Calls	RDPTR	RDVLT	XTV1D		
	Called by	XTVDR				
XTVVSL	Calls	GETTYPE	R2C32	XTVBUF	RVSLCM	
	Called by	XTVDR				
ZCORE	Called by	CORE1				
ZEROV	Called by	TF3DS1				
ZPWHCI	Called by	CORE1	IROD			
ZPWNRM	Called by	CORE1	RROD2			
ZPWRCI	Calls	ERROR				
	Called by	CORE1	IROD			

APPENDIX C

DESCRIPTION OF TRAC-M COMPONENT COMMON-BLOCK VARIABLES

C.1. POINTER TABLES

The pointer tables for 1D components (described below) use four general sets of pointers from header files DUALPT.H, HYDROPT.H, INTPT.H, and HEATPT.H.

C.1.1. DUALPT.H. These pointer variables are declared to be INTEGER and refer to variables whose values are stored for both old and new-time values.

Name	Array	Dimension	Description
LALP	ALP	NCELLS	Old gas volume fraction.
LALPD	ALPD	0	Variable not currently implemented.
LALPDN	ALPDN	0	Variable not currently implemented.
LALPN	ALPN	NCELLS	New gas volume fraction.
LALV	ALV	NCELLS	Old value of the flashing interfacial heat-transfer coefficient (HTC) times interfacial area.
LALVE	ALVE	NCELLS	Old value of the liquid-side interfacial HTC times interfacial area.
LALVEN	ALVEN	NCELLS	New value of the liquid-side interfacial HTC times interfacial area.
LALVN	ALVN	NCELLS	New value of the flashing interfacial HTC times interfacial area.
LARA	ARA	NCELLS	Old stabilizer value for $\alpha\rho_a$.

LARAN	ARAN	NCELLS	New stabilizer value for $\alpha\rho_a$.
LAREL	AREL	NCELLS	Old stabilizer value for $(1-\alpha)\rho_\ell e_\ell$.
LARELN	ARELN	NCELLS	New stabilizer value for $(1-\alpha)\rho_\ell e_\ell$.
LAREV	AREV	NCELLS	Old stabilizer value for $\alpha\rho_v e_v$.
LAREVN	AREVN	NCELLS	New stabilizer value for $\alpha\rho_v e_v$.
LARL	ARL	NCELLS	Old stabilizer value for $(1-\alpha)\rho_\ell$.
LARLN	ARLN	NCELLS	New stabilizer value for $(1-\alpha)\rho_\ell$.
LARV	ARV	NCELLS	Old stabilizer value for $\alpha\rho_v$.
LARVN	ARVN	NCELLS	New stabilizer value for $\alpha\rho_v$.
LBIT	BIT	NCELLS+1	Bit flags from the previous timestep.
LBITN	BITN	NCELLS+1	Bit flags for the current timestep.
LCHTI	CHTI	NCELLS	Old value of the vapor-side interfacial HTC times the interfacial area.
LCHTIA	CHTIA	NCELLS	Old value of the noncondensable-gas interfacial HTC times the interfacial area.
LCHTAN	CHTAN	NCELLS	New value of the non-condensable-gas interfacial HTC times the interfacial area.
LCHTIN	CHTIN	NCELLS	New value of the vapor-side interfacial HTC times the interfacial area.
LCIF	CIF	NCELLS+1	Old interfacial-drag coefficients.

LCIFN	CIFN	NCELLS+1	New interfacial-drag coefficients.
LCONC	CONC	NCELLS *ISOLUT	Old solute mass to liquid mass ratio. ISOLUT = 0 or 1.
LCONCN	CONCN	NCELLS *ISOLUT	New solute mass to liquid mass ratio. ISOLUT = 0 or 1.
LD(3)	D	NCELLS	Variable not currently implemented.
LDN(3)	DN	NCELLS	Variable not currently implemented.
LEA	EA	NCELLS	Old noncondensable-gas internal energy.
LEAN	EAN	NCELLS	New noncondensable-gas internal energy.
LEL	EL	NCELLS	Old liquid internal energy.
LELN	ELN	NCELLS	New liquid internal energy.
LEV	EV	NCELLS	Old gas internal energy.
LEVN	EVN	NCELLS	New gas internal energy.
LGAM	GAM	NCELLS	Old vapor generation rate per unit volume.
LGAMN	GAMN	NCELLS	New vapor generation rate per unit volume.
LHIG	HIG	NCELLS	New HTC between inside wall and gas.
LHIGO	HIGO	NCELLS	Old HTC between inside wall and gas.
LHIL	HIL	NCELLS	New HTC between inside wall and liquid.
LHILO	HILO	NCELLS	Old HTC between inside wall and liquid.

LHIV	HIV	NCELLS	New HTC between inside wall and gas.
LHIVO	HIVO	NCELLS	Old HTC between inside wall and gas.
LP	P	NCELLS	Old total pressure.
LPA	PA	NCELLS	Old noncondensable-gas partial pressure.
LPAN	PAN	NCELLS	New noncondensable-gas partial pressure.
LPN	PN	NCELLS	New total pressure.
LQPPC	QPPC	NCELLS	New critical heat flux (CHF).
LQPPCO	QPPCO	NCELLS	Old CHF.
LROA	ROA	NCELLS	Old noncondensable-gas density.
LROAN	ROAN	NCELLS	New noncondensable-gas density.
LROL	ROL	NCELLS	Old liquid density.
LROLN	ROLN	NCELLS	New liquid density.
LROV	ROV	NCELLS	Old gas density.
LROVN	ROVN	NCELLS	New gas density.
LS	S	NCELLS *ISOLUT	Old solute mass plated on stricture surface. ISOLUT = 0 or 1.
LSN	SN	NCELLS *ISOLUT	New solute mass plated on structure surface. ISOLUT = 0 or 1.
LTCE	TCE	1	Old total convective energy.
LTCEN	TCEN	1	New total convective energy.
LTD	TD	0	Variable not currently implemented.

LTDN	TDN	0	Variable not currently implemented.
LTL	TL	NCELLS	Old liquid temperature.
LTLN	TLN	NCELLS	New liquid temperature.
LTV	TV	NCELLS	Old gas temperature.
LTVN	TVN	NCELLS	New gas temperature.
LTW	TW	NCELLS *NODES	Old wall temperature.
LTWA	TWA	1	Old absolute total conduction.
LTWAN	TWAN	1	New absolute total conduction.
LTWE	TWE	1	Old effective total conduction.
LTWEN	TWEN	1	New effective total conduction.
LTWN	TWN	NCELLS *NODES	New wall temperature.
LVL	VL	NCELLS+1	Old liquid velocity.
LVLN	VLN	NCELLS+1	New liquid velocity.
LVLT	VLT	NCELLS+1	New stabilizer liquid velocity $\left(\tilde{V}_{\ell}^{n+1}\right)$.
LVLTO	VLTO	NCELLS+1	Old stabilizer liquid velocity $\left(\tilde{V}_{\ell}^{n}\right)$.
LVM	VM	NCELLS+1	Old mixture velocity.
LVMN	VMN	NCELLS+1	New mixture velocity.
LVV	VV	NCELLS+1	Old gas velocity.
LVVN	VVN	NCELLS+1	New gas velocity.
LVVT	VVT	NCELLS+1	New stabilizer gas velocity $\left(\tilde{V}_{g}^{n+1}\right)$.

LVVTO	VVTO	NCELLS+1	Old stabilizer gas velocity $\left(\tilde{V}_g^n\right)$.

C.1.2. **HYDROPT.H.** These pointer variables are declared to be INTEGER and refer to variables associated with the hydrodynamic calculations.

Name	Array	Dimension	Description
LALPMN	ALPMN	NCELLS	Minimum value of the gas volume fraction among a cell and all its neighbors.
LALPMX	ALPMX	NCELLS	Maximum value of the gas volume fraction among a cell and all its neighbors.
LALPO	ALPO	NCELLS	Gas volume fraction at the start of the previous step (α^{n-1}).
LAM	AM	NCELLS	Noncondensable-gas mass.
LARC	ARC	NCELLS *ISOLUT	Density of solute in cell, $c(1-\alpha)\rho_\ell$. ISOLUT = 0 or 1.
LCFZ	CFZ	0	Variable not currently implemented.
LCL	CL	NCELLS	Liquid thermal conductivity.
LCPL	CPL	NCELLS	Liquid specific heat at constant pressure.
LCPV	CPV	NCELLS	Gas specific heat at constant pressure.
LCV	CV	NCELLS	Gas thermal conductivity.
LDALVA	DALVA	NCELLS	Variable not currently implemented.
LDELDP	DELDP	NCELLS+1	Derivative of the liquid internal energy with respect to pressure.

LDELDT	DELDT	NCELLS+1	Derivative of the liquid internal energy with respect to liquid temperature.
LDEVAP	DEVAP	NCELLS+1	Derivative of the noncondensable-gas internal energy with respect to noncondensable-gas pressure.
LDEVAT	DEVAT	NCELLS+1	Derivative of the noncondensable-gas internal energy with respect to gas temperature.
LDEVDP	DEVDP	NCELLS+1	Derivative of the gas internal energy with respect to pressure.
LDEVDT	DEVDT	NCELLS+1	Derivative of the gas internal energy with respect to gas temperature.
LDFLDP	DFLDP	NCELLS+1	Derivative of liquid velocity with respect to pressure.
LDFVDP	DFVDP	NCELLS+1	Derivative of gas velocity with respect to pressure.
LDHLSP	DHLSP	NCELLS+1	Derivative of liquid enthalpy with respect to pressure.
LDHVSP	DHVSP	NCELLS+1	Derivative of vapor enthalpy with respect to vapor pressure.
LDRIV	DR	19*(NCELLS+1)	Storage array for thermodynamic derivatives and enthalpies.
LDROLP	DROLP	NCELLS+1	Derivative of the liquid density with respect to pressure.
LDROLT	DROLT	NCELLS+1	Derivative of the liquid density with respect to liquid temperature.
LDROVP	DROVP	NCELLS+1	Derivative of the gas density with respect to pressure.

LDROVT	DROVT	NCELLS+1	Derivative of the gas density with respect to gas temperature.
LDRVAP	DRVAP	NCELLS+1	Derivative of the noncondensable-gas density with respect to noncondensable-gas pressure.
LDRVAT	DRVAT	NCELLS+1	Derivative of the non-condensable-gas density with respect to gas temperature.
LDTSDP	DTSDP	NCELLS+1	Derivative of the TSAT saturation temperature with respect to pressure.
LDTSSP	DTSSP	NCELLS+1	Derivative of the TSSN saturation temperature with respect to vapor pressure.
LDX	DX	NCELLS	Cell length.
LELEV	ELEV	NCELLS*IELV	Cell-centered elevations (used only if IELV = 1 in NAMELIST input).
LFA	FA	NCELLS+1	Cell-edge (interface) flow area.
LFAVOL	FAVOL	NCELLS	Cell flow area used in choked-flow model.
LFINAN	FINAN	NCELLS	Inverted annular regime factor.
LFRIC	FRIC	(NCELLS+1) *NFRC1	Additive friction factor.
LFSMLT	FSMLT	NCELLS	Interphasic-area multiplier during condensation.
LGRAV	GRAV	NCELLS+1	Gravitation term (cosine θ).
LGRVOL	GRAVOL	NCELLS	Cell-averaged gravitation term.
LH(1)	WFHF	NCELLS+1	Weighting factor for stratified-flow regime.

LH(2)	SI*DX	NCELLS+1	Stratified interfacial area.
LH(3)	DHLDZ	NCELLS+1	Gravitational head force caused by gas volume fraction gradient.
LHD	HD	(NCELLS+1) *(NDIA1–1)	Hydraulic diameter.
LHDHT	HDHT	(NCELLS+1) *(NDIA1–1)	Heat-transfer hydraulic diameter.
LHFG	HFG	NCELLS	Latent heat of vaporization.
LHGAM	HGAM	NCELLS	Energy contribution to phase change from subcooled boiling.
LHLA	HLA	NCELLS	Sum of all products of liquid HTC with heat-transfer area.
LHLATW	HLATW	NCELLS	Similar to HLA except that the product includes wall temperature.
LHLST	HLST	NCELLS+1	Liquid enthalpy at the TSSN saturation temperature and total pressure.
LHVA	HVA	NCELLS	Sum of all products of gas HTC with heat-transfer area.
LHVATW	HVATW	NCELLS	Similar to HVA except that the product includes wall temperature.
LHVST	HVST	NCELLS+1	Vapor enthalpy at the TSSN saturation temperature and vapor pressure.
LNF1SM	NF1SM	3*NFACES	Special purpose DOE-model parameter.
LNF2SM	NF2SM	3*NFACES	Special purpose DOE-model parameter.
LNF3SM	NF3SM	3*NFACES	Special purpose DOE-model parameter.

LNFCLSM	NFCLSM	NFACES	Special purpose DOE-model parameter.
LNFCVSM	NFCVSM	NFACES	Special purpose DOE-model parameter.
LNFL4SM	NFL4SM	3*NFACES	Special purpose DOE-model parameter.
LNFLSM	NFLSM	3*NFACES	Special purpose DOE-model parameter.
LNFV4SM	NFV4SM	3*NFACES	Special purpose DOE-model parameter.
LNFVSM	NFVSM	3*NFACES	Special purpose DOE-model parameter.
LQP3F	QP3F	NCELLS	QPPP spatial-distribution factor applied to the wall heat source.
LQPPP	QPPP	NODES* NCELLS	QPPP spatial-distribution factor applied to the wall heat source.
LQRL	QRL	NCELLS	Radiation heat flux to the liquid.
LQRV	QRV	NCELLS	Radiation heat flux to the gas.
LR0SM	R0SM	3*NFACES	Special purpose DOE-model parameter.
LRARL	RARL	0	Variable not currently implemented.
LRARV	RARV	0	Variable not currently implemented.
LREGNM	REGNM	NCELLS+1	Flow-regime number.
LRHS	RHS	NCELLS	Implicit vs explicit weighting factor, g'.
LRMEM	RMEM	0	Variable not currently implemented.

LRMVM	RMVM	NCELLS+1	Mixture density times mixture velocity.
LROM	ROM	NCELLS	Mixture density.
LRSM	RSM	3*NFACES	Special purpose DOE-model parameter.
LRVMF	RVMF	NCELLS+1	Gas mass flow.
LSIG	SIG	NCELLS	Surface tension.
LTRID	TRID	6*(NCELLS+1)	Storage for stabilizer linear system.
LTSAT	TSAT	NCELLS	Saturation temperature at total pressure.
LTSSN	TSSN	NCELLS	Saturation temperature at steam (vapor) pressure.
LUVSM	UVSM	3*NFACES	Special purpose DOE-model parameter.
LVISL	VISL	NCELLS	Liquid viscosity.
LVISV	VISV	NCELLS	Gas viscosity.
LVLALP	VLALP	NCELLS	Liquid mass flux that enters the cell from the cell edges located above the cell.
LVLSM	VLSM	3*NFACES	Special purpose DOE-model parameter.
LVLVC	VLVC	NCELLS	Liquid velocity at a neighboring cell edge where the donor-celled liquid fraction is maximum.
LVLVOL	VLVOL	NCELLS	Choked-flow-model cell liquid velocity.
LVLX	VLX	0	Variable not currently implemented.
LVOL	VOL	NCELLS	Cell volume.

LVR	VR	NCELLS+1	Relative (gas – liquid) velocity.
LVRV	VRV	NCELLS	Cell-averaged relative velocity.
LVVSM	VVSM	3*NFACES	Special purpose DOE-model parameter.
LVVVOL	VVVOL	NCELLS	Choked-flow-model cell gas velocity.
LVVX	VVX	0	Variable not currently implemented.
LWA	WA	NCELLS	Wall area.
LWAT	WAT	NCELLS	Total heat-transfer area.
LWFL	WFL	NCELLS+1	Wall friction factor for the liquid.
LWFMFL	WFMFL	NCELLS+1	Wall-friction multiplier factor for the liquid.
LWFMFV	WFMFV	NCELLS+1	Wall-friction multiplier factor for the gas.
LWFV	WFV	NCELLS+1	Wall friction factor for the gas.
LXSM	XSM	NCELLS	Special purpose DOE-model parameter.
LYSM	YSM	NCELLS	Special purpose DOE-model parameter.
LZSM	ZSM	NCELLS	Special purpose DOE-model parameter.

C.1.3. INTPT.H. These pointer variables are declared to be INTEGER.

Name	Array	Dimension	Description
LIDR	IDR	NCELLS	Heat-transfer regime.
LLCCFL	LCCFL	NCELLS+1	Counter-Current Flow Limitation (CCFL) flag.

LMATID	MATID	NODES-1	Structural material identification.
LNFF	NFF	NCELLS+1	Friction-correlation option.

C.1.4. HEATPT.H. These pointer variables are declared to be INTEGER and refer to variables associated with the embedded wall heat-transfer calculations.

Name	**Array**	**Dimension**	**Description**
LCPW	CPW	(NODES–1) *NCELLS	Specific heat of wall.
LCW	CW	(NODES–1) *NCELLS	Wall thermal conductivity.
LDR	DR	NODES–1	Radial mesh size.
LEMIS	EMIS	NCELLS	Wall-surface emissivity.
LHOL	HOL	NCELLS	HTC between wall and outside liquid.
LHOV	HOV	NCELLS	HTC between wall and outside gas.
LRN	RN	NODES	Radii at nodes.
LRN2	RN2	NODES–1	Radii at node centers.
LROW	ROW	(NODES–1) *NCELLS	Wall density.
LTCHF	TCHF	NCELLS	CHF temperature.
LTOL	TOL	NCELLS	Liquid temperature outside wall.
LTOV	TOV	NCELLS	Gas temperature outside wall.

C.2. BREAK COMPONENT

C.2.1. BREAKVLT.H—BREAK Specific Component Table with Common Block breakCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
ALPOFF	ALPOFFIND=2	Gas volume fraction when the trip is OFF after it was ON.
BSA	BSAIND=3	Time-integrated noncondensable-gas mass from the BREAK.
BSMASS	BSMASSIND=4	Time-integrated mass flow from the BREAK.
BXA	BXAIND=5	Noncondensable-gas mass flow from the BREAK.
BXMASS	BXMASSIND=6	Current mass flow from the BREAK.
CONOFF	CONOFFIND=7	Ratio of solute mass to liquid mass when the trip is OFF after it was ON.
DELTL	DELTLIND=8	Liquid temperature offset from the saturation temperature.
DELTV	DELTVIND=9	Gas temperature offset from the saturation temperature.
PAOFF	PAOFFIND=10	Noncondensable-gas partial pressure when the trip is OFF after it was ON.
POFF	POFFIND=11	Total pressure when the trip is OFF after it was ON.
POFFS	POFFSIND=12	Saved value of total pressure when the trip is OFF after it was ON that has not been adjusted by a CSS type 5 controller.

RBMX	RBMXIND=13	Maximum rate of change of total pressure at the BREAK.
TIN	TININD=14	Fluid temperature at the BREAK.
TLOFF	TLOFFIND=15	Liquid temperature when the trip is OFF after it was ON.
TVOFF	TVOFFIND=16	Gas temperature when the trip is OFF after it was ON.
TYPE1	TYPE1IND=18	Type of adjacent component at JUN1.
Z11111	Z11111IND=17	Dummy variable that provides a known end to the COMMON block's real-valued parameters.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	**Parameter Constant**	**Description**
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
IBASV	IBASVIND=–2	Signal-variable or control-block ID number defining the gas volume fraction in the BREAK cell for the IBTY = 6 option.
IBCNSV	IBCNSVIND=–3	Signal-variable or control-block ID number defining the ratio of the solute mass to the liquid mass in the BREAK cell for the IBTY = 6 option.
IBF	IBFIND=–4	Last interpolated interval in the rate-factor table.
IBP	IBPIND=–5	Last interpolated interval in the BREAK composition parameter tables.
IBPASV	IBPASVIND=–6	Signal-variable or control-block ID number defining the noncondensable-gas partial pressure in the BREAK cell for the IBTY = 6 option.

IBPSV	IBPSVIND=–7	Signal-variable or control-block ID number defining the total pressure in the BREAK cell for the IBTY = 6 option.
IBSV	IBSVIND=–8	Signal-variable or control-block ID number defining the BREAK-table abscissa-coordinate variable.
IBTLSV	IBTLSVIND=–9	Signal-variable or control-block ID number defining the liquid temperature in the BREAK cell for the IBTY = 6 option.
IBTR	IBTRIND=–10	Trip ID number that controls evaluation of the BREAK tables.
IBTVSV	IBTVSVIND=–11	Signal-variable or control-block ID number defining the gas temperature in the BREAK cell for the IBTY = 6 option
IBTY	IBTYIND=–12	BREAK-table input option.
ICJ	ICJIND=–13	Iteration index of adjacent component.
INEXTI	INEXTIIND=–14	Variable no longer used.
IOFF	IOFFIND=–15	Fluid-state option when the trip is OFF after it was ON.
IONOFF	IONOFFIND=–16	Number of timesteps the trip is ON.
ISAT	ISATIND=–17	BREAK-table use option.
JS1	JS1IND=–18	Junction sequence number.
JUN1	JUN1IND=–19	Junction number for connection to the BREAK.
NBRF	NBRFIND=–20	Number of data pairs in the rate-factor table.
NBSV	NBSVIND=–21	Signal-variable or control-block ID number defining the rate-factor table's abscissa-coordinate variable.
NBTB	NBTBIND=–22	Number of data pairs in the BREAK table.

ZI1111	ZI1111IND=–23	Dummy variable that provides a known end to the COMMON block.

C.2.2. BREAKPT.H— BREAK Pointer Table. These pointer variables are declared to be INTEGER. For BREAKs, NCELLS = 1.

Name	Array	Dimension	Description
LALPTB	ALPTB	NBTB*2	Gas volume fraction table.
LCONTB	CONTB	NBTB*2	Ratio of solute mass to liquid mass table.
LPATB	PTAB	NBTB*2	Noncondensable-gas partial pressure table.
LPTB	PTB	NBTB*2	Total pressure table.
LRFTB	RFTB	NBRF*2	Rate-factor table.
LTLTB	TLTB	NBTB*2	Liquid temperature table.
LTVTB	TVTB	NBTB*2	Gas temperature table.

C.3. FILL COMPONENT

C.3.1. FILLVLT.H—FILL Specific Component Table with Common Block fillCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
ALPOFF	ALPOFFIND=2	Gas volume fraction when the trip is OFF after it was ON.
CONOFF	CONOFFIND=3	Ratio of solute mass to liquid mass when the trip is OFF after it was ON.

FLOWIN	FLOWININD=4	Initial fluid mass flow to or from adjacent component.
FLWOFF	FLWOFFIND=5	Fluid mass flow when the trip is OFF after it was ON.
FSMASS	FSMASSIND=6	Time-integrated fluid mass flow out of the FILL.
FXMASS	FXMASSIND=7	Current fluid mass-flow rate out of the FILL.
PAOFF	PAOFFIND=8	Noncondensable-gas partial pressure when the trip is OFF after it was ON.
POFF	POFFIND=9	Total pressure when the trip is OFF after it was ON.
RFMX	RFMXIND=10	Maximum rate of change of FILL velocity or mass flow.
TLOFF	TLOFFIND=11	Liquid temperature when the trip is OFF after it was ON.
TVOFF	TVOFFIND=12	Gas temperature when the trip is OFF after it was ON.
TWTOLD	TWTOLD=13	Fraction of a previous FILL fluid dynamic-state parameter that is averaged with the FILL table's defined parameter to define the FILL parameter value for this timestep (0.0 ≤ TWTOLD ≤ 1.0).
TYPE1	TYPE1IND=17	Type of adjacent component at JUN2.
VLOFF	VLOFFIND=14	Liquid velocity when the trip is OFF after it was ON.
VVOFF	VVOFFIND=15	Gas velocity when the trip is OFF after it was ON.
Z11111	Z11111IND=16	Dummy variable that provides a known end to the COMMON block.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
ICJ	ICJIND=–2	Iteration index of adjacent component.
IFASV	IFASVIND=–14	Signal-variable or control-block ID number defining the gas volume fraction in the FILL cell for the IFTY = 10 option.
IFCNSV	IFCNSVIND=–17	Signal-variable or control-block ID number defining the ratio of solute mass to liquid mass in the FILL cell for the IFTY = 10 option.
IFF	IFFIND=–3	Last interpolated interval in the rate-factor table.
IFMLSV	IFMLSV=–10	Signal-variable or control-block ID number defining liquid mass flow in the FILL cell for the IFTY = 10 option.
IFMVSV	IFMVSV=–11	Signal-variable or control-block ID number defining gas mass flow in the FILL cell for the IFTY = 10 option.
IFP	IFPIND=–4	Last interpolated interval in the FILL table.
IFPASV	IFPASVIND=–16	ID number of the signal variable or control block defining the noncondensable-gas partial pressure in the FILL cell for the IFTY = 10 option.
IFPSV	IFPSVIND=–15	ID number of the signal variable or control block defining the total pressure in the FILL cell for the IFTY = 10 option.
IFSV	IFSVIND=–5	The signal-variable ID number, which defines the FILL table's independent variable.

IFTLSV	IFTLSVIND=–12	ID number of the signal variable or control block defining the liquid temperature in the FILL cell for the IFTY = 10 option.
IFTR	IFTRIND=–6	FILL trip ID number.
IFTVSV	IFTVSVIND=–13	ID number of the signal variable or control block defining the vapor temperature in the FILL cell for the IFTY = 10 option.
IFTY	IFTYIND=–7	FILL-type option.
INEXTI	INEXTIIND=–8	Variable no longer used.
IOFF	IOFFIND=–9	FILL fluid-state option when the trip is OFF after it was ON.
IONOFF	IONOFFIND=–18	The number of timesteps the trip has been ON.
JS1	JS1IND=–19	Junction sequence number at JUN1.
JUN1	JUN1IND=–20	Junction number where the FILL is located.
NFRF	NFRFIND=–21	Number of rate-factor table data pairs whose rate factor is applied to the FILL table's independent variable.
NFSV	NFSVIND=–22	Signal-variable or control-block ID number defining the rate-factor table's abscissa-coordinate variable.
NFTB	NFTBIND=–23	Number of data pairs in the FILL table.
ZI1111	ZI1111IND=–24	Dummy variable that provides a known end to the COMMON block.

C.3.2. FILLPT.H—FILL Pointer Table. These pointer variables are declared to be INTEGER. For FILLS, NCELLS = 1.

Name	Array	Dimension	Description
LALPTB	ALPTB	|NFTB|*2	Gas volume fraction table.
LCONTB	CONTB	|NFTB|*2	Ratio of solute mass to liquid mass table.

LPATB	PATB	|NFTB|*2	Noncondensable-gas partial pressure table.
LPTB	PTB	|NFTB|*2	Total pressure table.
LRFTB	RFTB	|NFTB|*2	FILL rate-factor table.
LTLTB	TLTB	|NFTB|*2	Liquid temperature table.
LTVTB	TVTB	|NFTB|*2	Gas temperature table.
LVMTB	VMTB	|NFTB|*2	Liquid velocity table.
LVVTB	VVTB	|NFTB|*2	Gas velocity table.

C.4. HEAT-STRUCTURE COMPONENT

C.4.1. RODVLT.H—Heat-Structure ROD or SLAB Specific Component Table with Common Block rodCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
AMH2	AHM2IND=2	Hydrogen mass generated from metal-water reaction.
BCR0	BCR0IND=3	Zero-order coefficient of the first-order polynomial that defines the effective core-averaged concentration of control-rod pin boron.
BCR1	BCR1IND=4	First-order coefficient of the first-order polynomial that defines the effective core-averaged concentration of control-rod pin boron.
BEFF	BEFFIND=5	Total delayed-neutron fraction.

BPP0	BPP0IND=6	Zero-order coefficient of the first-order polynomial that defines the effective core-averaged concentration of burnable-poison pin boron.
BPP1	BPP1IND=7	First-order coefficient of the first-order polynomial that defines the effective core-averaged concentration of burnable-poison pin boron.
DRFB	DRFBIND=8	Reactivity-feedback change in K_{eff} over the last timestep.
DRI	DRIIND=9	Estimated change in power or reactivity over the previous timestep.
DRIO	DRIOIND=10	Old value of DRI; the old value of the power or reactivity-estimated correction.
DTNHT(1) DTNHT(2)	DTNHTIND=11 at 12	Delta temperature minimums used in the reflood calculation.
DTPK	DTPKIND=13	Kaganove-method integration timestep for solving the point-reactor kinetics equations.
DTXHT(1) DTXHT(2)	DTXHTIND=14 at 15	Delta temperature maximums used in the reflood calculation.
DZNHT	DZNHTIND=16	Delta Z_{min}.
ENEFF	ENEFFIND=17	Total decay-heat fraction.
EXTSOU	EXTSOUIND=18	Thermal power produced by external source neutrons in the reactor core.
FSI	FSIIND=19	Inner-surface area (or HTC) adjustment factor from a CSS type 5 controller.
FSO	FSOIND=20	Outer-surface area (or HTC) adjustment factor from a CSS type 5 controller.
FTCI	FTCIIND=21	Inner-surface node thermal-conductivity adjustment factor from a CSS type 5 controller.

FTCM	FTCMIND=22	Internal-nodes thermal-conductivity adjustment factor from a CSS type 5 controller.
FTCO	FTCOIND=23	Outer-surface node thermal-conductivity adjustment factor from a CSS type 5 controller.
FUCRAC	FUCRACIND=24	Fraction of uncracked fuel.
HDRI	HDRIIND=25	Thermal diameter (m) for the inner surface of the heat-structure ROD or SLAB element (used only when NAMELIST variable ITHD = 1).
HDRO	HDROIND=26	Thermal diameter (m) for the outer surface of the heat-structure ROD or SLAB element (used only when NAMELIST variable ITHD = 1).
HGAPO	HGAPOIND=27	Rod gap-conductance coefficient (for MATRD = 3).
HLI	HLIIND=28	Constant liquid heat-transfer coefficient at the inner surface (used when the inner-surface boundary condition flag IDBCI = 1, indicating constant HTCs and external temperatures).
HLO	HLOIND=29	Constant liquid heat-transfer coefficient at the outer surface (used when the outer-surface boundary condition flag IDBCO = 1, indicating constant HTCs and external temperatures).
HVI	HVIIND=30	Constant gas heat-transfer coefficient at the inner surface (used when the inner-surface boundary condition flag IDBCI = 1, indicating constant HTCs and external temperatures).
HVO	HVOIND=31	Constant gas heat-transfer coefficient at the outer surface (used when the outer-surface boundary condition flag IDBCO = 1, indicating constant HTCs and external temperatures).

PDRAT	PDRATIND=32	Rod pitch-to-diameter ratio.
PLDR	PLDRIND=33	Pellet dish radius. 0.0 = no pellet dish calculation; 1.0 = pellet dish calculation.
POWEXP	POWEXPIND=34	Exponent value to which the power distribution is raised to define the weighting function for averaging the reactivity-feedback parameters over the reactor-core volume.
QRDTOT	QRDTOTIND=35	Total rod heat flux.
REAC	REACIND=36	Reactivity feedback at the beginning of the previous timestep.
REACN	REACNIND=37	Reactivity-feedback estimate at the end of the present timestep.
REACT	REACTIND=38	Total reactivity at the beginning of the present timestep.
RMCK	RMCKIND=39	Reactor multiplication constant at the beginning of the present timestep.
RMCKN	RMCKNIND=40	Reactor multiplication constant estimate at the end of the present timestep.
RPOWER	RPOWERIND=41	Average reactor-core power over the timestep.
RPOWPF	RPOWPFIND=42	Prompt-fission power.
RPOWR	RPOWRIND=43	Beginning-of-timestep reactor-core power.
RPOWRI	RPOWRIIND=44	Initial reactor-core power.
RPOWRN	RPOWRNIND=45	End-of-timestep reactor-core power.
RPOWRO	RPOWROIND=46	End-of-timestep reactor-core power of the previous timestep.
RPOWTO	RPOWTOIND=47	Beginning-of-timestep reactor-core power of the previous timestep.

RPWOFF	RPWOFFIND=48	Programmed reactivity or reactor-core power when the controlling trip is OFF after it was ON.
RPWSCL	RPWSCLIND=49	Reactivity-power-table scale factor for programmed reactivity or reactor-core power.
RRPWMX	RRPWMXIND=50	Maximum rate of change of programmed reactivity or reactor-core power.
RZPWMX	RZPWMXIND=51	Maximum rate of change of the axial power shape.
SA(1) SA(2)	SAIND=52 at 53	Values of the inner- and outer-surface areas (or HTCs) adjusted by a CSS type 5 controller.
SAF	SAFIND=54	Adjustment factor evaluated by a CSS type 5 controller.
SDT	SDTIND=55	Time interval/s since the last reactivity change printout.
SHELV	SHELVIND=56	Axial elevation of the first (bottom) node row.
SHTD	SHTDIND=57	Numerical sign of the heat-transfer direction.
STIMET	STIMETIND=58	Problem time at which the last reactivity change was summed to variable storage for later printout.
TK(1) TK(2) TK(3)	TKIND=59 at 60 at 61	Values of the inner-node, internal-nodes, and outer-node thermal conductivity adjusted by a CSS type 5 controller.
TLI	TLIIND=62	Constant liquid temperature at the inner surface (used when the inner-surface boundary condition flag IDBCI = 1, indicating constant HTCs and external temperatures).

TLO	TLOIND=63	Constant liquid temperature at the outer surface (used when the outer-surface boundary condition flag IDBCO = 1, indicating constant HTCs and external temperatures).
TNEUT	TNEUTIND=64	Neutron generation time.
TPOWI	TPOWIIND=65	Total power across the inner surface of the HTSTR.
TPOWO	TPOWOIND=66	Total power across the outer surface of the HTSTR.
TRAMAX	TRAMAXIND=67	Average-rod peak-cladding temperature.
TRHMAX	TRHMAXIND=68	Maximum supplemental rod temperature.
TVI	TVIIND=69	Constant gas temperature at the inner surface (used when the inner-surface boundary condition flag IDBCI = 1, indicating constant HTCs and external temperatures).
TVO	TVOIND=70	Constant gas temperature at the outer surface (used when the outer-surface boundary condition flag IDBCO = 1, indicating constant HTCs and external temperatures).
WATLEV	WATLEVIND=71	Variable not used.
WIDTH	WIDTHIND=72	Width of the SLAB surface (used to compute surface area).
ZPWIN	ZPWININD=73	Axial-power-shape table's abscissa-coordinate variable value corresponding to the initial axial-power shape.
ZPWOFF	ZPWOFFIND=74	Axial-power-shape table's abscissa-coordinate variable value that corresponds to the axial-power shape that is used when the controlling trip is OFF after it was ON.
ZLPBOT	ZLPBOTIND=78	Axial location (m) of the bottom of the lower hot patch.

ZLPTOP	ZLPTOPIND=77	Axial location (m) of the top of the lower hot patch.
ZUPBOT	ZUPBOTIND=76	Axial location (m) of the bottom of the upper hot patch.
ZUPTOP	ZUPTOPIND=75	Axial location (m) of the top of the upper hot patch.
Z11111	Z11111IND=79	Dummy variable that provides a known end to the COMMON block of real-valued variables.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
IAF	IAFIND=–2	NAPCSS value of a CSS controller type 5.
IAXCND	IAXCNDIND=–3	Axial conduction indicator. 0 = no axial heat-transfer conduction calculated; 1 = axial heat-transfer conduction calculated in the heat-structure ROD or SLAB element.
IBU(1) IBU(2) IBU(3) IBU(4)	IBUIND=–4 at –5 at –6 at –7	Boron-unit flag for the Jth reactivity coefficient.
IDBCI	IDBCIIND=–8	Boundary condition option for the inner surface of the heat-structure ROD or SLAB element. 0 = adiabatic boundary condition; 1 = constant HTCs and external temperatures; 2 = coupled to specified cells in one or more hydro components.

IDBCO	IDBCOIND=–9	Boundary condition option for the outer surface of the heat-structure ROD or SLAB element. 0 = adiabatic boundary condition; 1 = constant HTCs and external temperatures; 2 = coupled to specified cells in one or more hydro components.
IEXT	IEXTIND=–10	Specifies if this HTSTR input was generated by the post processor EXTRACT. 0 = no; 1 = yes.
IONOFF	IONOFFIND=–11	Number of timesteps that the reactivity-power table's controlling trip has been ON.
IPATCH	IPATCHIND=–12	Hot-patch modeling indicator. 0 = no modeling; 1 = modeling of hot patches.
IPWDEP	IPWDEPIND=–13	Power-shape table-dependence option. –1 = power-shape-table power is defined by a signal variable or control block for each node with no normalization; 0 = power-shape-table independent variable is defined by signal-variable or control-block ID number IZPWSV; 1 = power-shape-table power is defined by a signal variable or control block for each node with normalization.
IPWRAD	IPWRADIND=–14	Spatial power-shape option. 0 = 1-D axial power-shape table; 1 = 2-D axial-r or axial-x power-shape table.
IRC(1) IRC(2) IRC(3) IRC(4)	IRCIND=–15 at –16 at –17 at –18	Number data values that defines the fuel-temperature (1) coolant-temperature, (2) gas-volume-fraction, (3) and solute-mass-concentration, and (4) reactivity-coefficient tables.

IRCJFM(1) IRCJFM(2) IRCJFM(3) IRCJFM(4)	IRCJFMIND=–19 at –20 at –21 at –22	Form number of reactivity coefficient for the argument number reactivity-feedback parameter.
IRCJTB(1,1) IRCJTB(2,1) IRCJTB(3,1) . . . . IRCJTB(4,4)	IRCJTBIND=–23 at –24 at –25 at –38	Number of first argument reactivity-feedback parameter value entries for the second argument reactivity-coefficient table.
IRF	IRFIND=–39	Last interpolated interval number in rate-factor table for the reactivity-power table.
IRFTR	IRFTRIND=–40	Trip ID number that controls the axial fine-mesh calculation.
IRFTR2	IRFTR2IND=–41	Trip ID number for evaluating the core reflood model when Namelist variable NEWRFD = 1.
IRP	IRPIND=–42	Last interpolated interval number in the reactivity-power table.
IRPWSV	IRPWSVIND=–43	Signal-variable or control-block ID number defining the reactivity-power table's abscissa-coordinate variable.
IRPWTR	IRPWTRIND=–44	Trip ID number that controls evaluation of the reactivity-power table.
IRPWTY	IRPWTYIND=–45	Neutronic point-reactor kinetics or reactor-core power option. 1 = point-reactor kinetics with constant prog. reactivity; 2 = point-reactor kinetics with table defined prog. reactivity; 3 = point-reactor kinetics with trip-initiated constant prog. reactivity; 4 = point-reactor kinetics with initial constant programmed reactivity and trip-initiated table defined prog. reactivity; 5 = constant reactor-core power; 6 = table defined reactor-core power;

		7 = initial constant reactor-core power with trip-initiated table defined reactor-core power. Add 10 to the above values to evaluate reactivity feedback.
ISNOTB	ISNOTBIND=–46	A flag variable that is defined if the solute is boron for the reactivity-feedback calculation. 0 = solute is boron; 1 = solute is not boron.
ITTCS	ITTCSIND=–47	Saved value of ITTC, the specification of an external thermocouple on the ROD- or SLAB-element surface.
IZF	IZFIND=–48	Last interpolated interval number in the rate-factor table for the axial power-shape table.
IZP	IZPIND=–49	Last interpolated interval number in the axial power-shape table.
IZPWSV	IZPWSVIND=–50	Signal-variable or control-block ID number defining the axial power-shape table's abscissa-coordinate variable.
IZPWTR	IZPWTRIND=–51	Trip ID number that controls evaluation of the axial power-shape table.
LENRD	LENRDIND=–52	Length of rod data.
LFVNR	LFVNRIND=–53	Relative position of new fundamental variables of rod data.
LFVNR1	LFVNR1IND=–54	Relative position of new heat-transfer data.
LFVR	LFVRIND=–55	Relative position of old fundamental variables of rod data.
LFVR1	LFVR1IND=–56	Relative position of old heat-transfer data.
LIQLEV	LIQLEVIND=–57	Specification of liquid level. 0 = no liquid level calculated on ROD or SLAB surface;

		1 = liquid level tracked on ROD or SLAB surface (this smooths the heat-transfer solution).
LNDRD	LNDRDIND=–58	Offset for double-sided HTSTRs. If the HTSTR is connected to hydro components on one side, then LNDRD = 0. If the HTSTR is connected to a hydro component on both the inside and outside surfaces, then LNDRD = the offset for the inside surface heat-transfer parameters.
LNFVR	LNFVRIND=–59	Length of fundamental variables of rod data.
LNFVR1	LNFVR1IND=–60	Length of heat-transfer data.
LNPTRR	LNPTRRIND=–61	Number of pointers of rod data.
LOCROD	LOCRODIND=–62	Pointer for beginning of rod data.
M1D	M1DIND=–63	Multiple 1D hydraulic-component coupling option.
M1DT	M1DTIND=–64	Type of HTSTR for the purpose of a neutronics calculation. 0 = Not part of a neutronics calculation. 1 = First HTSTR coupled to a neutronics calculation. 2 = Between the first and last HTSTR coupled to a neutronics calculation. 3 = Last HTSTR coupled to a neutronics calculation that evaluates the point-reactor kinetics calculation for all the coupled HTSTRs.
NCRX	NCRXIND=–65	Number of average ROD or SLAB elements that affect fluid dynamics.
NCRZ	NCRZIND=–66	Number of (course) axial intervals between temperature node rows.
NDG	NDGIND=–67	Input-specified number of delayed-neutron groups.
NDGX	NDGXIND=–68	Number of delayed-neutron groups.

NDH	NDHIND=–69	Input specified number of decay-heat groups.
NDHX	NDHXIND=–70	Number of decay-heat groups.
NFBPWT	NFBPWTIND=–71	Flag that defines the spatial distribution used to weight the averaging of the reactivity-feedback parameters over the reactor-core volume.
NFCI	NFCIIND=–72	FCI flag. 0 = no calculation; 1 = calculation.
NFCIL	NFCILIND=–73	Limit on FCI calculations per timestep.
NFUEL	NFUELIND=–74	Number of nodes in fuel pellet.
NHIST	NHISTIND=–75	Number of data pairs in the power-history table.
NINT	NINTIND=–76	Maximum possible number of interfaces between dissimilar materials in ROD or SLAB elements.
NMWRX	NMWRXIND=–77	Metal-water reaction flag. 0 = no calculation; 1 = calculation.
NONOFF	NONOFFIND=–78	Number of timesteps that the trip-controlling evaluation of the axial power-shape table has been ON.
NOPOWR	NOPOWRIND=–79	Specification of whether a power source is present in the heat-structure ROD or SLAB element. 0 = power source present in the ROD or SLAB; 1 = no power source present in the ROD or SLAB.
NRAMAX	NRAMAXIND=–80	Location of average-rod peak-cladding temperature used in the reflood calculation.

NRFD	NRFDIND=–81	Reflood flag. 0 = takes no action; 1 = turns on axial fine-mesh flag if it is off.
NRHMAX	NRHMAXIND=–82	Location of the supplemental-rod peak-cladding temperature.
NRIDR	NRIDRIND=–83	Specification of the hydro-cell location that is coupled to the inner and/or outer surfaces of the heat-structure ROD or SLAB element. 0 = the IDROD array is input for only the supplemental RODs or SLABs; 1 = the IDROD array is input for all RODs or SLABs; 2 = the IDROD array is input for all RODs or SLABs for both surfaces of the HTSTR.
NRODS	NRODSIND=–84	Number of computational (average plus supplemental) rods including "hot" rods. See NCRX.
NRPWI	NRPWIIND=–85	Radial- or thickness-direction power-shape integration option. –1 = histogram with step changes at the r or x locations; 0 = histogram with step changes midway between the r or x locations; 1 = trapezoidal integration.
NRPWR	NRPWRIND=–86	Number of radial or thickness locations that define the 2D axial-r or axial-x power shape.
NRPWRF	NRPWRFIND=–87	Number of rate-factor table data pairs whose rate factor is applied to the power or reactivity table's independent variable.
NRPWSV	NRPWSVIND=–88	Signal-variable or control-block ID number defining the reactivity-power rate-factor table's abscissa-coordinate variable.
NRPWTB	NRPWTBIND=–89	Number of data pairs in the reactivity-power table.

NRTS	NRTSIND=–90	Number of timesteps over which programmed reactivity and reactivity-feedback changes are summed for printout.
NSET	NSETIND=–91	Absolute value of the reflood axial fine-mesh trip set-status number during the previous timestep.
NSET2	NSET2IND=–92	Absolute value of the core-reflood trip set-status number.
NZMAX	NZMAXIND=–93	Maximum number of rows of heat-transfer nodes used in reflood calculation.
NZPWRF	NZPWRFIND=–94	Number of data pairs in the axial power-shape rate-factor table.
NZPWSV	NZPWSVIND=–95	Signal-variable or control-block ID number defining the axial power-shape rate-factor table's abscissa-coordinate variable.
NZPWTB	NZPWTBIND=–96	Number of axial power shapes in the axial power-shape table.
NZPWI	NZPWIIND=–97	Axial power shape integration option for the heat-transfer calculation. –1 = histogram with step changes at the axial locations; 0 = histogram with step changes midway between the axial locations; 1 = trapezoidal integration.
NZPWZ	NZPWZIND=–98	Number of axial locations defining the axial-power shape.
NZZNHC	NZZNHCIND=–99	Number of hydro-cell axial-direction channels that this powered HTSTR is coupled to.
ZI1111	ZI1111IND=–100	Dummy variable that provides a known end to the COMMON block.

C.4.2. RODPT.H— Heat-Structure Pointer Table. These pointer variables are declared to be INTEGER.

GENERAL ROD-DATA POINTERS:

Name	Array	Dimension	Description
LBETA	BETA	NDGX	Delayed-neutron group fraction.
LCDG	CDG	NDGX	Old concentration of delayed-neutron group.
LCDGN	CDGN	NDGX	New concentration of delayed-neutron group.
LCDH	CDH	NDHX	Old concentration of decay-heat group.
LCDHN	CDHN	NDHX	New concentration of decay-heat group.
LCLEN	CLEN	NCRX	Old total cladding length.
LCLENN	CLENN	NCRX	New total cladding length.
LCPOWR	CPOWR	NCRX	Relative power per average rod.
LEDH	EDH	NDHX	Energy-yield fraction of decay-heat group.
LFPUO2	FPUO2	NCRX	Fraction of plutonium oxide in mixed-oxide fuel fraction.
LFTD	FTD	NCRX	Fuel density (fraction of theoretical).
LGMIX	GMIX	NCRX*7	Mole fraction of gap-gas constituent.
LGMLES	GMLES	NCRX	Moles of gap gas.
LGRAVR	GRAVR	NCRZ	Cosine of the angle between a vector pointing upward and a

			vector from the lower-to-higher numbered axial cells.
LHCELI	NHCELI	NCRZ+2	Cell number coupled to the heat-structure nodes at the inner surface.
LHCELO	NHCELO	NCRZ+2	Cell number coupled to the heat-structure nodes at the outer surface.
LHCOMI	NHCOMI	NCRZ+2	Component number of the hydro cell coupled to the heat-structure inner surface.
LHCOMO	NHCOMO	NCRZ+2	Component number of the hydro cell coupled to the heat-structure outer surface.
LHIGH	HIGH	0	Variable not used.
LHS	HS	NCRX* (NFBPWT/4)	Pointer variable for the horizontal-plane shape weight function used.
LHTMLI	HTMLI	NCRZ	Liquid-phase wall heat-transfer multiplier factor for the inner surface.
LHTMLO	HTMLO	NCRZ	Liquid-phase wall heat-transfer multiplier factor for the outer surface.
LHTMVI	HTMVI	NCRZ	Gas-phase wall heat-transfer multiplier factor for the inner surface.
LHTMVO	HTMVO	NCRZ	Gas-phase wall heat-transfer multiplier factor for the outer surface.
LIDROD	IDROD	NRODS	Cell-coupling identifier for rods.
LLAMDA	LAMDA	NDGX	Decay constant of delayed-neutron groups.

LLAMDH	LAMDH	NDHX	Decay constant of decay-heat groups.
LLCHCI	LCHCI	2*(NCRZ+2)	The hydro-cell parameters for heat-transfer coupling to the heat-structure inner surface.
LLCHCO	LCHCO	2*(NCRZ+2)	The hydro-cell parameters for heat-transfer coupling to the heat-structure outer surface.
LMATRD	MATRD	NINT	ROD or SLAB material identification numbers.
LNFAX	NFAX	NCRZ	Rod fine-mesh noding factor.
LNRDX	NRDX	NCRX	Number of actual rods or slabs modeled by the calculational ROD or SLAB element.
LNTSXX	NTSXX	MAX(1, NRIDR)	Number of mesh cells in the plane transverse to the axial direction.
LPGAPT	PGAPT	NCRX	Gap-gas total pressure.
LPLVOL	PLVOL	NCRX	Rod plenum volume.
LPOWLI	POWLI	NCRZ	Total power across the heat-structure inner surface to the liquid.
LPOWLO	POWLO	NCRZ	Total power across the heat-structure outer surface to the liquid.
LPOWVI	POWVI	NCRZ	Total power across the heat-structure inner surface to the gas.
LPOWVO	POWVO	NCRZ	Total power across the heat-structure outer surface to the gas.
LPSLEN	PSLEN	NCRX	Pellet stack length.
LRADRD	RADRD	NODES	Rod node radii (cold).

LRCAL	RCAL	ΣIRCJTB(i,3)+ π_iIRCJTB(i,3)+	Gas volume fraction reactivity-coefficient table. The symbol π_i indicates the product of the following variable taken over the *i* subscript.
LRCBM	RCBM	ΣIRCJTB(i,4)+ π_iIRCJTB(i,4)+	Boron reactivity-coefficient table. The symbol π_i indicates the product of the following variable taken over the *i* subscript.
LRCN	RCN	0 or 4	Reactivity-coefficient values at the beginning of the previous timestep.
LRCTC	RCTC	ΣIRCJTB(i,2)+ π_iIRCJTB(i,2)+	Coolant temperature reactivity-coefficient table. The symbol π_i indicates the product of the following variable taken over the *i* subscript.
LRCTF	RCTF	ΣIRCJTB(i,1)+ π_iIRCJTB(i,1)+	Fuel temperature reactivity-coefficient table. The symbol π_i indicates the product of the following variable taken over the *i* subscript.
LRDPWR	RDPWR	NODES	ROD or SLAB relative radial or thickness power density.
LRDZ	RDZ	NCRZ+1	Axial node positions.
LRPKF	RPKF	NRODS	Supplemental rod power-peaking factor.
LRPWRF	RPWRF	|NRPWRF|*2	Rate-factor table for the power or reactivity table.
LRPWRT	RPWRT	NRPWR	ROD-radial or SLAB-thickness locations where the power shape's relative power densities are defined.
LRPWTB	RPWTB	|NRPWTB|*2	Power or reactivity table.

LRS	RS	NODES*MOD (NFBPWT,2)	Relative ROD-radial or SLAB-thickness power-density distribution used to average reactivity feedback parameters over the reactor-core volume.
LSRP	SRP	0 or 15	Summed programmed and feed-back reactivity changes.
LTC	TC	10	Thermocouple-model input parameters.
LXN	XN	0 or 4	New reactivity-feedback parameter values.
LXO	XO	0 or 4	Old reactivity-feedback parameter values.
LZPW	ZPW	NCRZ+1	Last interpolated axial power
LZPWF	ZPWF	NZPWZ *NODES *IPWRAD	2-D axial-r or axial-x power-shape after numerical integration over the node and hydro-cell lengths.
LZPWFB	ZPWFB	NCRZ+1	Subroutine ZPWHCI evaluated axial-power shape at NCRZ+1 nodes based on the input axial-power shape defined at NZPWZ node locations.
LZPWRF	ZPWRF	|NZPWRF|*2	Axial-power-shape rate-factor table.
LZPWTB	ZPWTB	|NZPWTB|* NZPWZ+1	Relative power density axial-power-shape table.
LZPWZT	ZPWZT	NZPWZ	Axial locations where the axial-power-shape relative power densities are defined.
LZS	ZS	NCRZP1*(MOD (NFBPWT,4)/2)	Relative axial-power-shape power-density distribution used to volume average the reactivity-feedback parameters over the reactor-core volume.

LZZRD	ZZRD	1	Dummy pointer that provides a known end to the common block.

C.4.3. RODPT1.H—Heat-Structure Pointer Table. These pointer variables are declared to be INTEGER.

ROD DATA POINTERS:

Name	Array	Dimension	Description
LALPR	ALPR	NCRZ+2	Gas volume fraction.
LALVR	ALVR	NCRZ+2	Liquid HTC times the interfacial area.
LBITR	BITR	0	Variable not used.
LBITRN	BITRN	0	Variable not used.
LBURN	BURN	NCRZ+1	Fuel burnup.
LCEPWN	CEPWN	1	New total convective power.
LCEPWO	CEPWO	1	Old total convective power.
LCHTIR	CHTIR	NCRZ+2	Gas HTC times interfacial area.
LCLR	CLR	NCRZ+2	Liquid thermal conductivity.
LCND	CND	NODES* (NCRZ+1)	ROD or SLAB thermal conductivity.
LCNDR	CNDR	NINT* (NCRZ+1)	ROD or SLAB thermal conductivity to the right of the interface.
LCONCR	CONCR	NCRZ+2	Mass ratio of dissolved solute in the liquid (kg solute/kg liquid).
LCPDR	CPDR	NINT* (NCRZ+1)	ROD or SLAB specific heat to the right of the interface.
LCPLR	CPLR	NCRZ+2	Liquid specific heat.

LCPND	CPND	NODES* (NCRZ+1)	ROD or SLAB specific heat.
LCPVR	CPVR	NCRZ+2	Gas specific heat.
LCVR	CVR	NCRZ+2	Gas thermal conductivity.
LDRLDT	DRLDT	NCRZ+2	Derivative of the liquid density with respect to the liquid temperature.
LDRVDT	DRVDT	NCRZ+2	Derivative of the gas density with respect to the gas temperature.
LDRZ	DRZ	NCRZ+1	Old zirconium-dioxide reaction depth.
LDRZN	DRZN	NCRZ+1	New zirconium-dioxide reaction depth.
LEAR	EAR	NCRZ+2	Specific internal energy of the noncondensable-gas component.
LELR	ELR	NCRZ+2	Liquid internal energy.
LEMIS	EMIS	NODES* (NCRZ+1)	ROD or SLAB surface emissivity.
LEVR	EVR	NCRZ+2	Gas internal energy.
LFINAR	FINAR	NCRZ+2	Variable not used.
LHDR	HDR	NCRZ+2	Rod-bundle hydraulic diameter.
LHFGR	HFGR	NCRZ+2	Latent heat of vaporization of the fluid.
LHGAMR	HGAMR	NCRZ	Energy contribution to subcooled boiling.
LHGAP	HGAP	NCRZ+1	Gap-gas conductance.
LHLAR	HLAR	NCRZ	Sum of the products of the liquid HTC and the heat-transfer area.

LHLATR	HLATR	NCRZ	Sum of the products of the liquid HTC, the heat-transfer area, and the wall temperature.
LHLSR	HLSR	NCRZ+2	Specific enthalpy of the liquid phase at saturation (corresponding to saturation temperature at the partial pressure of steam).
LHRFG	HRFG	NCRZ+1	New subcooled-boiling HTC.
LHRFGO	HRFGO	NCRZ+1	Old subcooled-boiling HTC.
LHRFL	HRFL	NZMAX	New fine-mesh liquid HTC.
LHRFLO	HRFLO	NZMAX	Old fine-mesh liquid HTC.
LHRFV	HRFV	NZMAX	New fine-mesh gas HTC.
LHRFVO	HRFVO	NZMAX	Old fine-mesh gas HTC.
LHRLG	HRLG	NZMAX	New fine-mesh subcooled-boiling HTC.
LHRLGO	HRLGO	NZMAX	Old fine-mesh subcooled--boiling HTC.
LHRLL	HRLL	NCRZ+1	New liquid HTC for the lower half-node.
LHRLLO	HRLLO	NCRZ+1	Old liquid HTC for the lower half-node.
LHRLV	HRLV	NCRZ+1	New gas HTC for the lower half-node.
LHRLVO	HRLVO	NCRZ+1	Old gas HTC for the lower half-node.
LHQRAD	HQRAD	NCRZ+1	New radiation power absorbed by the coolant.
LHQRDO	HQRDO	NCRZ+1	Old radiation power absorbed by the coolant.

LHVAR	HVAR	NCRZ	Sum of the products of the gas HTC and the heat-transfer area.
LHVATR	HVATR	NCRZ	Sum of the products of the gas HTC, the heat-transfer area, and the wall temperature.
LHVSR	HVSR	NCRZ+2	Specific enthalpy of the steam (not gas) at saturation (at the partial pressure of steam and saturation temperature).
LIDHT	IDHT	NZMAX	ROD or SLAB node identifier.
LIDRGR	IDRGR	NCRZ+2	Flow-regime flag.
LIHTF	IHTF	NZMAX	Fine-mesh heat-transfer regime flag.
LNOHT	NOHT	1	Number of rows of heat-transfer nodes for each ROD or SLAB.
LPAR	PAR	NCRZ+2	Noncondensable-gas partial pressure.
LPGAP	PGAP	NCRZ+1	Gap-gas pressure.
LPINT	PINT	NCRZ+1	Pellet-cladding contact pressure.
LPLDV	PLDV	NCRZ	Pellet dish volume.
LPR	PR	NCRZ+2	Total pressure.
LQCHFF	QCHFF	NCRZ+1	New CHF.
LQCHFO	QCHFO	NCRZ+1	Old CHF.
LQCHFR	QCHFR	NZMAX	New fine-mesh CHF.
LQCHRO	QCHRO	NZMAX	Old fine-mesh CHF.
LQWRX	QWRX	NCRZ+1	Metal-water reaction heat source.
LRADR	RADR	NODES* (NCRZ+1)	Old radial-node positions.

LRADRN	RADRN	NODES*(NCRZ+1)	New radial-node positions.
LRDHLO	RDHLO	NCRZ	Variable not currently implemented.
LRDHLR	RDHLR	NCRZ	Liquid HTC.
LRDHVO	RDHVO	NCRZ	Variable not currently implemented.
LRDHVR	RDHVR	NCRZ	Gas HTC.
LRFT	RFT	NODES*NZMAX	Old fine-mesh ROD or SLAB temperatures.
LRFTN	RFTN	NODES*NZMAX	New fine-mesh ROD or SLAB temperatures.
LRLQLV			Variable not used.
LRND	RND	NODES*(NCRZ+1)	ROD or SLAB density.
LRNDR	RNDR	NINT*(NCRZ+1)–1	ROD or SLAB density to right of the material interface.
LROAR	ROAR	NCRZ+2	Noncondensable-gas density.
LROLR	ROLR	NCRZ+2	Liquid density.
LROMR	ROMR	NCRZ+2	Mixture density.
LROVR	ROVR	NCRZ+2	Gas density.
LRPOWF	RPOWF	NODES	ROD or SLAB power density.
LSIGR	SIGR	NCRZ+2	Surface tension.
LSR	SR	NCRZ+2	Density of plated-out solute.
LSTNU	STNU	NZMAX	Stanton number.
LTCEFN	TCEFN	1	New total convective power.
LTCEFO	TCEFO	1	Old total convective power.

LTCHFF	TCHFF	NZMAX	Fine-mesh wall temperature at the CHF point.
LTCHFR	TCHFR	NCRZ	Wall temperature at the CHF point.
LTLD	TLD	NZMAX	Liquid temperature at bubble departure.
LTLR	TLR	NCRZ+2	Old liquid temperature.
LTLRN	TLRN	NCRZ+2	New liquid temperature.
LTSATR	TSATR	NCRZ+2	Saturation temperature.
LTSSNR	TSSNR	NCRZ+2	Saturation temperature corresponding to the partial pressure of steam.
LTVR	TVR	NCRZ+2	Old gas temperature.
LTVNR	TVNR	NCRZ+2	New gas temperature.
LTWAEN	TWAEN	1	New absolute total conduction.
LTWAEO	TWAEO	1	Old absolute total conduction.
LTWEEN	TWEEN	1	New effective total conduction.
LTWEEO	TWEEO	1	Old effective total conduction.
LVISLR	VISLR	NCRZ+2	Liquid viscosity.
LVISVR	VISVR	NCRZ+2	Gas viscosity.
LVLCR	VLCR	NCRZ+2	Variable not used.
LVLZR	VLZR	NCRZ+2	Axial liquid velocity.
LVMZR	VMZR	NCRZ+2	Axial mixture velocity.
LVOLR	VOLR	NCRZ+2	Fluid volume in hydrodynamic mesh cells.
LVVCR	VVCR	NCRZ+2	Gas cross-flow velocity.

LVVZR	VVZR	NCRZ+2	Axial gas velocity.
LWATR	WATR	NCRZ	ROD or SLAB total heat-transfer area.
LZHT	ZHT	NZMAX	Axial location of the heat-transfer node.

C.5. PIPE COMPONENT

C.5.1. PIPEVLT.H—PIPE Specific Component Table with Common Block pipeCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
BSMASS	BSMASSIND=2	Time-integrated fluid mass flow from the PIPE.
CPOW	CPOWIND=3	Special PIPE power input.
ENINP	ENINPIND=4	Total (time-integrated) energy directly input to the PIPE.
EPSW	EPSWIND=5	Wall surface roughness.
FL(1) FL(2)	FLIND=6 at 7	Liquid mass-flow corrections for mass-conservation checks.
FV(1) FV(2)	FVIND=8 at 9	Gas mass-flow corrections for mass-conservation checks.
HOUTL	HOUTLIND=10	HTC between outer boundary of the PIPE wall and liquid.
HOUTV	HOUTVIND=11	HTC between outer boundary of the PIPE wall and gas.
PLENT	PLENTIND=12	Total length of the PIPE.

POWIN	PWININD=13	Initial power deposited in the liquid.
POWOFF	PWOFFIND=14	Power deposited in the liquid when the trip is OFF after it was ON.
QINT	QINTIND=15	Initial liquid volume in the PIPE.
QOUT	QOUTIND=16	Volume of liquid that has been discharged from the PIPE modeled as an accumulator.
QP3IN	QP3ININD=17	Initial QPPP factor.
QP3OFF	QP3OFFIND=18	QPPP factor when its trip is OFF after it was ON.
RADIN	RADININD=19	Inner radius of the PIPE wall.
RPOWMX	RPOWMXIND=20	Maximum rate of change of power deposited in the coolant.
RQP3MX	RQP3MXIND=21	Maximum rate of change of the QPPP factor.
TH	THIND=22	Thickness of the PIPE wall.
TOUTL	TOUTLIND=23	Liquid temperature outside the PIPE.
TOUTV	TOUTVIND=24	Gas temperature outside the PIPE.
TYPE1	TYPE1IND=28	Type of adjacent component at JUN1.
TYPE2	TYPE2IND=29	Type of adjacent component at JUN2.
VFLOW	VFLOWIND=25	Volume flow rate at fluid discharged from the PIPE modeled as an accumulator.
Z	ZIND=26	Water height above discharge.
Z11111	Z11111IND=27	Dummy variable that provides a known end to the COMMON block real-value variables.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
IACC	IACCIND=–2	PIPE modeled as an accumulator option.
ICHF	ICHFIND=–3	CHF calculation option.
ICJ1	ICJ1IND=–4	Variable not used.
ICJ2	ICJ2IND=–5	Variable not used.
ICONC	ICONCIND=–6	Presence of solute in the liquid option.
IONOFF	IONOFFIND=–7	Number of timesteps the power-deposited-in-the-coolant trip has been ON.
IPF	IPFIND=–8	Last interpolated interval in the power-deposited-in-the-coolant's rate-factor table.
IPOW	IPOWIND=–9	Presence of power deposited in the coolant option.
IPOWSV	IPOWSVIND=–10	Signal-variable or control-block ID number defining the power-deposited-in-the-coolant table's abscissa-coordinate variable.
IPOWTR	IPOWTRIND=–11	Trip ID number that controls the evaluation of the power-deposited-in-the-coolant table.
IPP	IPPIND=–12	Last interpolated interval in the power-deposited-in-the-coolant table.
IQF	IQFIND=–13	Last interpolated interval in the QPPP-factor table's rate-factor table.
IQP	IQPIND=–14	Last interpolated interval in the QPPP-factor table.

IQP3SV	IQP3SVIND=–15	Signal-variable or control-block ID number defining the QPPP-factor table's abscissa-coordinate variable.
IQP3TR	IQP3TRIND=–16	Trip ID number that controls evaluation of the QPPP-factor table.
ISOLLB	ISOLLBIND=–17	Indicator for velocity update at JUN1.
ISOLRB	ISOLRBIND=–18	Indicator for velocity update at JUN2.
JS1	JS1IND=–19	Junction sequence number at cell 1 of the PIPE.
JS2	JS2IND=–20	Junction sequence number at cell NCELLS of the PIPE.
JUN1	JUN1IND=–21	Junction number at cell 1.
JUN2	JUN2IND=–22	Junction number at cell NCELLS.
NCELLS	NCELLSIND=–23	Number of fluid cells in the PIPE.
NONOFF	NONOFFIND=–24	Number of timesteps the QPPP-factor table's controlling trip has been ON.
NPOWRF	NPOWRFIND=–25	Number of data pairs in the power-deposited-in-the-coolant table's rate-factor table.
NPOWSV	NPOWSVIND=–26	Signal-variable or control-block ID number defining the power-deposited-in-the-coolant rate-factor table's abscissa-coordinate variable.
NPOWTB	NPOWTBIND=–27	Number of data pairs in the power-deposited-in-the-coolant table.
NQP3RF	NQP3RFIND=–28	Number of data pairs in the QPPP-factor table's rate-factor table.
NQP3SV	NQP3SVIND=–29	Signal-variable or control-block ID number defining the QPPP-factor table's rate-factor table's abscissa-coordinate variable.

NQP3TB	NQP3TBIND=–30	Number of data pairs in the QPPP-factor table.
ZI1111	ZI1111IND=–31	Dummy variable that provides a known end to the COMMON block.

C.5.2. PIPEPT.H— PIPE Pointer Table. These pointer variables are declared to be INTEGER.

Name	Array	Dimension	Description
LPOWRF	POWRF	NPOWRF*2	Rate-factor table for the power-deposited-in-the-coolant table.
LPOWTB	POWTB	NPOWTB*2	Power-deposited-in-the-coolant table.
LQP3RF	QP3RF	NQP3RF*2	Rate-factor table for the QPPP-factor table.
LQP3TB	QP3TB	NQP3TB*2	QPPP-factor table.

C.6. PLENUM COMPONENT

C.6.1. PLENVLT.H—PLENUM Specific Component Table with Common Block plenCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
BL	BLIND=2	Temporary storage for liquid mass-conservation checks.
BSMASS	BSMASSIND=3	Time-integrated fluid mass flow from the plenum.
BV	BVIND=4	Temporary storage for gas mass-conservation checks.
EPSW	EPWSIND=5	Wall surface roughness.

FAS1	FAS1IND=6	Summed flow area of all junctions on side 1 of the plenum cell.
FAS2	FAS2IND=7	Summed flow area of all junctions on side 2 of the plenum cell.
FLXA	FLXAIND=8	Total noncondensable-gas mass flow into the plenum cell during a timestep.
FLXAL	FLXALIND=9	Total liquid volumetric flow into the plenum cell during a timestep.
FLXAV	FLXAVIND=10	Total gas volumetric flow into the plenum cell during a timestep.
FLXC	FLXCIND=11	Total solute mass flow into the plenum cell during a timestep.
FLXEL	FLXELIND=12	Total liquid internal-energy flow into the plenum cell during a timestep.
FLXEV	FLXEVIND=13	Total gas internal-energy flow into the plenum cell during a timestep.
FLXL	FLXLIND=14	Total liquid mass flow into the plenum cell during a timestep.
FLXV	FLXVIND=15	Total gas mass flow into the plenum cell during a timestep.
RXCL	RXCLIND=16	Temporary storage for the right-hand side of the liquid stabilizer mass and energy equations.
RXCV	RXCVIND=17	Temporary storage for the right-hand side of the gas stabilizer mass and energy equations.
XL	XLIND=18	Gross total liquid volumetric flow from the plenum cell during a timestep.
XV	XVIND=19	Gross total gas volumetric flow from the plenum cell during a timestep.

Z11111	Z11111IND=20	Dummy variable that provides a known end to the COMMON block real-value variables.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
K	KIND=–2	Address location that designates the location of the array-data variables.
ICONC	ICONCIND=–3	Presence of solute in the liquid option.
IPOW	IPOWIND=–4	Presence of power-deposited-in-the-coolant option.
JUNS1	JUNS1IND=–5	Number of junctions on side 1 of the plenum cell that convect momentum across the cell.
JUNS2	JUNS2IND=–6	Number of junctions on side 2 of the plenum cell that convect momentum across the cell.
NCELLS	NCELLSIND=–7	Number of fluid cells (1 for a PLENUM).
NPLJN	NPLJNIND=–8	Number of plenum junctions.
ZI1111	ZI1111IND=–9	Dummy variable that provides a known end to the COMMON block.

C.6.2. PLENPT.H—PLENUM Pointer Table. These pointer variables are declared to be INTEGER.

Name	Array	Dimension	Description
LALW	ALW	NPLJN	Temporary storage for the right-hand side of the liquid stabilizer mass and energy equations.

LAVW	AVW	NPLJN	Temporary storage for the right-hand side of the gas stabilizer mass and energy equations.
LDALP	DALP	NPLJN	Donor-cell gas volume fraction α.
LDBND	DBND	5*NPLJN	Donor-cell quantities $\alpha\rho_v$, $(1-\alpha)\rho_\ell$, $\alpha\rho_a$, $\alpha\rho_v e_v$, and $(1-\alpha)\rho_\ell e_\ell$
LDNFL	DONFL	NPLJN	Donor-cell flag for liquid. 0.0 = defines flow to the plenum cell; 1.0 = defines flow from the plenum cell.
LDNFV	DONFV	NPLJN	Donor-cell flag for gas. 0.0 = defines flow to the plenum cell; 1.0 = defines flow from the plenum cell.
LDXVOL	DXVOL	1	Junction-averaged cell-centered cell length.
LFASMLT	FASMLT	1	Cell-centered interfacial area for stratified flow.
LFAVUL	FAVUL	1	Junction-averaged cell-centered flow area.
LGRAVOL	GRAVOL	1	Difference of junction-averaged positive- and negative-valued GRAVs.
LI0J	I0J	NPLJN	Network-junction numbers.
LJSN	JSN	NPLJN	PLENUM junction-sequence numbers.
LJUNJ	JUNJ	NPLJN	PLENUM junction numbers.
LPAK	PAK	NPLJN	BIT array for the plenum junctions (used only for storing the water packing and stretching bits).

LSGN	SGN	NPLJN	Junction flow-reversal indicators.
LVLVUL	VLVUL	1	Net (output – input) cell-centered liquid velocity per junction.
LVRVPL	VRVPL	1	Absolute value of the net (output – input) cell-centered relative (gas – liquid) velocity.
LVVVUL	VVVUL	1	Net (output – input) cell-centered gas velocity per junction.
LZZZZZ	ZZZZZ	1	Dummy variable that provides a known end to the COMMON block.

C.7. PRESSURIZER COMPONENT

C.7.1. PRIZEVLT.H—PRIZER Specific Component Table with Common Block prizCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
BSMASS	BSMASSIND=2	Time-integrated mass flow from pressurizer.
BSMSSP	BSMSSPIND=3	Current fluid mass-flow rate during a transient calculation.
BXMASS	BXMASSIND=4	Current fluid mass-flow rate during a steady-state calculation.
DPMAX	DPMAXIND=5	Differential pressure at which heaters have maximum power.
EPSW	EPSWIND=6	Wall surface roughness.

FL(1) FL(2)	FLIND=7 at 8	Liquid mass-flow corrections for mass-conservation checks.
FLOW	FLOWIND=9	Volume flow rate at discharge.
FV(1) FV(2)	FVIND=10 at 11	Gas mass-flow corrections for mass-conservation checks.
HOUTL	HOUTLIND=12	HTC between outer boundary of pressurizer wall and liquid.
HOUTV	HOUTVIND=13	HTC between outer boundary of pressurizer wall and gas.
PSET	PSETIND=14	Pressurizer pressure set point for heater-spray control.
QHEAT	QHEATIND=15	Total heater power.
QIN	QININD=16	Heater power being input to the liquid.
QINT	QINTIND=17	Initial liquid volume in pressurizer.
QOUT	QOUTIND=18	Volume of liquid that has been discharged from the pressurizer.
QP3IN	QP3ININD=19	Initial QPPP factor.
RADIN	RADININD=20	Inner radius of pressurizer wall.
TH	THIND=21	Thickness of pressurizer wall.
TOUTL	TOUTLIND=22	Liquid temperature outside the pressurizer.
TOUTV	TOUTVIND=23	Gas temperature outside the pressurizer.
TYPE1	TYPE1IND=27	Type of adjacent component at JUN1.
TYPE2	TYPE2IND=28	Type of adjacent component at JUN2.
Z	ZIND=24	Liquid height above discharge.
ZHTR	ZHTRIND=25	Liquid height for heater cutoff.

Z11111	Z11111IND=26	Dummy variable that provides a known end to the COMMON block for real-value variables.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
ICHF	ICHFIND=–2	CHF calculation option.
ICJ	ICJIND=–3	Variable not used.
ICONC	ICONCIND=–4	Presence of solute in the liquid option.
ICT1	ICT1IND=–5	The sequence number (position in the IORDER array) of the component next to the junction of the pressurizer (this variable is computed but not used).
IUV1	IUV1IND=–6	Indicator for velocity update at JUN1.
IUV2	IUV2IND=–7	Indicator for velocity update at JUN2.
JS1	JS1IND=–8	Junction sequence number at cell 1 of the pressurizer.
JS2	JS2IND=–9	Junction sequence number at cell NCELLS of the pressurizer.
JUN1	JUN1IND=–10	Junction number at cell 1.
JUN2	JUN2IND=–11	Junction number at cell NCELLS.
NCELLS	NCELLSIND=–12	Number of fluid cells.
ZI1111	ZI1111IND=-13	Dummy variable that provides a known end to the COMMON block.

C.8. PUMP COMPONENT

C.8.1. PUMPVLT.H—PUMP Specific Component Table with Common Block pumpCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
AEFFMI	AEFFMIIND=2	The coefficient for the (OMEGA/ROMEGA) **2 term in the calculation of the variable moment of inertia ($kg \cdot m^2$).
ALPHA	ALPHAIND=3	Gas volume fraction.
ALPHAO	ALPHAOIND=4	Gas volume fraction used on previous timestep for the pump-head calculation.
BEFFMI	BEFFMIIND=5	The coefficient for the (OMEGA/ROMEGA) term in the calculation of the variable moment of inertia ($kg \cdot m^2$).
BSMASS	BSMASSIND=6	Time-integrated mass flow from the PUMP.
CEFFMI	CEFFMIIND=7	The constant term in the calculation of the variable moment of inertia ($kg \cdot m^2$).
DELP	DELPIND=8	Delta P across the pump-impeller interface.
DSMOM	DSMOMIND=9	Derivative of the pump head with respect to velocity.
EFFMI	EFFMIIND=10	Moment of inertia.
EFFMI1	EFFMI1IND=11	The alternate effective moment of inertia.
EPSW	EPSWIND=12	Wall surface roughness.
FL(1) FL(2)	FLIND=13 at 14	Liquid mass-flow corrections for mass-conservation checks.

FLOW	FLOWIND=15	PUMP volumetric fluid-flow rate.
FV(1) FV(2)	FVIND=16 at 17	Gas mass-flow corrections for mass-conservation checks.
HEAD	HEADIND=18	Pump head.
HOUTL	HOUTLIND=19	HTC between outer boundary of the PUMP wall and liquid.
HOUTV	HOUTVIND=20	HTC between outer boundary of the PUMP wall and gas.
MFLOW	MFLOWIND=21	PUMP fluid mass-flow rate.
OMEGA	OMEGAIND=22	Pump-impeller rotational speed at old time.
OMEGAN	OMEGANIND=23	Pump-impeller rotational speed at new time.
OMGOFF	OMGOFFIND=24	Pump-impeller rotational speed when its controlling trip is OFF after it was ON.
OMTEST	OMTESTIND=25	The pump-impeller rotational speed below which EFFMI1 (the alternate effective moment of inertia) is used.
QP3IN	QP3ININD=26	Initial QPPP factor.
QP30FF	QP3OFFIND=27	QPPP factor when its controlling trip is OFF after it was ON.
RADIN	RADININD=28	Inner radius of wall.
RFLOW	RFLOWIND=29	Rated fluid flow.
RHEAD	RHEADIND=30	Rated head.
RHO	RHOIND=31	Fluid mixture density.
ROMEGA	ROMEGAIND=32	Rated pump-impeller rotational speed.
ROMGMX	ROMGMXIND=33	Maximum rate of change of the pump-impeller rotational speed.

RQP3MX	RQP3MXIND=34	Maximum rate of change of the QPPP factor.
RRHO	RRHOIND=35	Rated fluid density.
RTORK	RTORKIND=36	Rated torque.
SMOM	SMOMIND=37	Momentum source.
TFR0	TFR0IND=38	Frictional torque constant coefficient.
TFR1	TFR1IND=39	Frictional torque linear coefficient.
TFR2	TFR2IND=40	Frictional torque quadratic coefficient.
TFR3	TFR3IND=41	Frictional torque third-order coefficient.
TFRB	TFRBIND=42	Pump-impeller rotational speed that defines the low-speed regime.
TFRL0	TFRL0IND=43	Low-speed frictional torque constant coefficient.
TFRL1	TFRL1IND=44	Low-speed frictional torque linear coefficient.
TFRL2	TFRL2IND=45	Low-speed frictional torque quadratic coefficient.
TFRL3	TFRL3IND=46	Low-speed frictional torque third-order coefficient.
TH	THIND=47	Wall thickness.
TORQUE	TORQUEIND=48	Pump torque.
TOUTL	TOUTLIND=49	Liquid temperature outside the PUMP wall.
TOUTV	TOUTVIND=50	Gas temperature outside the PUMP wall.
TYPE1	TYPE1IND=52	Type of adjacent component at JUN1.
TYPE2	TYPE2IND=53	Type of adjacent component at JUN2.

Z11111	Z11111IND=51	Dummy variable that provides a known end to the COMMON block real-value variables.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
ICHF	ICHFIND=–2	CHF calculation option.
ICJ1	ICJ1IND=–3	Variable not used.
ICJ2	ICJ2IND=–4	Variable not used.
ICONC	ICONCIND=–5	Presence of solute in the coolant option.
INDXHM	INDXHMIND=–6	Index on head degradation multiplier curve.
INDXTM	INDXTMIND=–7	Index on torque degradation multiplier curve.
IONOFF	IONOFFIND=–8	Number of timesteps the pump-speed controlling trip has been ON.
IPF	IPFIND=–9	Last interpolated interval in the pump-speed table's rate-factor table.
IPM	IPMIND=–10	Two-phase indicator. 0 = use single-phase curves; 1 = use two-phase curves.
IPMPS	IPMPSIND=–11	Flag that indicates whether or not the pump-impeller rotational speed previously has dropped below OMTEST. 0 = pump speed always has been greater than OMTEST; 1 = pump speed has dropped below OMTEST at some time.

IPMPS2	IPMPS2IND=–12	Flag that indicates the evaluation of variable pump inertia in subroutine RPUMP.
IPMPSV	IPMPSVIND=–13	Signal-variable or control-block ID number defining the pump-speed table's independent variable.
IPMPTR	IPMPTRIND=–14	PUMP trip ID number.
IPMPTY	IPMPTYIND=–15	Pump-type number (1 or 2).
IPP	IPPIND=–16	Last interpolated interval in the pump-speed table.
IQF	IQFIND=–17	Last interpolated interval in the QPPP-factor table's rate-factor table.
IQP	IQPIND=–18	Last interpolated interval in the QPPP-factor table.
IQP3SV	IQP3SVIND=–19	Signal-variable or control-block ID number defining the QPPP-factor table's rate-factor table's abscissa-coordinate variable.
IQP3TR	IQP3TRIND=–20	Trip ID number that controls evaluation of the QPPP-factor table.
IRP	IRPIND=–21	Reverse speed indicator. 0 = reverse speed not allowed; 1 = reverse speed allowed.
ISOL1	ISOL1IND=–22	Indicator for velocity update at JUN1.
ISOL2	ISOL2IND=–23	Indicator for velocity update at JUN2.
JS1	JS1IND=–24	Junction sequence number at cell 1 of the PUMP.
JS2	JS2IND=–25	Junction sequence number at cell NCELLS of the PUMP.
JUN1	JUN1IND=–26	Junction number at cell 1.
JUN2	JUN2IND=–27	Junction number at cell NCELLS.

NCELLS	NCELLSIND=–28	Number of fluid cells.
NDATA(1) NDATA(2) .... NDATA(16)	NDATAIND=–29 at –30 at –44	Number of sets of points in head and torque curves.
NDMAX	NDMAXIND=–45	Size of scratch storage array.
NHDM	NHDMIND=–46	Number of data pairs in the head-degradation multiplier curve.
NONOFF	NONOFFIND=–47	Number of timesteps the QPPP-factor table's controlling trip has been ON.
NPMPRF	NPMPRFIND=–48	The number of rate-factor table data pairs whose rate factor is applied to the pump-speed table's independent variable.
NPMPSD	NPMPSDIND=–49	Signal-variable or control-block ID number defining the pump-impeller rotational speed when the pump-speed controlling trip is initially OFF.
NPMPSV	NPMPSVIND=–50	Signal-variable or control-block ID number defining the pump-speed rate-factor table's abscissa-coordinate variable.
NPMPTB	NPMPTBIND=–51	Number of data pairs in the pump-speed table.
NQP3RF	NQP3RFIND=–52	Number of data pairs in the QPPP-factor table's rate-factor table.
NQP3SV	NQP3SVIND=–53	Signal-variable or control-block ID number defining the QPPP-factor rate-factor table's abscissa-coordinate variable.
NQP3TB	NQP3TBIND=–54	Number of data pairs in the QPPP-factor table.
NTDM	NTDMIND=–55	Number of data pairs in the torque-degradation multiplier curve.
OPTION	OPTIONIND=–56	Pump-curve option.

ZI1111	ZI1111IND=–57	Dummy variable that provides a known end to the COMMON block.

C.8.2. PUMPPT.H—PUMP Pointer Table. These pointer variables are declared to be INTEGER.

HEAD- AND TORQUE-TABLE POINTERS:

Name	**Array**	**Dimension**	**Description**
LHSP1	HSP1	2*NDATA(1)	Single-phase head curve 1.
LHSP2	HSP2	2*NDATA(2)	Single-phase head curve 2.
LHSP3	HSP3	2*NDATA(3)	Single-phase head curve 3.
LHSP4	HSP4	2*NDATA(4)	Single-phase head curve 4.
LHTP1	HTP1	2*NDATA(5)	Two-phase head curve 1.
LHTP2	HTP2	2*NDATA(6)	Two-phase head curve 2.
LHTP3	HTP3	2*NDATA(7)	Two-phase head curve 3.
LHTP4	HTP4	2*NDATA(8)	Two-phase head curve 4.
LTSP1	TSP1	2*NDATA(9)	Single-phase torque curve 1.
LTSP2	TSP2	2*NDATA(10)	Single-phase torque curve 2.
LTSP3	TSP3	2*NDATA(11)	Single-phase torque curve 3.
LTSP4	TSP4	2*NDATA(12)	Single-phase torque curve 4.
LTTP1	TTP1	2*NDATA(13)	Two-phase torque curve 1.
LTTP2	TTP2	2*NDATA(14)	Two-phase torque curve 2.
LTTP3	TTP3	2*NDATA(15)	Two-phase torque curve 3.
LTTP4	TTP4	2*NDATA(16)	Two-phase torque curve 4.

MISCELLANEOUS POINTERS:

Name	Array	Dimension	Description
LBD4	BD4	LENBD	Dummy variable.
LHDM	HDM	2*NHDM	Head-degradation multiplier curve.
LIDXCS	IDXCS	16	Curve-set index array.
LNDATA	NDATA	16	Number of data pairs in the head and torque curves.
LPMPRF	PMPRF	NPMPRF*2	Rate-factor table for the pump-speed table.
LPMPTB	PMPTB	NPMPTB*2	Pump-impeller rotational-speed table.
LQP3RF	QP3RF	NQP3RF*2	Rate-factor table for the QPPP-factor table.
LQP3TB	QP3TB	NQP3TB*2	QPPP-factor table.
LTDM	TDM	2*NTDM	Torque-degradation multiplier curve.

C.9. SEPD AND TEE COMPONENTS

C.9.1. TEEVLT.H—SEPD or TEE Specific Component Table with Common Block teeCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
AI	AIIND=2	Standpipe flow area.
ALPD	ALPDIND=6	JCELL gas volume fraction for the separator component.

ALPOT	ALPOTIND=3	Old offtake gas volume fraction.
ALPOTN	ALPOTNIND=4	New offtake gas volume fraction.
ALPOTO	ALPOTOIND=5	Old-old offtake gas volume fraction.
ALPS	ALPSIND=7	Side-arm separator gas volume fraction for the SEPD component.
AN	ANIND=8	Separator nozzle exit area.
BSMASS	BSMASSIND=9	Time-integrated fluid mass flow from the SEPD or TEE.
CA	CAIND=10	Fraction of the liquid velocity at the left face of JCELL that contributes to the momentum transfer into the SEPD or TEE side tube.
CA1	CA1IND=11	Fraction of the liquid velocity at the right face of JCELL that contributes to the momentum transfer into the SEPD or TEE side tube.
CA1V	CA1VIND=12	Fraction of the gas velocity at the right face of JCELL that contributes to the momentum transfer into the SEPD or TEE side tube.
CAV	CAVIND=13	Fraction of the gas velocity at the left face of JCELL that contributes to the momentum transfer into the SEPD or TEE side tube.
COST	COSTIND=14	Cosine of the angle between the low-numbered cells of the SEPD or TEE main tube and side tube.
DELDIM	DELDIMIND=15	Constant in the dryer model (variable not used).
DPSEP	DPSEPIND=16	Pressure drop across the separator.
DPSS	DPSSIND=17	Desired pressure drop across the separator.
ENIN1	ENIN1IND=19	Total (time-integrated) energy directly input to the SEPD or TEE main tube.

ENIN2	ENIN2IND=20	Total (time-integrated) energy directly input to the SEPD or TEE side tube.
EPSW	EPSWIND=18	Wall surface roughness.
FL(1) FL(2) FL(3) FL(4)	FLIND=21 at 22 at 23 at 24	Liquid mass-flow corrections for mass-conservation checks.
FV(1) FV(2) FV(3) FV(4)	FVIND=25 at 26 at 27 at 28	Gas mass-flow corrections for mass-conservation checks.
HOUTL1	HOUTL1IND=29	HTC to liquid at the outer boundary of the SEPD or TEE main-tube wall.
HOUTL2	HOUTL2IND=30	HTC to liquid at the outer boundary of the SEPD or TEE side-tube wall.
HOUTV1	HOUTV1IND=31	HTC to gas at the outer boundary of the SEPD or TEE main-tube wall.
HOUTV2	HOUTV2IND=32	HTC to gas at the outer boundary of the SEPD or TEE side-tube wall.
POWR1	POWR1IND=33	Power per length added to the SEPD or TEE main tube.
POWR2	POWR2IND=34	Power per length added to the SEPD or TEE side tube.
PWIN1	PWIN1IND=35	Initial power deposited in the coolant of the SEPD or TEE main tube.
PWIN2	PWIN2IND=36	Initial power deposited in the coolant of the SEPD or TEE side tube.
PWOFF1	PWOFF1IND=37	Power deposited in the coolant of the SEPD or TEE main tube when its controlling trip is OFF after it was ON.
PWOFF2	PWOFF2IND=38	Power deposited in the coolant of the SEPD or TEE side tube when its controlling trip is OFF after it was ON.

QPIN1	QPIN1IND=39	Initial QPPP factor for the SEPD or TEE main tube.
QPIN2	QPIN2IND=40	Initial QPPP factor for the SEPD or TEE side tube.
QPOFF1	QPOFF1IND=41	QPPP factor for the SEPD or TEE main tube when its controlling trip is OFF after it was ON.
QPOFF2	QPOFF2IND=42	QPPP factor for the SEPD or TEE side tube when its controlling trip is OFF after it was ON.
RADIN1	RADIN1IND=43	Inner radius of the SEPD or TEE main tube.
RADIN2	RADIN2IND=44	Inner radius of the SEPD or TEE side tube.
RH	RHIND=45	Radius of the separator hub at inlet.
RPWMX1	RPWMX1IND=46	Maximum rate of change of power deposited in the coolant for the SEPD or TEE main tube.
RPWMX2	RPWMX2IND=47	Maximum rate of change of power deposited in the coolant for the SEPD or TEE side tube.
RQPMX1	RQPMX1IND=48	Maximum rate of change of the QPPP factor for the SEPD or TEE main tube.
RQPMX2	RQPMX2IND=49	Maximum rate of change of the QPPP factor for the SEPD or TEE side tube.
RR1	RR1IND=50	Radius of larger pickoff ring at first stage of two-stage separator.
RT1L	RT1LIND=51	Coefficient of the SEPD or TEE side-tube coupled momentum-convection term at the left interface of JCELL for liquid.
RT1V	RT1VIND=52	Coefficient of the SEPD or TEE side-tube coupled momentum-convection term at the left interface of JCELL for gas.

RT2L	RT2LIND=53	Coefficient of the SEPD or TEE side-tube coupled momentum-convection term at the right interface of JCELL for liquid.
RT2V	RT2VIND=54	Coefficient of the SEPD or TEE side-tube coupled momentum-convection term at the right interface of JCELL for gas.
TH1	TH1IND=55	Wall thickness of the SEPD or TEE main tube.
TH2	TH2IND=56	Wall thickness of the SEPD or TEE side tube.
THETA	THETAIND=57	Angle between swirling vane and horizontal plane.
TLEN1	TLEN1IND=58	Length of the SEPD or TEE main tube.
TLEN2	TLEN2IND=59	Length of the SEPD or TEE side tube.
TOUTL1	TOUTL1IND=60	Temperature of liquid outside the SEPD or TEE main-tube wall.
TOUTL2	TOUTL2IND=61	Temperature of liquid outside the SEPD or TEE side-tube wall.
TOUTV1	TOUTV1IND=62	Temperature of gas outside the SEPD or TEE main-tube wall.
TOUTV2	TOUTV2IND=63	Temperature of gas outside the SEPD or TEE side-tube wall.
TYPE1	TYPE1IND=70	Type of adjacent component at junction JUN1.
TYPE2	TYPE2IND=71	Type of adjacent component at junction JUN2.
TYPE3	TYPE3IND=72	Type of adjacent component at junction JUN3.
VDRYL	VDRYLIND=64	Lower limit for dryer velocity (currently not available).

VDRYU	VDRYUIND=65	Upper limit for dryer velocity (currently not available).
WLI0	WLI0IND=66	Liquid flow rate into the separator from the previous timestep.
XCO	XCOIND=67	Carryover ratio of liquid mass flow to total mass flow.
XCU	XCUIND=68	Carryunder ratio of gas mass flow to total mass flow.
Z11111	Z11111IND=69	Dummy variable that provides a known end to the COMMON block for real-value variables.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	**Parameter Constant**	**Description**
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
ICBS1	ICBS1IND=–2	Control-block ID number that defines the separator carryover (the liquid mass flow divided by the total mass flow at the JCELL + 1 interface).
ICBS2	ICBS2IND=–3	Control-block ID number that defines the separator carryunder (the gas mass flow divided by the total mass flow at the JCELL + 1 interface).
ICHF	ICHFIND=–4	CHF calculation option.
ICJ1	ICJ1IND=–5	Iteration index of adjacent component to SEPD or TEE at JUN1.
ICJ2	ICJ2IND=–6	Iteration index of adjacent component to SEPD or TEE at JUN2.
ICJ3	ICJ3IND=–7	Iteration index of adjacent component to SEPD or TEE at JUN3.

ICONC1	ICONC1IND=–8	Indicator for presence of solute in the liquid of the SEPD or TEE main tube.
ICONC2	ICONC2IND=–9	Indicator for presence of solute in the liquid of the SEPD or TEE side tube.
IDRY	IDRYIND=–10	Dryer option flag (currently not available).
IENTRN	IENTRNIND=–11	Offtake model option. 0 = model off; 1 = model on (side-tube internal-junction mass flow determined using offtake model).
IONOF1	IONOF1IND=–12	Number of timesteps the power-deposited-in-the-coolant table for the SEPD or TEE main tube's controlling trip has been ON.
IONOF2	IONOF2IND=–13	Number of timesteps the power-deposited-in-the-coolant table for the SEPD or TEE side tube's controlling trip has been ON.
IPF1	IPF1IND=–14	Last interpolated interval number of the rate-factor table for the power-deposited-in-the-coolant table of the SEPD or TEE main tube.
IPF2	IPF2IND=–15	Last interpolated interval number of the rate-factor table for the power-deposited-in-the-coolant table of the SEPD or TEE side tube.
IPOW1	IPOW1IND=–16	Presence of power-deposited-in-the-coolant option for the SEPD or TEE main tube.
IPOW2	IPOW2IND=–17	Presence of power-deposited-in-the-coolant option for the SEPD or TEE side tube.
IPP1	IPP1IND=–18	Last interpolated interval number of the power-deposited-in-the-coolant table for the SEPD or TEE main tube.
IPP2	IPP2IND=–19	Last interpolated interval number of the power-deposited-in-the-coolant table for the SEPD or TEE side tube.

IPWSV1	IPWSV1IND=–20	Signal-variable or control-block ID number defining the power-deposited-in-the-coolant table's abscissa-coordinate variable for the SEPD or TEE main tube.
IPWSV2	IPWSV2IND=–21	Signal-variable or control-block ID number defining the power-deposited-in-the-coolant table's abscissa-coordinate variable for the SEPD or TEE side tube.
IPWTR1	IPWTR1IND=–22	Trip ID number that controls the evaluation of the power-deposited-in-the-coolant table for the SEPD or TEE main tube.
IPWTR2	IPWTR2IND=–23	Trip ID number that controls the evaluation of the power-deposited-in-the-coolant table for the SEPD or TEE side tube.
IQF1	IQF1IND=–24	Last interpolated interval number of the rate-factor table for the QPPP-factor table of the SEPD or TEE main tube.
IQF2	IQF2IND=–25	Last interpolated interval number of the rate-factor table for the QPPP-factor of the SEPD or TEE side tube.
IQP1	IQP1IND=–26	Last interpolated interval number of the QPPP-factor table for the SEPD or TEE main tube.
IQP2	IQP2IND=–27	Last interpolated interval number of the QPPP-actor table for the SEPD or TEE side tube.
IQPSV1	IQPSV1IND=–28	Signal-variable or control-block ID number defining the QPPP-factor table's abscissa-coordinate variable for the SEPD or TEE main tube.
IQPSV2	IQPSV2IND=–29	Signal-variable or control-block ID number defining the QPPP-factor table's abscissa-coordinate variable for the SEPD or TEE side tube.

IQPTR1	IQPTR1IND=–30	Trip ID number that controls the evaluation of the QPPP-factor table for the SEPD or TEE main tube.
IQPTR2	IQPTR2IND=–31	Trip ID number that controls the evaluation of the QPPP-factor table for the SEPD or TEE side tube.
ISEP	ISEPIND=–32	Separator flag.
ISOL1	ISOL1IND=–33	Indicator for velocity update at junction JUN1.
ISOL2	ISOL2IND=–34	Indicator for velocity update at junction JUN2.
ISOL3	ISOL3IND=–35	Indicator for velocity update at junction JUN3.
ISOLN	ISOLNIND=–36	Advanced separator flag.
ISTAGE	ISTAGE=–37	Separator-type option.
JCELL	JCELLIND=–38	Cell number of the internal-junction cell within the SEPD or TEE main tube.
JS1	JS1IND=–39	Junction sequence number at cell 1 of the SEPD or TEE main tube.
JS2	JS2IND=–40	Junction sequence number at cell NCELL1 of the SEPD or TEE main tube.
JS3	JS3IND=–41	Junction sequence number at cell NCELL2 of the SEPD or TEE side tube.
JUN1	JUN1IND=–42	Junction number at cell 1 of the SEPD or TEE main tube.
JUN2	JUN2IND=–43	Junction number at cell NCELL1 of the SEPD or TEE main tube.
JUN3	JUN3IND=–44	Junction number at cell NCELL2 of the SEPD or TEE side tube.
NCELL1	NCELL1IND=–45	Number of fluid cells in the SEPD or TEE main tube.

NCELL2	NCELL2IND=–46	Number of fluid cells in the SEPD or TEE side tube.
NCELLS	NCELLSIND=–47	Total number of main- and side-tube cells plus the internal pseudo cell of the SEPD or TEE (NCELLS = NCELL1 + NCELL2 + 1).
NCSEP	NCSEPIND=–48	Separator flag.
NDRYR	NDRYRIND=–49	Dryer option flag (dryer not available).
NONOF1	NONOF1IND=–50	Number of timesteps the QPPP-factor table controlling trip for the SEPD or TEE main tube has been ON.
NONOF2	NONOF2IND=–51	Number of timesteps the QPPP-factor table controlling trip for the SEPD or TEE side tube has been ON.
NPWRF1	NPWRF1IND=–52	Number of data pairs in the rate-factor table for the power-deposited-in-the-coolant table of the SEPD or TEE main tube.
NPWRF2	NPWRF2IND=–53	Number of data pairs in the rate-factor table for the power-deposited-in-the-coolant table of the SEPD or TEE side tube.
NPWSV1	NPWSV1IND=–54	Signal-variable or control-block ID number defining the rate-factor table's abscissa-coordinate variable for the power-deposited-in-the-coolant table of the SEPD or TEE main tube.
NPWSV2	NPWSV2IND=–55	Signal-variable or control-block ID number defining the rate-factor table's abscissa-coordinate variable for the power-deposited-in-the-coolant table of the SEPD or TEE side tube.
NPWTB1	NPWTB1IND=–56	Number of data pairs in the power-deposited-in-the-coolant table for the SEPD or TEE main tube.

NPWTB2	NPWTB2IND=–57	Number of data pairs in the power-deposited-in-the-coolant table for the SEPD or TEE side tube.
NQPRF1	NQPRF1IND=–58	Number of data pairs in the rate-factor table for the QPPP-factor table of the SEPD or TEE main tube.
NQPRF2	NQPRF2IND=–59	Number of data pairs in the rate-factor table for the QPPP-factor table of the SEPD or TEE side tube.
NQPSV1	NQPSV1IND=–60	Signal-variable or control-block ID number defining the rate-factor table's abscissa-coordinate variable for the QPPP-factor table of the SEPD or TEE main tube.
NQPSV2	NQPSV2IND=–61	Signal-variable or control-block ID number defining the rate-factor table's abscissa-coordinate variable for the QPPP-factor table of the SEPD or TEE side tube.
NQPTB1	NQPTB1IND=–62	Number of data pairs in the QPPP-factor table for the SEPD or TEE main tube.
NQPTB2	NQPTB2IND=–63	Number of data pairs in the QPPP-factor table for the SEPD or TEE side tube.
NSEPS	NSEPSIND=–64	Number of physical separators modeled.
ZI1111	ZI1111IND=–65	Dummy variable that provides a known end to the COMMON block.

C.9.2. TEEPT.H—SEPD or TEE Pointer Table. These pointer variables are declared to be INTEGER. For a SEPD or TEE, NCELLS = NCELL1 + NCELL2 + 1.

Name	Array	Dimension	Description
LAA	AA	ISTAGE	Void profile coefficient inside water layer radius.
LADS	ADS	ISTAGE	Flow area of discharge path.
LBB	BB	ISTAGE	Void profile coefficient within water layer.

LBD4	BD4	LENBD	BD4 array.
LCKS	CKS	ISTAGE	Loss coefficient for discharge passage.
LDDS	DDS	ISTAGE	Hydraulic diameter of discharge passage.
LEFFLD	EFFLD	ISTAGE	Effective L/D coefficient at pick-off ring.
LHBS	HBS	ISTAGE	Length of the separator band.
LHSK	HSK	ISTAGE	Axial distance between discharge and swirling vane.
LPOWRF	POWRF	(|NPWRF1| +|NPWRF2|)*2	Rate-factor table for the power-deposited-in-the-coolant table.
LPOWTB	POWTB	(|NPWTB1| +|NPWTB2|)*2	Power-deposited-in-the-coolant table.
LRWS	RWS	ISTAGE	Inner radius of separator wall.
LRRS	RRS	ISTAGE	Inner radius of the pickoff ring.
LQP3RF	QP3RF	(|NQPRF1| +|NQPRF2|)*2	Rate-factor table for the QPPP-factor table.
LQP3TB	QP3TB	(|NQPTB1| +|NQPTB2|)*2	QPPP-factor table.

C.10. TURBINE COMPONENT

C.10.1. TURBNVLT.H—TURB Specific Component Table with Common Block turbCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.

ALPHA1	ALPHA1IND=2	Upstream gas volume fraction.
ALPHA2	ALPHA2IND=3	Downstream gas volume fraction.
AR	ARIND=4	Area ratio (bucket exit area/nozzle exit area).
BSMASS	BSMASSIND=5	Time-integrated fluid mass flow from the TURB.
COEF1	COEF1IND=6	Nozzle coefficient.
COEF2	COEF2IND=7	Bucket coefficient.
COF3SQ	COF3SQIND=8	Fraction of reaction energy actually delivered in the stage.
CP	CPIND=9	Specific heat at constant pressure.
CPOW	CPOWIND=10	Special turbine input.
DIA	DIAIND=11	Bucket centerline diameter.
DSMOM	DSMOMIND=12	Derivative of SMOM with respect to velocity.
EFFDSN	EFFDSNIND=13	Stage efficiency at design conditions.
EFFSTG	EFFSTGIND=14	Stage efficiency.
ENINP	ENINPIND=15	Total (time-integrated) energy directly input to the TURB.
EPSW	EPSWIND=16	Wall surface roughness.
FL(1) FL(2)	FLIND=17 at 18	Liquid mass-flow corrections for mass-conservation checks.
FLODIR	FLODIRIND=19	Flow direction flag. −1 = indicates normal flow direction is from JUN2 to JUN1; 1 = indicates normal flow direction is from JUN1 to JUN2.
FLOW	FLOWIND=20	Fluid mass-flow rate.

FV(1) FV(2)	FVIND=21 at 22	Gas mass-flow corrections for mass-conservation checks.
GAMMA	GAMMAIND=23	Isentropic exponent of expansion.
PHIREM	PHIREMIND=24	Remaining losses (rotation or diaphragm-packing).
PLENT	PLENTIND=25	Total length of the turbine stage.
POWIN	POWININD=26	Initial power deposited in the coolant.
POWDSN	POWDSNIND=28	Stage power output at design conditions.
POWOFF	POWOFFIND=27	Power deposited in the coolant when the controlling trip is OFF after it was ON.
POWSTG	POWSTGIND=29	Stage power output.
PRES1	PRES1IND=30	Upstream pressure.
PRES2	PRES2IND=31	Downstream pressure.
QUALTY	QUALTYIND=32	Thermodynamic quality of steam.
REACTN	REACTNIND=33	Degree of reaction at design conditions.
RHOL1	RHOL1IND=34	Upstream liquid density.
RHOL2	RHOL2IND=35	Downstream liquid density.
RHOM1	RHOM1IND=36	Upstream mixture density.
RHOM2	RHOM2IND=37	Downstream mixture density.
RHOV1	RHOV1IND=38	Upstream gas density.
RHOV2	RHOV2IND=39	Downstream gas density.
RPOWMX	RPOWMXIND=40	Maximum rate of change of the power deposited in the coolant.
SMOM	SMOMIND=41	Source term in the momentum equation (head gain).
SUPRHT	SUPRHTIND=42	Upstream degree of superheat of steam.

TEMPL1	TEMPL1IND=43	Upstream liquid temperature.
TEMPL2	TEMPL2IND=44	Downstream liquid temperature.
TEMPV1	TEMPV1IND=45	Upstream gas temperature.
TEMPV2	TEMPV2IND=46	Downstream gas temperature.
TYPE1	TYPE1IND=55	Type of adjacent component at junction JUN1.
TYPE2	TYPE2IND=56	Type of adjacent component at junction JUN2.
VELL1	VELL1IND=47	Upstream liquid velocity.
VELL2	VELL2IND=48	Downstream liquid velocity.
VELM1	VELM1IND=49	Upstream mixture velocity.
VELM2	VELM2IND=50	Downstream mixture velocity.
VELV1	VELV1IND=51	Upstream gas velocity.
VELV2	VELV2IND=52	Downstream gas velocity.
VSTAG	VSTAGIND=53	Stagnation velocity.
Z11111	Z11111IND=54	Dummy variable that provides a known end to the COMMON block for real-value variables.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	**Parameter Constant**	**Description**
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
ICJ1	ICJ1IND=–2	Iteration index of adjacent component at junction JUN1.
ICJ2	ICJ2IND=–3	Iteration index of adjacent component at junction JUN2.

ICONC	ICONCIND=–4	Presence of solute dissolved in the liquid coolant option.
IONOFF	IONOFFIND=–5	Number of timesteps the power-deposited-in-the-coolant trip has been ON.
IPF	IPFIND=–6	Last interpolated interval number in the rate-factor table for the power-deposited-in-the-coolant table.
IPOW	IPOWIND=–7	Presence of power-deposited-in-the-coolant option.
IPOWSV	IPOWSVIND=–8	Signal-variable or control-block ID number defining the power-deposited-in-the-coolant table's abscissa-coordinate variable.
IPOWTR	IPOWTRIND=–9	Trip ID number that controls the power-deposited-in-the-coolant table evaluation.
IPP	IPPIND=–10	Last interpolated interval number in the power-deposited-in-the-coolant table.
ISOLLB	ISOLLBIND=–11	Indicator for velocity update at junction JUN1.
ISOLRB	ISOLRBIND=–12	Indicator for velocity update at junction JUN2.
ISTG	ISTGIND=–13	Stage number.
JS1	JS1IND=–14	Junction sequence number at cell 1 of the TURB.
JS2	JS2IND=–15	Junction sequence number at cell NCELLS of the TURB.
JUN1	JUN1IND=–16	Junction number at cell 1 of the TURB.
JUN2	JUN2IND=–17	Junction number at cell NCELLS of the TURB.
LENTRB	LENTRBIND=–18	Length of the TURB block in array data (information pertaining to the entire

		turbine-generator assembly, that is, the sum over all stages).
NCELLS	NCELLSIND=–19	Total number of fluid cells in the TURB.
NEFCON	NEFCONIND=–20	Turbine efficiency. 0 = stage efficiency to be computed at off-design conditions; 1 = constant efficiency.
NPOWRF	NPOWRFIND=–21	Number of data pairs in the power-deposited-in-the-coolant table's rate-factor table.
NPOWSV	NPOWSVIND=–22	Signal-variable or control-block ID number defining the power-deposited-in-the-coolant table's rate-factor table's abscissa-coordinate variable.
NPOWTB	NPOWTBIND=–23	Number of data pairs in the power-deposited-in-the-coolant table.
NROWS	NROWSIND=–24	Number of rows of moving blades.
ZI1111	ZI1111IND=–25	Dummy variable that provides a known end to the COMMON block.

C.10.2. TURBPT.H—TURB Pointer Table. These pointer variables are declared to be INTEGER.

Name	Array	Dimension	Description
LANGL	ANGL	NROWS2	Blade angles.
LPOWRF	POWRF	|NPOWRF|*2	Power-deposited-in-the-coolant table's rate-factor table.
LPOWTB	POWTB	|NPOWTB|*2	Power-deposited-in-the-coolant table.
LTURB	TURB	1	Absolute LCM address for the TURB data common among all stages.

C.11. VALVE COMPONENT

C.11.1. VALVEVLT.H—VALVE Specific Component Table with Common Block valveCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
AVLVE	AVLVEIND=2	VALVE-interface open flow area.
BSMASS	BSMASSIND=3	Time-integrated fluid mass flow from the VALVE.
EPSW	EPSWIND=4	Wall surface roughness.
FAVLVE	FAVLVEIND=5	Fraction of the fully open flow area AVLVE to which the adjustable-valve interface is set.
FL(1) FL(2)	FLIND=6 at 7	Liquid mass-flow corrections for mass-conservation checks.
FMAXOV	FMAXOVIND=8	Maximum flow area fraction or relative valve-stem position during VALVE-interface adjustment by the over-riding trip.
FMINOV	FMINOVIND=9	Minimum flow area fraction or relative valve-stem position during VALVE-interface adjustment by the over-riding trip.
FRIC0	FRIC0IND=10	Fully open VALVE-interface form-loss FRIC for forward flow.
FRIC0R	FRIC0RIND=11	Fully open VALVE-interface form-loss FRIC for reverse flow.
FV(1) FV(2)	FVIND=12 at 13	Gas mass-flow corrections for mass-conservation checks.

HDRDX	HDRDXIND=14	Fully open VALVE-interface hydraulic diameter over DX.
HOUTL	HOUTLIND=15	HTC between outer boundary of the VALVE wall and liquid.
HOUTV	HOUTVIND=16	HTC between outer boundary of the VALVE wall and gas.
HVLVE	HVLVEIND=17	VALVE-interface open hydraulic diameter.
QP3IN	QP3ININD=18	Initial QPPP factor.
QP3OFF	QP3OFFIND=19	QPPP factor when the controlling trip is OFF after it was ON.
RADIN	RADININD=20	Inner radius of the VALVE wall.
RQP3MX	RQP3MXIND=21	Maximum allowed rate of change of the QPPP factor.
RVMX	RVMXIND=22	Maximum rate of change of the VALVE-interface flow area fraction or relative valve-stem position.
RVOV	RVOVIND=23	Rate of change of the VALVE-interface flow area fraction or relative valve-stem position when controlled by the overriding trip that is ON.
TH	THIND=24	Thickness of the VALVE wall.
TOUTL	TOUTLIND=25	Liquid temperature outside the VALVE wall.
TOUTV	TOUTVIND=26	Gas temperature outside the VALVE wall.
TYPE1	TYPE1IND=29	Type of adjacent component at junction JUN1.
TYPE2	TYPE2IND=30	Type of adjacent component at junction JUN2.
XPOS	XPOSIND=27	Adjustable VALVE-interface relative valve-stem position.

Z11111	Z11111IND=28	Dummy variable that provides a known end to the COMMON block for real-value variables.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
ICHF	ICHFIND=–2	CHF calculation option.
ICJ1	ICJ1IND=–3	Iteration index of adjacent component at junction JUN1.
ICJ2	ICJ2IND=–4	Iteration index of adjacent component at junction JUN2.
ICONC	ICONCIND=–5	Presence of solute in the liquid option.
IONOFF	IONOFFIND=–6	Number of timesteps the VALVE table's controlling trip has been ON.
IQF	IQFIND=–7	Last interpolated interval number in the rate-factor table for the QPPP-factor table.
IQP	IQPIND=–8	Last interpolated interval number in the QPPP-factor table.
IQP3SV	IQP3SVIND=–9	Signal-variable or control-block ID number defining the QPPP-factor table's abscissa-coordinate variable.
IQP3TR	IQP3TRIND=–10	Trip ID number that controls evaluation of the QPPP-factor table.
ISOLLB	ISOLLBIND=–11	Indicator for velocity update at junction JUN1.
ISOLRB	ISOLRBIND=–12	Indicator for velocity update at junction JUN2.
IVF	IVFIND=–13	Last interpolated interval number in the rate-factor table for the VALVE table.

IVP1	IVP1IND=–14	Last interpolated interval number in the first VALVE table.
IVP2	IVP2IND=–15	Last interpolated interval number in the second VALVE table.
IVPS	IVPSIND=–16	Adjustable VALVE-interface number.
IVSV	IVSVIND=–17	Signal-variable or control-block ID number that defines the VALVE table's independent variable.
IVTR	IVTRIND=–18	Trip ID number that controls evaluation of the VALVE table(s).
IVTROV	IVTROVIND=–19	Trip ID number that overrides trip IVTR control of VALVE-interface adjustment.
IVTY	IVTYIND=–20	Valve-type option.
IVTYOV	IVTYOVIND=–21	Type of VALVE-interface adjustment by the overriding trip IVTROV.
JS1	JS1IND=–22	Junction sequence number at cell 1 of the VALVE.
JS2	JS2IND=–23	Junction sequence number at cell NCELLS of the VALVE.
JUN1	JUN1IND=–24	Junction number at cell 1 of the VALVE.
JUN2	JUN2IND=–25	Junction number at cell NCELLS of the VALVE.
MODE	MODEIND=–26	Indicator for valve movement over the previous timestep. –1 = closing; 0 = no movement; 1 = opening.
NCELLS	NCELLSIND=–27	Total number of fluid cells in the VALVE.
NONOFF	NONOFFIND=–28	Number of timesteps the QPPP-factor table's controlling trip has been ON.

NQP3RF	NQP3RFIND=–29	Number of data pairs in the rate-factor table for the QPPP-factor table.
NQP3SV	NQP3SVIND=–30	Signal-variable or control-block ID number defining the rate-factor table's abscissa-coordinate variable for the QPPP-factor table.
NQP3TB	NQP3TBIND=–31	Number of data pairs in the QPPP-factor table.
NVRF	NVRFIND=–32	Number of rate-factor table data pairs whose rate factor is applied to the VALVE table's independent variable.
NVSV	NVSVIND=–33	Signal-variable or control-block ID number defining the rate-factor table's abscissa-coordinate variable for the VALVE table(s).
NVTB1	NVTB1IND=–34	Number of data pairs in the first VALVE table.
NVTB2	NVTB2IND=–35	Number of data pairs in the second VALVE table.
ZI1111	ZI1111IND=–36	Dummy variable that provides a known end to the COMMON block for real-value variables.

C.11.2. VLVEPT.H—VALVE Pointer Table. These pointer variables are declared to be INTEGER.

Name	Array	Dimension	Description
LQP3RF	QP3RF	|NQP3RF|*2	Rate-factor table for the QPPP-factor table.
LQP3TB	QP3TB	|NQP3TB|*2	QPPP-factor table.
LVRF	VRF	|NVRF|*2	Rate-factor table for the VALVE table(s).
LVTB1	VTB1	|NVTB1|*2	First VALVE table.
LVTB2	VTB2	|NVTB2|*2	Second VALVE table.

C.12. VESSEL COMPONENT

C.12.1. VSSELVLT.H—VESSEL Specific Component Table with Common Block vssCom.

REAL*8 VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	Parameter Constant	Description
AA1111	AA1111IND=1	Dummy variable that provides a known start to the COMMON block.
BSMASS	BSMASSIND=2	Integrated fluid mass flow from the VESSEL at the start of the timestep.
BSMSSN	BSMSSNIND=3	Integrated fluid mass flow from the VESSEL at the end of the timestep.
CIMFR	CIMFRIND=4	Core inlet fluid mass-flow rate.
CIMFRL	CIMFRLIND=5	Core inlet liquid mass-flow rate.
CIMFRV	CIMFRVIND=6	Core gas volume fraction.
COMFR	COMFRIND=7	Core outlet fluid mass-flow rate.
COMFRL	COMFRLIND=8	Core outlet liquid mass-flow rate.
COMFRV	COMFRVIND=9	Core outlet gas mass-flow rate.
CORELQ	CORELQIND=10	Core liquid volume fraction.
DCFLOW	DCFLOWIND=11	Downcomer fluid mass-flow rate.
DCLQVL	DCLQVLIND=12	Downcomer liquid volume fraction.
EPSW	EPSWIND=13	Wall surface roughness.
GCC	GCCIND=14	Gravity-acceleration constant.
GEOMFC	GEOMFCIND=15	Geometry factor (1.0 = cylindrical, 0.0 = Cartesian).
GRAVZ	GRAVZIND=16	GRAV component in the z-direction.

PCORE	PCOREIND=17	Core average pressure.
PDC	PDCIND=18	Downcomer average pressure.
PLP	PLPIND=19	Lower-plenum average pressure.
PUP	PUPIND=20	Upper-plenum average pressure.
QHSTOT	QHSTOTIND=21	Total heat flux from heat-structure components coupled to the VESSEL.
R0VSM(1) R0VSM(2) R0VSM(3)	R0VSMIND=61 at 62 at 63	Special purpose DOE-model parameter.
SHELV	SHELVIND=22	An addition to the input Z coordinates to give elevations for computing GRAV in one dimension.
TCILMF	TCILMFIND=23	Integrated core-inlet liquid mass flow.
TCIVMF	TCIVMFIND=24	Integrated core-inlet gas mass flow.
TCOLMF	TCOLMFIND=25	Integrated core-outlet liquid mass flow.
TCORE	TCOREIND=26	Core average liquid temperature.
TCOVMF	TCOVMFIND=27	Integrated core-outlet gas mass flow.
TDC	TDCIND=28	Downcomer average liquid temperature.
TLP	TLPIND=29	Lower-plenum average liquid temperature.
TSCORE	TSCOREIND=30	Core average saturation temperature.
TSDC	TSDCIND=31	Downcomer average saturation temperature.
TSLP	TSLPIND=32	Lower-plenum average saturation temperature.
TSUP	TSUPIND=33	Upper-plenum average saturation temperature.
TUP	TUPIND=34	Upper-plenum average liquid temperature.

TX0VSM	TX0VSMIND=67	Special purpose DOE-model parameter.
TY0VSM	TYOVSMIND=68	Special purpose DOE-model parameter.
TZ0VSM	TZOVSMIND=69	Special purpose DOE-model parameter.
VBMASS	VBMASSIND=35	Old fluid mass flow to internal BREAKs.
VBMSSN	VBMSSNIND=36	New fluid mass flow to internal BREAKs.
VCORE	VCOREIND=37	Total liquid mass in the core.
VDCLQ	VDCLQIND=38	Total liquid mass in the downcomer.
VFMASS	VFMASSIND=39	Old fluid mass flow from internal FILLs.
VFMSSN	VFMSSNIND=40	New fluid mass flow from internal FILLs.
VLCORE	VLCOREIND=41	Core liquid mass.
VLPLIQ	VLPLIQIND=42	Lower-plenum liquid volume fraction.
VLPLM	VLPLMIND=43	Lower-plenum liquid mass.
VLPLQ	VLPLQIND=44	Total liquid mass in the lower plenum.
VLQMSS	VLQMSSIND=45	Total liquid mass in the VESSEL.
VOLDC	VOLDCIND=46	Downcomer volume.
VOLLP	VOLLPIND=47	Lower-plenum volume.
VOLUP	VOLUPIND=48	Upper-plenum volume.
VRMATSM(1,1)	VRMATSMIND=52	Special purpose DOE-model parameter.
VRMATSM(2,1)	at 53	
....		
V RMATSM(3,3)	at 60	
VSFLOW	VSFLOWIND=49	VESSEL mass flow.
VUPLIQ	VUPLIQIND=50	Upper-plenum liquid volume fraction.
VUPLM	VUPLMIND=51	Upper-plenum liquid mass.

X0VSM	X0VSMIND=64	Special purpose DOE-model parameter.
Y0VSM	Y0VSMIND=65	Special purpose DOE-model parameter.
Z0VSM	Z0VSMIND=66	Special purpose DOE-model parameter.
Z11111	Z11111IND=70	Dummy variable that provides a known end to the COMMON block for real-value variables.

INTEGER VARIABLES WITH INTEGER PARAMETER CONSTANTS

Variable	**Parameter Constant**	**Description**
IA1111	IA1111IND=–1	Dummy variable that provides a known start to the COMMON block.
IALL	IALLIND=–2	ICXL + NXBCP*NV.
IALLL	IALLLIND=–3	IALL.
IC0	IC0IND=–4	IC0MM + NXBCM*NV.
IC0L	IC0LIND=–5	IC0.
IC0M	IC0MIND=–6	IC0MM + (NXBCM-1)*NV.
IC0ML	IC0MLIND=–7	IC0M.
IC0MM	IC0MMIND=–8	A-array starting location for the VESSEL three-dimensional arrays.
IC0MML	IC0MMLIND=–9	IC0MM.
ICONC	ICONCIND=–10	Presence of solute in the liquid option.
ICRL	ICRLIND=–11	Core lower-boundary segment number, Z(ICRL).
ICRR	ICRRIND=–12	Core outer radial-boundary segment number, RAD(ICRR).
ICRU	ICRUIND=–13	Core upper-boundary segment number, Z(ICRU).

ICX	ICXIND=–14	IC0MM + (NXBCM + NXR-2)*NV.
ICXL	ICXLIND=–15	ICX.
ICXP	ICXPIND=–16	ICXL + NV.
ICXPL	ICXPLIND=–17	ICXP.
IDCL	IDCLIND=–18	Downcomer lower-boundary segment number, Z(IDCL).
IDCR	IDCRIND=–19	Downcomer radial-boundary segment number, RAD(IDCR).
IDCU	IDCUIND=–20	Downcomer upper-boundary segment number, Z(IDCU).
IEXT	IEXTIND=–21	Specifies if the VESSEL input was generated by the post processor EXTRACT. 0 = no; 1 = yes.
IF0	IF0IND=–22	IC0ML if IGEOM.EQ.1.AND.IGBCXR.EQ.1, else IC0L.
IF0L	IF0LIND=–23	IF0.
IFX	IFXIND=–24	IF0L + (NXRV-1)*NV.
IFXL	IFXLIND=–25	IFX.
IGBC	IGBCIND=–26	IGBCXR + IGBCYT + IGBCZ.
IGBCXR	IGBCXRIND=–27	Flag (0 or 1) for internal pressure/velocity boundary conditions on the x- or r-direction walls of the VESSEL.
IGBCYT	IGBCYTIND=–28	Flag (0 or 1) for internal pressure/velocity boundary conditions on the y- or θ-direction walls of the VESSEL.
IGBCZ	IGBCZIND=–29	Flag (0 or 1) for internal pressure/velocity boundary conditions on the z-direction walls of the VESSEL.
IGEOM	IGEOMIND=–30	Vessel-geometry option.

		0 = cylindrical geometry; 1 = Cartesian geometry.
ILCSP	ILCSPIND=–31	Lower-core support-plate axial segment number.
INHSMX	INHSMXIND=–32	Variable not used.
IUCSP	IUCSPIND=–33	Upper-core support-plate axial segment number.
IUHP	IUHPIND=–34	Upper head-plate axial segment number.
IVSSBF	IVSSBFIND=–35	Internal VESSEL boundary condition. 0 = no internal boundaries (default); 2 = first axial level acts like a FILL, last axial level acts like a BREAK; 20 = first axial level acts like a BREAK, last axial level acts like a FILL; 22 = both the first and last axial levels act like BREAKs.
IZBK	IZBKIND=–36	Switch for backup on water packing.
IZBK2	IZBK2IND=–37	Switch for re-donor-cell logic.
IZNX	IZNXIND=–38	Variable used in water-packing logic.
JALL	JALLIND=–39	JCX + NYBCP.
JC0	JC0IND=–40	NYBCM + 1.
JC0M	JC0MIND=–41	NYBCM.
JC0MM	JC0MMIND=–42	NYBCM – 1.
JCX	JCXIND=–43	JC0 + NYT – 1.
JCXP	JCXPIND=–44	JCX + 1.
JF0	JF0IND=–45	JC0M if IGEOM.EQ.1 .AND. IGBCYT.EQ.1, else JC0.
JFX	JFXIND=–46	JF0 + NYT – 1.
KALL	KALLIND=–47	KCX + NZBCP.

KC0	KC0IND=–48	NZBCM + 1.
KC0M	KC0MIND=–49	NZBCM.
KC0MM	KC0MMIND=–50	NZBCM – 1.
KCX	KCXIND=–51	KC0 + NZZ – 1.
KCXP	KCXPIND=–52	KCX + 1.
KF0	KF0IND=–53	KC0M if IGBCZ.EQ.1, else KC0.
KFX	KFXIND=–54	KF0 + NZZ – 1.
LENLD	LENLDIND=–55	Length of level data.
LENLDO	LENLDOIND=–56	Defined to be zero (variable not used).
LFVL	LFVLIND=–57	Relative position of old level-data variables.
LFVNL	LFVNLIND=–58	Relative position of new level-data variables.
LNFVL	LNFVLIND=–59	Number of level-data variables.
LNPTRL	LNPTRLIND=–60	Number of level-data pointers.
LOCVSP	LOCVSPIND=–61	Beginning offset for the VESSEL pointer table.
LSTVSP	LSTVSPIND=–62	Length of the VESSEL pointer table.
LTEMPL	LTEMPLIND=–63	Location of temporary space in the A array to contain one level of data for level-data editing (calculated but variable not used).
NASX	NASXIND=–64	Number of axial segments (levels).
NCELLS	NCELLSIND=–65	Number of fluid cells.
NCLX	NCLXIND=–66	Number of fluid cells per level.
NCRX	NCRXIND=–67	Number of core cells per level.

NCSHM	NCSHMIND=–79	Special purpose DOE-model parameter.
NCSR	NCSRIND=–68	Number of source connections to VESSEL cells.
NIJT	NIJTIND=–69	NI * JALL.
NODHS	NODHSIND=–70	Variable not used.
NRSX	NRSXIND=–71	Number of radial segments.
NSGRID	NSGRIDIND=–72	Number of spacer grids present in the core region (spacer grids are modeled only when the reflood model has been selected by inputting NAMELIST variable NEWRFD = 1).
NTISM	NTISMIND=–80	Special purpose DOE-model parameter.
NTSX	NTSXIND=–73	Number of θ segments.
NVENT	NVENTIND=–74	Number of cells with vent valves in the outer radial surface.
NVVTB	NVVTBIND=–75	Number of input data pairs in the multiple-point vent-valve table.
NXRV	NXRVIND=–76	NRSX if IGEOM.EQ.0 .AND. IGBCXR.NE. 0, else NRSXH if IGEOM. NE.0 .AND. IGBCXR.NE.0, else NRSX–1.
NYTV	NYTVIND=–77	NTSX–1 if IGEOM.EQ.0 .AND. IGBCYT.EQ.0, else NTSX+1 if IGEOM.NE.0 .AND. IGBCYT.NE.0, else 0 if IGEOM.EQ.0 .AND. NTSX.EQ.1, else NTSX.
NZISM	NZISMIND=–81	Special purpose DOE-model parameter.
NZZV	NZZVIND=–78	NASX+1 if IGBCZ.NE.0, else NASX–1.
ZI1111	ZI1111IND=–82	Dummy variable that provides a known end to the COMMON block.

C.12.2. VSSLPT.H—VESSEL Pointer Table. These pointer variables are declared to be INTEGER.

COMMON DATA POINTERS:

Name	Array	Dimension	Description
LALPAG	ALPAG	TSX*NCSX	Old gas volume fraction at the agitated inverted annular flow (IAF).
LALPAN	ALPAN	NTSX*NRSX	New gas volume fraction at the agitated IAF.
LALPCN	ALPCN	NTSX*NRSX	New gas volume fraction at the CHF point.
LALPRN	ALPRN	NTSX*NRSX	New gas volume fraction at the rough-wavy IAF.
LALPRW	ALPRW	NTSX*NRSX	Old gas volume fraction at the rough-wavy IAF.
LALPSM	ALPSM	NTSX*NRSX	Old gas volume fraction at the smooth IAF.
LALPSN	ALPSN	NTSX*NRSX	New gas volume fraction at the smooth IAF.
LALPTN	ALPTN	NTSX*NRSX	New gas volume fraction at the transition boiling.
LAVENT	AVENT	NVENT	Pointer for vent-valve area.
LCTHRZ	CTHRZ	NTSX	Special purpose DOE-model parameter.
LCTHT	CTHT	NTSX	Special purpose DOE-model parameter.
LDPCVN	DPCVN	NVENT	Pointer for vent-valve maximum ΔP to be closed.
LDPOVN	DPOVN	NVENT	Pointer for vent-valve minimum ΔP to be open.

LDR	DR	NRSX	Radial- or x-direction cell length (Δr or Δx).
LDTH	DTH	NTSX	Theta- or y-direction cell length ($\Delta\theta$ or Δy).
LDVLDP	DVLDP	NCSR	Derivative of the liquid velocity with respect to pressure.
LDVVDP	DVVDP	NCSR	Derivative of gas velocity with respect to pressure.
LDZ	DZ	NASX	Axial-direction cell length (Δz).
LESM	ESM	3*NSIZESM	Special purpose DOE-model parameter (NSIZESM = NXRV*NTSX *NASX+NRSX*NYTV*NASX+ NRSX*NTSX*NZZV).
LEVSM	EVSM	3*NSIZESM	Special purpose DOE-model parameter.
LFRCVN	FRCVN	NVENT	FRIC value when the vent valve is closed.
LFROVN	FROVN	NVENT	FRIC value when the vent valve is open.
LFUNH	FUNH	NCLX*NEWRFD	Fraction of the heat-structure surface in each horizontal-plane cell that is unheated.
LGRAVR	GRAVR	NYBCM + NTSX + NYBCP	Radial- or x-direction component of the gravity unit vector on each r- or x-direction interface of a VESSEL cell.
LGRAVT	GRAVT	NYBCM + NTSX + NYBCP	Theta- or y-direction component of the gravity unit vector on each θ- or y-direction interface of a VESSEL cell.

LICJ	ICJ	NCSR	Component number adjacent to a source connection.
LISOLB	ISOLB	NCSR	Indicator for velocity update.
LISRC	ISRC	NCSR	Relative cell number associated with the source connection.
LISRF	ISRF	NCSR	Cell face number associated with the source connection.
LISRL	ISRL	NCSR	Level number associated with the source connection.
LIZINL	IZINL	0	Variable not used.
LIZINS	IZINS	0	Variable not used.
LJSN	JSN	NCSR	Junction sequence number associated with the source connection.
LJUNS	JUNS	NCSR	Junction number associated with the source connection.
LLOCVN	LOCVN	NVENT	Vent-valve location.
LMSC	MSC	NCSR	Absolute cell number of source connection.
LNF1SM	NF1SM	3*NSIZESM	Special purpose DOE-model parameter.
LNF2SM	NF2SM	3*NSIZESM	Special purpose DOE-model parameter.
LNF3SM	NF3SM	3*NSIZESM	Special purpose DOE-model parameter.
LNFCLSM	NFCLSM	NSIZESM	Special purpose DOE-model parameter.
LNFCVSM	NFCVSM	NSIZESM	Special purpose DOE-model parameter.

LNFL4SM	NFL4SM	3*NSIZESM	Special purpose DOE-model parameter.
LNFLSM	NFLSM	3*NSIZESM	Special purpose DOE-model parameter.
LNFV4SM	NFV4SM	3*NSIZESM	Special purpose DOE-model parameter.
LNFVSM	NFVSM	3*NSIZESM	Special purpose DOE-model parameter.
LNHSCA	NHSCA	NTSX*NRSX	Heat-structure element number for average-power rod.
LNSRL	NSRL	NASX	Number of source connections on an axial level.
LPSNEW	PSNEW	NCSR	New source pressure.
LPSOLD	PSOLD	NCSR	Old source pressure.
LRAD	RAD	NRSX	Radial cell outer radius.
LRADSM	RADSM	NRSX*NYTV *NASX	Special purpose DOE-model parameter.
LREFLD	REFLD	NTSX*NRSX	Reflood flag.
LRSM	RSM	3*NSIZESM	Special purpose DOE-model parameter.
LRVSM	RVSM	3*NSIZESM	Special purpose DOE-model parameter.
LSAC	SAC	NCSR*2	Noncondensable-gas mass source.
LSCC	SCC	NCSR *ISOLUT	Solute-concentration mass source. ISOLUT = 0 or 1.
LSLC	SLC	NCSR*2	Liquid mass source.
LSLE	SLE	NCSR*2	Liquid energy source.
LSMOML	SMOML	NCSR*6	Liquid momentum source.

LSMOMV	SMOMV	NCSR*6	Gas momentum source.
LSTHRZ	STHRZ	NTSX	Special purpose DOE-model parameter.
LSTHT	STHT	NTSX	Special purpose DOE-model parameter.
LSVC	SVC	NCSR*2	Gas mass source.
LSVE	SVE	NCSR*2	Vapor energy source.
LTEMPS	TEMPS	LENLD	Temporary array used to output a level of VESSEL data.
LTH	TH	NTSX	Theta-direction cell angle.
LTMSM	TMSM	NRSX*NYTV *NASX	Special purpose DOE-model parameter.
LTPSM	TPSM	NRSX*NYTV *NASX	Special purpose DOE-model parameter.
LVELSL	VELSL	NCSR	Liquid source velocity.
LVELSV	VELSV	NCSR	Gas source velocity.
LVLSM	VLSM	3*NSIZESM	Special purpose DOE-model parameter.
LVVSM	VVSM	3*NSIZESM	Special purpose DOE-model parameter.
LVVTB	VVTB	NVVTB*2	Multiple-point vent-valve table.
LZ	Z	NASX	Axial-direction cell upper elevation.
LZAGS	ZAGS	NTSX*NRSX	Old location of agitated IAF.
LZAGSN	ZAGSN	NTSX*NRSX	New location of agitated IAF.
LZCHFN	ZCHFN	NTSX*NRSX	New location of CHF point.
LZDFS	ZDFS	NTSX*NRSX	Old location of dispersed IAF.

LZDFSN	ZDFSN	NTSX*NRSX	New location of disposed IAF.
LZRWS	ZRWS	NTSX*NRSX	Old location of rough-wavy IAF.
LZRWSN	ZRWSN	NTSX*NRSX	New location of rough-wavy IAF.
LZSGRD	ZSGRD	NTSX*NRSX	New location of grid spacer.
LZSMS	ZSMS	NTSX*NRSX	Old location of smooth IAF.
LZSMSN	ZSMSN	NTSX*NRSX	New location of smooth IAF.
LZTBN	ZTBN	NTSX*NRSX	New location of transition boiling.

LEVEL DATA GRAPHICS IDENTIFIERS:

Name	Array	Dimension	Description
LAID1	AID1	0	Variable not used.
LAID1N	AID1N	0	Variable not used.
LAID2	AID2	0	Variable not used.
LAID2N	AID2N	0	Variable not used.
LALD1	ALD1	0	Variable not used.
LALD1N	ALD1N	0	Variable not used.
LALD2	ALD2	0	Variable not used.
LALD2N	ALD2N	0	Variable not used.
LCFZL	CFZL	NCLX*3 *NFRC3	Graphics identifier for directional form-loss coefficient for liquid.
LCFZV	CFZV	NCLX*3 *NFRC3	Graphics identifier for directional form-loss coefficient for gas.
LCNHS	CNHS	0	Variable not used.
LCNHSN	CNHSN	0	Variable not used.

LCPNS	CPNS	0	Variable not used.
LCPHSN	CPHSN	0	Variable not used.
LDLL	DLL	0	Variable not used.
LDRIV	DRIV	0	Variable not used.
LDROP	DROP	0	Variable not used.
LDVD1	DVD1	0	Variable not used.
LDVD2	DVD2	0	Variable not used.
LDVV	DVV	0	Variable not used.
LEMHS	EMHS	0	Variable not used.
LFAG	FAG	0	Variable not used.
LFRCIN	FRCIN	0	Variable not used.
LFRICI	FRICI	0	Variable not used.
LFRICL	FRICL	0	Variable not used.
LFRICV	FRICV	0	Variable not used.
LGCOND	GCOND	0	Variable not used.
LGEVAP	GEVAP	0	Variable not used.
LHD	HD	0	Variable not used.
LHLV	HLV	0	Variable not used.
LHLVN	HLVN	0	Variable not used.
LHSA	HSA	0	Variable not used.
LHSHL	HSHL	0	Variable not used.
LHSHLO	HSHLO	0	Variable not used.
LHSHV	HSHV	0	Variable not used.

LHSHVO	HSHVO	0	Variable not used.
LHST	HST	0	Variable not used.
LHSTN	HSTN	0	Variable not used.
LHSX	HSX	0	Variable not used.
LICMSH	ICMSH	0	Variable not used.
LIDRGS	IDRGS	0	Variable not used.
LIHSN	IHSN	0	Variable not used.
LISRN	ISRN	0	Variable not used.
LMATHS	MATHS	0	Variable not used.
LQVD1	QVD1	0	Variable not used.
LQVD2	QVD2	0	Variable not used.
LROHS	ROHS	0	Variable not used.
LROHSN	ROHSN	0	Variable not used.
LS1	S1	0	Variable not used.
LS2	S2	0	Variable not used.
LST	ST	0	Variable not used.
LTCHFS	TCHF	0	Variable not used.
LVD1	VD1	0	Variable not used.
LVD1N	VD1N	0	Variable not used.
LVD2	VD2	0	Variable not used.
LVD2N	VD2N	0	Variable not used.
LVL	VL	0	Variable not used.
LVLC	VLC	0	Variable not used.

LHSHVO	HSHVO	0	Variable not used.
LHST	HST	0	Variable not used.
LHSTN	HSTN	0	Variable not used.
LHSX	HSX	0	Variable not used.
LICMSH	ICMSH	0	Variable not used.
LIDRGS	IDRGS	0	Variable not used.
LIHSN	IHSN	0	Variable not used.
LISRN	ISRN	0	Variable not used.
LMATHS	MATHS	0	Variable not used.
LQVD1	QVD1	0	Variable not used.
LQVD2	QVD2	0	Variable not used.
LROHS	ROHS	0	Variable not used.
LROHSN	ROHSN	0	Variable not used.
LS1	S1	0	Variable not used.
LS2	S2	0	Variable not used.
LST	ST	0	Variable not used.
LTCHFS	TCHF	0	Variable not used.
LVD1	VD1	0	Variable not used.
LVD1N	VD1N	0	Variable not used.
LVD2	VD2	0	Variable not used.
LVD2N	VD2N	0	Variable not used.
LVL	VL	0	Variable not used.
LVLC	VLC	0	Variable not used.

AM	Noncondensable-gas mass.
QSL	Wall heat flux.
ARC	Density of solute in cell, $c(1-\alpha)\rho_\ell$.
VOL	Cell fluid volume.
VOLG	Cell geometric volume.
VMFRL	Liquid mass flux in the axial direction.
VMFRV	Gas mass flux in the axial direction.
CPL	Liquid specific heat at constant pressure.
CPV	Gas specific heat at constant pressure.
TSN	Saturation temperature at total pressure.
TSSN	Saturation temperature at steam pressure.
CL	Liquid thermal conductivity.
CV	Gas thermal conductivity.
VISL	Liquid viscosity.
VISV	Gas viscosity.
HFG	Latent heat of vaporization.
HGAM	Energy contribution to phase change from subcooled boiling.
LCCFL	Counter-current flow limitation (CCFL) flag.
FAYT	Actual flow area of the azimuthal θ or y face.
FAZ	Actual flow area of the axial z face.
FAXR	Actual flow area of the radial r or x face.
FAGYT	Geometric flow area of the azimuthal θ or y face.
FAGZ	Geometric flow area of the axial z face.

FAGXR	Geometric flow area of the radial r or x face.
VMYT	Mixture velocity at the azimuthal θ or y face.
VMZ	Mixture velocity at the axial z face.
VMXR	Mixture velocity at the radial r or x face.
HDYT	Hydraulic diameter at the azimuthal θ or y face.
HDZ	Hydraulic diameter at the axial z face.
HDXR	Hydraulic diameter at the radial r or x face.
WFLYT	Wall friction factor for liquid at the azimuthal θ or y face.
WFLZ	Wall friction factor for liquid at the axial z face.
WFLXR	Wall friction factor for liquid at the radial r or x face.
WFVYT	Wall friction factor for gas at the azimuthal θ or y face.
WFVZ	Wall friction factor for gas at the axial z face.
WFVXR	Wall friction factor for gas at the radial r or x face.
VWFMLY	Wall-friction multiplier factor for the liquid at the azimuthal θ or y face.
VWFMLZ	Wall-friction multiplier factor for the liquid at the axial z face.
VWFMLX	Wall-friction multiplier factor for the liquid at the radial r or x face.
VWFMVY	Wall-friction multiplier factor for the gas at the azimuthal θ or y face.
VWFMVZ	Wall-friction multiplier factor for the gas at the axial z face.
VWFMVX	Wall-friction multiplier factor for the gas at the radial r or x face.

DVVYT	Derivative of the gas velocity with respect to pressure at the azimuthal θ or y face.
DVVZ	Derivative of the gas velocity with respect to pressure at the axial z face.
DVVXR	Derivative of the gas velocity with respect to pressure at the radial r or x face.
DVLYT	Derivative of the liquid velocity with respect to pressure at the azimuthal θ or y face.
DVLZ	Derivative of the liquid velocity with respect to pressure at the axial z face.
DVLXR	Derivative of the liquid velocity with respect to pressure at the radial r or x face.
CFZLYT	Liquid forward-flow-direction additive friction-loss coefficient at the azimuthal θ or y face.
CFZLZ	Liquid forward-flow-direction additive friction-loss coefficient at the axial z face.
CFZLXR	Liquid forward-flow-direction additive friction-loss coefficient at the radial r or x face.
CFRLYT	Liquid reverse-flow-direction additive friction-loss coefficient at the azimuthal θ or y face.
CFRLZ	Liquid reverse-flow-direction additive friction-loss coefficient at the axial z face.
CFRLXR	Liquid reverse-flow-direction additive friction-loss coefficient at the radial r or x face.
CFZVYT	Gas forward-flow-direction additive friction-loss coefficient at the azimuthal θ or y face.
CFZVZ	Gas forward-flow-direction additive friction-loss coefficient at the axial z face.
CFZVXR	Gas forward-flow-direction additive friction-loss coefficient at the radial r or x face.

CFRVYT	Gas reverse-flow-direction additive friction-loss coefficient at the azimuthal θ or y face.
CFRVZ	Gas reverse-flow-direction additive friction-loss coefficient at the axial z face.
CFRVXR	Gas reverse-flow-direction additive friction-loss coefficient at the radial r or x face.
DTSDP	Derivative of TSAT with respect to pressure.
DELDP	Derivative of the liquid internal energy with respect to pressure at constant temperature.
DEGDP	Derivative of the steam internal energy with respect to pressure at constant temperature.
DELDT	Derivative of the liquid internal energy with respect to temperature at constant pressure.
DEGDT	Derivative of the steam internal energy with respect to temperature at constant pressure.
DRLDP	Derivative of the liquid density with respect to pressure at constant temperature.
DRGDP	Derivative of the steam density with respect to pressure at constant temperature.
DRLDT	Derivative of the liquid density with respect to temperature at constant pressure.
DRGDT	Derivative of the steam density with respect to temperature at constant pressure.
HVS	Enthalpy of the steam at TSAT.
HLS	Enthalpy of the liquid at TSAT.
DHVS	Derivative of the enthalpy of the gas at TSAT with respect to pressure.
DHLS	Derivative of the enthalpy of the liquid at TSAT with respect to pressure.

DTSSDP	Derivative of the saturation temperature corresponding to the steam pressure with respect to pressure.
DEADT	Derivative of the noncondensable-gas internal energy with respect to temperature at constant pressure.
DEADP	Derivative of the noncondensable-gas internal energy with respect to pressure at constant temperature.
DRADP	Derivative of the noncondensable-gas density with respect to pressure at constant temperature.
DRADT	Derivative of the noncondensable-gas density with respect to temperature at constant pressure.
DRLAST	Variable not used.
ORYT	Scale factor used to reduce cross-flow at the azimuthal θ or y face to simulate the presence of an orifice (currently set to 1).
ORZ	Scale factor used to reduce cross-flow at the axial z face to simulate the presence of an orifice (currently set to 1).
ORXR	Scale factor used to reduce cross-flow at the radial r or x face to simulate the presence of an orifice (currently set to 1).
WMYT	Fraction of the momentum cell at the azimuthal θ or y face that is associated with flow from the upstream cell.
WMZ	Fraction of the momentum cell at the axial z face that is associated with flow from the upstream cell.
WMXR	Fraction of the momentum cell at the radial r or x face that is associated with flow the upstream cell.
DYT	Cell length in the y direction or azimuthal θ sector angle in radians.
DZZ	Cell length in the axial z direction.
DXR	Cell length in the radial r or x direction.
RDYT	Reciprocal of DYT.
RDZ	Reciprocal of DZZ.

RDXR	Reciprocal of DXR.
RMEAN	Radius to the cell center.
RDYTA	Reciprocal of the momentum cell length in the azimuthal θ or y direction.
RDZA	Reciprocal of the momentum cell length in the axial z direction.
RDXRA	Reciprocal of the momentum cell length in the radial r or x direction.
RDDYT	The maximum of RDYTA and FA/VOL of the momentum cell in the azimuthal θ or y direction.
RDDZ	The maximum of RDZA and FA/VOL of the momentum cell in the axial z direction.
RDDXR	The maximum of RDXRA and FA/VOL of the momentum cell in the radial r or x direction.
ALPO	Gas volume fraction at the start of the previous step (α^{n-1}).
DALVA	Variable not used.
DALP	Weighting factor for the new-time level contribution to outflow in the basic mass and energy equations.
FAVYT	Donor-cell averaged gas volume fraction at the azimuthal θ or y face.
FAVZ	Donor-cell averaged gas volume fraction at the axial z face.
FAVXR	Donor-cell averaged gas volume fraction at the radial r or x face.
FALYT	Donor-cell averaged liquid volume fraction at the azimuthal θ or y face.
FALZ	Donor-cell averaged liquid volume fraction at the axial z face.
FALXR	Donor-cell averaged liquid volume fraction at the radial r or x face.

FRVYT	Product of the donor-cell-averaged gas macroscopic density, flow area, and timestep size at the azimuthal θ or y face.
FRVZ	Product of the donor-cell-averaged gas macroscopic density, flow area, and timestep size at the axial z face.
FRVXR	Product of the donor-cell-averaged gas macroscopic density, flow area, and timestep size at the radial r or x face.
FEVYT	Product of the donor-cell-averaged gas internal energy, flow area, and timestep size at the azimuthal θ or y face.
FEVZ	Product of the donor-cell-averaged gas internal energy, flow area, and timestep size at the axial z face.
FEVXR	Product of the donor-cell averaged gas internal energy, flow area, and timestep size at the radial r or x face.
FRAYT	Product of the donor-cell-averaged noncondensable-gas macroscopic density, flow area, and timestep size at the azimuthal θ or y face.
FRAZ	Product of the donor-cell-averaged noncondensible-gas macroscopic density, flow area, and timestep size at the axial z face.
FRAXR	Product of the donor-cell-averaged noncondensible-gas macroscopic density, flow area, and timestep size at the radial r or x face.
FRLYT	Product of the donor-cell-averaged liquid macroscopic density, flow area, and timestep size at the azimuthal θ or y face.
FRLZ	Product of the donor-cell-averaged liquid macroscopic density, flow area, and timestep size at the axial z face.
FRLXR	Product of the donor-cell-averaged liquid macroscopic density, flow area, and timestep size at the radial r or x face.
FELYT	Product of the donor-cell-averaged liquid internal energy, flow area, and timestep size at the azimuthal θ or y face.

FELZ	Product of the donor-cell-averaged liquid internal energy, flow area, and timestep size at the axial z face.
FELXR	Product of the donor-cell averaged liquid internal energy, flow area, and timestep size at the radial r or x face.
CnPm	Variables used as temporaries in a number of routines. Also the coefficient of the change in pressure across the m-th cell face in the equation for the n-th primary dependent variable in the basic step. The variables in order for n = 1, 2, 3, 4, 5 are total pressure, gas temperature, liquid temperature, gas volume fraction, and partial pressure of noncondensable gas. The faces in order from m = 1, 2, 3, 4, 5, 6 are the lower-numbered radial r or x face, the higher-numbered radial r or x face, the lower-numbered azimuthal θ or y face, the higher-numbered azimuthal θ or y face, the lower-numbered axial z face, and the higher-numbered axial z face.
DPRHS	Iterate change in pressure during the basic step before inclusion of effects due to the relative change in pressure across the cell faces.
DARHS	Iterate change in gas volume fraction during the basic step before inclusion of effects due to the relative change in pressure across the cell faces.
DTVRHS	Iterate change in gas temperature during the basic step before inclusion of effects due to the relative change in pressure across the cell faces.
DTLRHS	Iterate change in liquid temperature during the basic step before inclusion of effects due to the relative change in pressure across the cell faces.
DPARHS	Iterate change in the partial pressure of the noncondensable gas during the basic step before inclusion of effects due to the relative change in pressure across the cell faces.
FBIT	Time-independent bit flags.
DVVS1	Scale factor applied to the derivative of the gas velocity at the outer radial r or x face with respect to cell pressure for the water-packing model.

DVVS1M	Scale factor applied to the derivative of the gas velocity at the inner radial r or x face with respect to cell pressure for the water-packing model.
DVLS1	Scale factor applied to the derivative of the liquid velocity at the outer radial r or x face with respect to cell pressure for the water-packing model.
DVLS1M	Scale factor applied to the derivative of the liquid velocity at the inner radial r or x face with respect to cell pressure for the water-packing model.
SC1	Area-ratio scale factor applied to the outer radial r or x face convecting velocities for cross-term contribution to the azimuthal-θ or y and axial-z motion equations.
SC1M	Area-ratio scale factor applied to the inner radial r or x face convecting velocities for cross-term contribution to the azimuthal-θ or y and axial-z motion equations.
DVVS3	Scale-factor applied to the derivative of the gas velocity at the upper axial z face with respect to cell pressure for the water-packing model.
DVVS3M	Scale factor applied to the derivative of the gas velocity at the lower axial z face with respect to cell pressure for the water-packing model.
DVLS3	Scale factor applied to the derivative of the liquid velocity at the upper axial z face with respect to cell pressure for the water-packing model.
DVLS3M	Scale factor applied to the derivative of the liquid velocity at the lower axial z face with respect to cell pressure for the water-packing model.
SC3	Area-ratio scale factor applied to the upper axial z face convecting velocities for cross-term contribution to the radial-r or x and azimuthal-θ or y motion equations.
SC3M	Area-ratio scale factor applied to the lower axial z face convecting velocities for cross-term contribution to the radial-r or x and azimuthal-θ or y motion equations.

DVVS2	Scale factor applied to the derivative of the gas velocity at the forward azimuthal-θ or y face with respect to cell pressure for the water-packing model.
DVLS2	Scale factor applied to the derivative of the liquid velocity at the forward azimuthal-θ or y face with respect to cell pressure for the water-packing model.
SC2	Area-ratio scale factor applied to the forward azimuthal-θ or y face convecting velocities for cross-term contribution to the radial-r or x and axial-z motion equations.
SCD1	Area-ratio scale factor associated with the outer face used in the diagonal V del V term in the radial-r or x motion equation.
SCD1M	Area-ratio scale factor associated with the inner face used in the diagonal V del V term in the radial-r or x motion equation.
SCD2	Area-ratio scale factor associated with the forward face used in the diagonal V del V term in the azimuthal-θ or y motion equation.
SCD3	Area-ratio scale factor associated with the upper face used in the diagonal V del V term in the axial-z motion equation.
SCD3M	Area-ratio scale factor associated with the lower face used in the diagonal V del V term in the axial-z motion equation.
BIT	Bit flags from the previous timestep.
FRCI1	Variable not used.
FRCI2	Variable not used.
FRCI3	Variable not used.
CIYT	Old interfacial drag coefficient at the azimuthal θ or y face.
CIZ	Old interfacial drag coefficient at the axial z face.
CIXR	Old interfacial drag coefficient at the radial r or x face.
CHTI	Old vapor interfacial HTC times the interfacial area.

CHTIA	Old noncondensable-gas interfacial HTC times the interfacial area.
ALV	Old flashing interfacial HTC times the interfacial area.
ALVE	Old liquid interfacial HTC times the interfacial area.
ARV	Old stabilizer macroscopic gas density, $\alpha\rho_v$.
CONCO	Old ratio of the solute mass to the liquid mass.
PA	Old noncondensable-gas partial pressure.
ROA	Old noncondensable-gas density.
EA	Old noncondensable-gas internal energy.
ALP	Old gas volume fraction.
ROV	Old gas density.
ROL	Old liquid density.
S	Old solute mass plated out on structure.
VVYT	Old basic gas velocity at the azimuthal θ or y face.
VVZ	Old basic gas velocity at the axial z face.
VVXR	Old basic gas velocity at the radial r or x face.
VLYT	Old basic liquid velocity at the azimuthal θ or y face.
VLZ	Old basic liquid velocity at the axial z face.
VLXR	Old basic liquid velocity at the radial r or x face.
EV	Old gas internal energy.
EL	Old liquid internal energy.
TV	Old gas temperature.
TL	Old liquid temperature.

GAM	Old vapor generation rate per unit volume.
P	Old total pressure.
AREV	Old stabilizer gas internal energy, $\alpha\rho_v e_v$..
VVTYT	Old stabilizer gas velocity at the azimuthal θ or y face.
VVTZ	Old stabilizer gas velocity at the axial z face.
VVTXR	Old stabilizer gas velocity at the radial r or x face.
ARL	Old stabilizer $(1-\alpha)\rho_\ell$.
AREL	Old stabilizer $(1-\alpha)\rho_\ell e_\ell$.
VLTYT	Old stabilizer liquid velocity at the azimuthal θ or y face.
VLTZ	Old stabilizer liquid velocity at the axial z face.
VLTXR	Old stabilizer liquid velocity at the radial r or x face.
ARA	Old stabilizer $\alpha\rho_a$.
OWVYT	Old donor-cell factor at the azimuthal θ or y)face for gas.
OWVZ	Old donor-cell factor at the axial z face for gas.
OWVXR	Old donor-cell factor at the radial r or x face for gas.
OWLYT	Old donor-cell factor at the azimuthal θ or y face for liquid.
OWLZ	Old donor-cell factor at the axial z face for liquid.
OWLXR	Old donor-cell factor at the radial r or x face for liquid.
BITN	Bit flags for the current timestep.
FRCI1N	Variable not used.
FRCI2N	Variable not used.
FRCI3N	Variable not used.

CINYT	New interfacial drag coefficient at the azimuthal θ or y face.
CINZ	New interfacial drag coefficient at the axial z face.
CINXR	New interfacial drag coefficient at the radial r or x face.
CHTIN	New vapor interfacial HTC times the inter-facial area.
CHTAN	New noncondensable-gas interfacial HTC times the interfacial area.
ALVN	New flashing interfacial HTC times the interfacial area.
ALVEN	New liquid interfacial HTC times the interfacial area.
ARVN	New stabilizer $\alpha\rho_v$.
CONC	New ratio of solute mass to liquid mass.
PAN	New noncondensable-gas partial pressure.
ROAN	New noncondensable-gas density.
EAN	New noncondensable-gas internal energy.
ALPN	New gas volume fraction.
ROVN	New gas density.
ROLN	New liquid density.
SN	New solute mass plated out on structure surfaces.
VVNYT	New basic gas velocity at the azimuthal θ or y face.
VVNZ	New basic gas velocity at the axial z face.
VVNXR	New basic gas velocity at the radial r or x face.
VLNYT	New basic liquid velocity at the azimuthal θ or y face.
VLNZ	New basic liquid velocity at the axial z face.
VLNXR	New basic liquid velocity at the radial r or x face.

EVN	New gas internal energy.
ELN	New liquid internal energy.
TVN	New gas temperature.
TLN	New liquid temperature.
GAMN	New vapor generation rate per unit volume.
PN	New total pressure.
AREVN	New stabilizer $\alpha\rho_v e_v$.
VVNTYT	New stabilizer gas velocity at the azimuthal θ or y face.
VVNTZ	New stabilizer gas velocity at the axial z face.
VVNTXR	New stabilizer gas velocity at the radial r or x face.
ARLN	New stabilizer $(1-\alpha)\rho_\ell$.
ARELN	New stabilizer $(1-\alpha)\rho_\ell e_\ell$.
VLNTYT	New stabilizer liquid velocity at the azimuthal θ or y face.
VLNTZ	New stabilizer liquid velocity at the axial z face.
VLNTXR	New stabilizer liquid velocity at the radial r or x face.
ARAN	New stabilizer $\alpha\rho_a$.
WVYT	New donor-cell factor at the azimuthal θ or y face for gas.
WVZ	New donor-cell factor at the axial z face for gas.
WVXR	New donor-cell factor at the radial r or x face for gas.
WLYT	New donor-cell factor at the azimuthal θ or y face for liquid.
WLZ	New donor-cell factor at the axial z face for liquid.
WLXR	New donor-cell factor at the radial r or x face for liquid.

SPIFZ	Stratified-flow weighting factor for the inter-facial heat-transfer correlations.
DVVS2M	Scale-factor applied to the derivative of the gas velocity at the backward azimuthal θ or y face with respect to cell pressure for the water-packing model.
DVLS2M	Scale-factor applied to the derivative of the liquid velocity at the backward azimuthal θ or y face with respect to cell pressure for the water-packing model.
SC2M	Area-ratio scale factor applied to the back-ward azimuthal θ or y face convecting velocities for cross-term contribution to the radial-r or x and axial-z motion equations.
SCD2M	Area-ratio scale factor associated with the backward face used in the diagonal V del V term in the azimuthal-θ or y motion equation.

These array variables are declared to be REAL*8, dimensioned (NI, NJ), and stored in common block vssWhat. They are used to save VESSEL level data in the heat-structure ROD- or SLAB-surface heat-flux calculation.

Array	**Description**
UALPAG	Gas volume fraction at the agitated-inverted-annular-flow (agitated-IAF) location for moving VESSEL data to heat-structure data.
UALPRW	Gas volume fraction at the rough-wavy-inverted-annular-flow (rough-wavy-IAF) location for moving VESSEL data to heat-structure data.
UALPSM	Gas volume fraction at the smooth-inverted-annular-flow (smooth-IAF) location for moving VESSEL data to heat-structure data.
UZAGS	Location of agitated IAF for moving VESSEL data to heat-structure data.
UZDFS	Location of dispersed IAF for moving VESSEL data to heat-structure data.
UZRWS	Location of rough-wavy IAF for moving VESSEL data to heat-structure data.

UZSMS	Location of smooth IAF for moving VESSEL data to heat-structure data.
UFUNH	Fraction of the heat-structure ROD or SLAB surface that is heated.
INHSCA	Heat-structure element number for the average-power ROD or SLAB.
VALPAG	Gas volume fraction at the agitated-IAF location for moving heat-structure data to VESSEL data.
VALPCF	Gas volume fraction at the CHF-point location for moving heat-structure data to VESSEL data.
VALPRW	Gas volume fraction at the rough-wavy-IAF location for moving heat-structure data to VESSEL data.
VALPSM	Gas volume fraction at the smooth-IAF location for moving heat-structure data to VESSEL data.
VALPTB	Gas volume fraction at the transition-boiling-point location for moving heat-structure data to VESSEL data.
VZAGS	Location of agitated IAF for moving heat-structure data to VESSEL data.
VZCHFL	Location of the CHF point for moving heat-structure data to VESSEL data.
VZDFS	Location of dispersed IAF for moving heat-structure data to VESSEL data.
VZRWS	Location of rough-wavy IAF for moving heat-structure data to VESSEL data
VZSMS	Location of smooth IAF for moving heat-structure data to VESSEL data.
VZTB	Location of transition boiling point for moving heat-structure data to VESSEL data.
MREFLD	Reflood model flag (0=off, 1=on).

APPENDIX D

DESCRIPTION OF TRAC-M COMMON-BLOCK VARIABLES

D.1. BANDW.H

COMMON/BANDW/ MUX, MUY, MUZ

INTEGER VARIABLES:

MUX	The number of diagonal rows above and below the main diagonal lying within the MUX+1+MUX bandwidth of the VESSEL matrix for the x- or θ-directional stabilizer motion equation.
MUY	The number of diagonal rows above and below the main diagonal lying within the MUY+1+MUY bandwidth of the VESSEL matrix for the y- or θ-direction stabilizer motion equation.
MUZ	The number of diagonal rows above and below the main diagonal lying within the MUZ+1+MUZ bandwidth of the VESSEL matrix for the z-direction stabilizer motion equation, pressure semi-implicit equation, and the stabilizer mass and energy equations.

D.2. BKCNTRL.H

COMMON/BKCTRL/	IPREIT, LBCKV, LREIT, LREITV
COMMON/DONR/	ITDON, JDONP, NCOMDP
LOGICAL	LBCKV, LREIT, LREITV

INTEGER VARIABLE:

IPREIT	Flag to print messages on forced reiteration.

LOGICAL VARIABLES:

LBCKV	If .TRUE., then variable forces a timestep backup.
LREIT	If .TRUE., then variable forces a reiteration.
LREITV	If .TRUE., then variable forces a reiteration.

INTEGER VARIABLES:

ITDON	If flow reversals occur for OITNO > ITDON, the timestep is backed up.
JDONP	Cell number in NCOMDP.
NCOMDP	Component number of flow reversal forcing backup.

D.3. BKPOST.H

COMMON/BKPOST/	BKPALL, BKPALU, BKPSTA, BKPSTP, BKPSTT
COMMON/BKPOST/	IBKPST, JBKPST, LBKPST, LBKCYL
LOGICAL	LBKPST, LBKCYL

REAL*8 VARIABLES:

BKPALL	Maximum lower limits on the gas volume fraction such that a backup is forced if the gas volume fraction lies within these limits.
BKPALU	Maximum upper limits on the gas volume fraction such that a backup is forced if the gas volume fraction lies within these limits.
BKPSTA	Gas-volume-fraction variation that is allowed in the POST stage. If the gas-volume-fraction change exceeds BKPSTA, a backup is forced.
BKPSTP	Maximum fractional pressure change that is allowed in the POST stage. If the fractional pressure change exceeds BKSTP, a backup is forced.
BKPSTT	Maximum variation in liquid and gas temperatures that is allowed in the POST stage. If the temperature change exceeds BKPSTT, a backup is forced.

INTEGER VARIABLES:

IBKPST	Component that forces a backup.
JBKPST	Cell number that forces a backup.

LOGICAL VARIABLES:

LBKPST	If .TRUE., then a timestep backup is forced from POST.
LBKCYL	If .TRUE., then a timestep backup is forced because heat-transfer energy conservation is not satisfied.

D.4. BLANKCOM.H

INTEGER	IFXSIZ
PARAMETER	(IFXSIZ=7000000)
REAL*8	A(IFXSIZ)
COMMON	A

REAL*8 VARIABLE:

A	Blank-common container array dimensioned IFXSIZ.

INTEGER LENTITLE
COMMON/NEWSTUF/LENTITLE

INTEGER VARIABLE:

LENTITLE	Number of REAL*8 words of computer memory for the problem title.

INTEGER IFREEAG, IFREEIG, IGSIZE
PARAMETER (IGSIZE=7500)
REAL*8 AG(IGSIZE)
INTEGER IG(IGSIZE)
COMMON/IGCOM/ AG, IFREEAG, IG, IFREEIG

REAL*8 VARIABLE:

AG	Global data container array for REAL*8 variables.

INTEGER VARIABLES:

IFREEAG	First free element of the AG array.
IG	Global data container array for INTEGER variables.
IFREEIG	First free element of the IG array.

INTEGER CSSSIZE
PARAMETER (CSSSIZE=1000)
REAL*8 ACS(CSSSIZE)
INTEGER ICS(CSSSIZE)
COMMON/CSSCOM/ ACS, ICS

REAL*8 VARIABLE:

ACS	Constrained steady-state data container array for REAL*8 variables.

INTEGER VARIABLE:

ICS	Constrained steady-state data container array for INTEGER variables.

INTEGER CTLSIZE
PARAMETER (CTLSIZE=15000)
REAL*8 ACT(CTLSIZE)
INTEGER ICT(CTLSIZE)
COMMON/CTCOM/ ACT, ICT

REAL*8 VARIABLE:

ACT	Control-parameter data container array for REAL*8 variables.

INTEGER VARIABLE:

ICT	Control-parameter data container array for INTEGER variables.

```
INTEGER          SCRSIZE
PARAMETER        (SCRSIZE=15000)
REAL*8           SCRATCH(SCRSIZE), SCRATCH1(SCRSIZE)
INTEGER          ISCRATCH4(SCRSIZE), ISCRATCH(SCRSIZE)
COMMON/SCRCOM/   SCRATCH, SCRATCH1, ISCRATCH4, ISCRATCH
```

REAL*8 VARIABLES:

SCRATCH	Temporary scratch data container array for REAL*8 variables.
SCRATCH1	Temporary scratch data container array for REAL*8 variables.

INTEGER VARIABLES:

ISCRATCH4	Temporary scratch data container array for INTEGER variables.
ISCRATCH	Temporary scratch data container array for INTEGER variables.

```
INTEGER          MAXCOMPS, CURRENTCOMPIND
PARAMETER        (MAXCOMPS=500)
INTEGER          COMPINDICES(MAXCOMPS)
COMMON/COMPINDCOM/COMPINDICES, CURRENTCOMPIND
```

INTEGER VARIABLES:

COMPINDICES	Component-data starting indices in the container A array of blank common.
CURRENTCOMPIND	Element of the COMPINDICES array for the current component.

```
INTEGER          GENTABLESIZE, GENDUMPSIZE
PARAMETER        (GENTABLESIZE=21), (GENDUMPSIZE=29)
REAL*8           RGENTABLE(MAXCOMPS, GENTABLESIZE)
INTEGER          IGENTABLE(MAXCOMPS, GENTABLESIZE)
COMMON/GENTABLECOM/RGENTABLE, IGENTABLE
```

REAL*8 VARIABLE:

RGENTABLE	Generic component table container array for REAL*8 variables.

INTEGER VARIABLE:

IGENTABLE	Generic component table container array for INTEGER variables.

```
INTEGER              SPECTABLESIZE, BREAKDUMPSIZE, FILLDUMPSIZE,
                     PIPEDUMPSIZE, PLENDUMPSIZE, PRIZEDUMPSIZE,
                     PUMPDUMPSIZE, RODDUMPSIZE, TEEDUMPSIZE,
                     TURBNDUMPSIZE, VALVEDUMPSIZE, VSSELDUMPSIZE
PARAMETER            (SPECTABLESIZE=100), (BREAKDUMPSIZE=41),
                     (FILLDUMPSIZE=41), (PIPEDUMPSIZE=60),
                     (PLENDUMPSIZE=29), (PRIZEDUMPSIZE=41),
                     (PUMPDUMPSIZE=110), (RODDUMPSIZE=179),
                     (TEEDUMPSIZE=137), (TURBNDUMPSIZE=81),
                     (VALVEDUMPSIZE=66), (VSSELDUMPSIZE=152)
REAL*8               RSPECTABLE(MAXCOMPS, SPECTABLESIZE)
INTEGER              ISPECTABLE(MAXCOMPS, SPECTABLESIZE)
COMMON/SPECTABLECOM/RSPECTABLE, ISPECTABLE
```

REAL*8 VARIABLE:

RSPECTABLE Specific component table container array for REAL*8 variables.

INTEGER VARIABLE:

ISPECTABLE Specific component table container array for INTEGER variables.

```
INTEGER              H1IND, R1IND, RH1IND
COMMON/CMPINDXIN_RST/H1IND, R1IND, RH1IND
```

INTEGER VARIABLES:

H1IND	Component index of the first heat structure in the TRACIN file.
R1IND	Component index of the first restart hydraulic component.
RH1IND	Component index of the first restart heat-structure component.

```
INTEGER              LASTI
COMMON/LASTINDEX/LASTI
```

INTEGER VARIABLE:

LASTI	Last index element of the container A array in blank common that is used.

D.5. BOIL.H

```
COMMON/BOIL/         COND(NK), CVFAL(NK), DALVJ(NK), DHSDP(NK),
                     DHSDT(NK), EHG(NK), EVAP(NK), FLASH(NK),
                     GAMDP(NK), GAMDPA(NK), GAMMA(NK), SCL(NK)
```

COMMON/BOIL/ ITLEQ(NK)

REAL*8 VARIABLES:

COND	Gas-side heat-transfer coefficient to the gas/liquid interface.
CVFAL	Energy transfer between the gas and liquid based on DALVJ scaling.
DALVJ	Derivative of ALV (FLASH coefficient) with respect to the gas volume fraction (currently set to zero).
DHSDP	Derivative of EHG with respect to the total pressure.
DHSDT	Derivative of EHG with respect to the saturation temperature.
EHG	Internal energy of saturation-temperature gas.
EVAP	Liquid-side heat-transfer coefficient to the gas/liquid interface based on evaporation when the liquid temperature is above the saturation temperature based on vapor pressure.
FLASH	Liquid-side heat-transfer coefficient to the gas/liquid interface based on flashing when the liquid temperature is above the saturation temperature based on total pressure.
GAMDP	Derivative of Γ with respect to the total pressure.
GAMDPA	Derivative of Γ with respect to the noncondensable-gas pressure.
GAMMA	Energy transfer between the gas and liquid based on SCL scaling.
SCL	Scale factor for the phase-change heat-transfer coefficients.

INTEGER VARIABLE:

ITLEQ	Flag to indicate that no evaporation or condensation is expected to occur to the single-phase fluid during the timestep. 0 = evaporation or condensation is evaluated; 1 = no evaporation or condensation is evaluated.

D.6. CCFLCM.H

PARAMETER (MCCFL=10)

COMMON/CCFL/ CCFLM(MCCFL), CCFLC(MCCFL), CBETA(MCCFL), CTRANS(MCCFL), DIAH(MCCFL)

COMMON/CCFL/ NCCFL, NHOLES(MCCFL)

REAL*8 VARIABLES:

CCFLM	Slope of the CCFL correlation.
CCFLC	Constant of the CCFL correlation.
CBETA	Bankoff interpolation constant for interpolating between Wallis and Kutalatze characteristic length dimensions.

CTRANS	Bond number above which the CCFL constant is independent of the Bond number.
DIAH	Diameter of one hole in the perforated plate.

INTEGER VARIABLES:

NCCFL	Number of CCFL parameter sets.
NHOLES	Number of holes in the perforated plate.

D.7. CDBLKS.H

COMMON/CODEBK/ MAX1LV, MAXLEN, MAXLN3, MLNVMT

INTEGER VARIABLES:

MAX1LV	Maximum amount of computer-memory space needed for 3D components when only one level of data is required.
MAXLEN	Maximum amount of computer-memory space needed to process any 1D component.
MAXLN3	Maximum amount of computer-memory space needed to process any 3D component.
MLNVMT	Amount of computer-memory space required to solve the VESSEL matrix.

D.8. CFLOW.H

COMMON/CFLOW/	C1RC, C1RCLT, C2RC, C2RCLT, CHM1, CHM2, CHMLT1, CHMLT2
COMMON/CFLOW/	ICFLOW, IHOR, IHORG
DIMENSION	C1RC(5), C2RC(5), CHM1(5), CHM2(5)

REAL*8 VARIABLES:

C1RC	Five sets of C1 time constants to constrain the choked-flow model interface velocities during transient calculations.
C1RCLT	Default value of the C1 time constant to constrain the choked-flow model interface velocities during transient calculations.
C2RC	Five sets of C2 time constants to constrain the choked-flow model interface velocities during transient calculations.
C2RCLT	Default value of the C2 time constant to constrain the choked-flow model interface velocities during transient calculations.
CHM1	Five sets of choked-flow multipliers for subcooled critical flow.
CHM2	Five sets of choked-flow multipliers for two-phase critical flow.
CHMLT1	Default multiplier for subcooled critical flow.

CHMLT2 Default multiplier for two-phase critical flow.

INTEGER VARIABLES:

ICFLOW Choked-flow option (Namelist variable).

0 = model turned off;

1 = model using default multipliers turned on only for components connected to a BREAK (default value);

2 = model using optional multipliers turned on at cell edges defined by component input (note that this option requires additional array data for all 1D hydrodynamic components).

IHOR Wall-drag form option (Namelist variable).

0 = uses dispersed drag only;

1 = uses stratified drag in 1D if conditions are met (default value);

2 = always uses stratified drag;

3 = turns off head gradient force.

IHORG Variable not used.

D.9. CHECKS.H

COMMON/CHECKS/ DTEND, HDUMP, HEDIT, HGRAF, HSEDIT

COMMON/CHECKS/ NALT, NDID

REAL*8 VARIABLES:

DTEND Time interval during which the special timestep data are used.

HDUMP Saved value of the next data-dump-edit time from the regular timestep data when the special timestep data are used.

HEDIT Saved value of the next large-edit time from the regular timestep data when the special timestep data are used.

HGRAF Saved value of the next graphics-edit time from the regular timestep data when the special timestep data are used.

HSEDIT Saved value of the next small-edit time from the regular timestep data when the special timestep data are used.

INTEGER VARIABLES:

NALT Constant used to determine if gas-volume-fraction adjustments are needed when the interfacial drag is calculated at a 1D junction connected to a BREAK.

NDID ID number of the special timestep data that are being used.

D.10. CHFINT.H

COMMON/CHFINT/ ALPCHF

REAL*8 VARIABLE:

ALPCHF	Gas volume fraction at the critical heat flux (CHF) location.

D.11. CHGALP.H

COMMON/CHGALP/ DAL, DAU, OAL, OAU, XDAL, XDAU, XOAL, XOAU

COMMON/CHGALP/ JDAL, JDAU, JOAL, JOAU, NDAL, NDAU, NOAL, NOAU

REAL*8 VARIABLES:

DAL	Maximum decrease in the gas volume fraction over the timestep.
DAU	Maximum increase in the gas volume fraction over the timestep.
OAL	Maximum decrease in the gas volume fraction immediately following an increase.
OAU	Maximum increase in the gas volume fraction immediately following a decrease.
XDAL	Limit on DAL beyond which the timestep is reduced.
XDAU	Limit on DAU beyond which the timestep is reduced.
XOAL	Limit on OAL beyond which the timestep is reduced.
XOAU	Limit on OAU beyond which the timestep is reduced.

INTEGER VARIABLES:

JDAL	Cell where DAL occurred.
JDAU	Cell where DAU occurred.
JOAL	Cell where OAL occurred.
JOAU	Cell where OAU occurred.
NDAL	Component where DAL occurred.
NDAU	Component where DAU occurred.
NOAL	Component where OAL occurred.
NOAU	Component where OAU occurred.

D.12. CIFLIM.H

COMMON/CIFLIM/ FIFI, FIFR

REAL*8 VARIABLES:

FIFI	Maximum decrease factor for the time-constant constraint on the interfacial-drag coefficient (0.4).

FIFR Maximum increase factor for the time-constant constraint on the interfacial-drag coefficient (2.0).

D.13. CNRSLV.H

COMMON/CNRSLV/ AA(NRFMX1,NRZFMX), BB(NRZFMX), W(NRZFMX)
COMMON/CNRSLV/ KEY, M, M1, N, NRSLV
COMMON/CNRSLV/ ERR
LOGICAL ERR

REAL*8 VARIABLES:

AA	Coefficient matrix.
BB	Right-hand-side (known) vector.
W	Working-area vector.

INTEGER VARIABLES:

KEY	Evaluation-flag option. 1 = solves the linear matrix equation by forward-elimination and backward-substitution. 2 = performs the forward-elimination only. 3 = performs the backward-substitution only.
M	Number of r- or x-direction nodes in the heat-transfer mesh that defines the matrix AA bandwidth of M+1+M.
M1	M + 1.
N	Order of matrix A that is stored in matrix AA.
NRSLV	Axial-direction heat-transfer-calculation numerics option (Namelist variable). 0 = evaluate axial direction explicitly (default); 1 = evaluate axial direction implicitly.

LOGICAL VARIABLE:

ERR	Error flag from subroutine BANSOL that indicates a singular matrix when .TRUE.

D.14. CONCCK.H

COMMON/CONCCK/ JFLAGC

INTEGER VARIABLE:

JFLAGC	Flag that indicates an error in specifying the 1D component input-parameter values.

D.15. CONDHT.H

COMMON/CONDHT/ YLV, YLL

REAL*8 VARIABLES:

YLV	Axial distance above node row JL where the gas-liquid interface is located.
YLL	Axial distance above node row JL where the gas-liquid interface is located.

D.16. CONSTANT.H

COMMON/CONST/ PI, GC, ZERO, ONE, EPSALP, EXPLIM

REAL*8 VARIABLES:

PI	Constant pi (3.1415926535898).
GC	Gravitational constant (9.80665 $m \cdot s^{-2}$).
ZERO	Real constant zero.
ONE	Real constant one.
EPSALP	Gas-volume-fraction cutoff for thermodynamic vapor properties.
EXPLIM	Maximum value of the exponent for time-constant constraint of the heat-transfer coefficients.

D.17. CONTRLLR.H

COMMON/CONTRL/ CPUFLG, DAMMC, DAMX, DELT, DELTHT, DIFMIN, DPRMX, DTLMX, DTMAX, DTMIN, DTO, DTRAT, DTRMX, DTSMX, DTVMX, ENCMAX, EPS1, EPS2, EPSO, ERCEMX, EPSS, ETIME, FRGH, HTLOSI, HTLOSO, ODELT, POWERC, PSSMN, PSSMX, RFAT, RVMAX, TEND, TERCMX, TIMEC, TIMET, TMMAX, VARER, VCMN, VCMX, VMAXO, VMAXT, VMAXT3, VMCON, VMNEW, VMOLD, VMXT3O, XTABLE, X0SM, Y0SM, Z0SM, OMSASM, WSASM, WDSASM, T0SM(3), XVSM, YVSM, ZVSM, RMATSM(3,3), OMSM(3), WSM(3), WDSM(3), ERRSM, DTSM

COMMON/CONTRL/ DSTEP, IADDED, IBLAUS, ICCMX, ICMP, ICMPMX, ICP, IDIAG, IDIAGS, IECCPX, IEOS, IFF3D, IFPREP, IGEOM3, IM100, IM100X, IMFR, INVAN, IOFFTK, IPAK, IPAK3D, IPAKON, IPKPMP, IRESET, IRSFLG, IRSTFL, ISOLUT,

	ISSFLG, ISTDY, ISTTC, ITHD, ITMIN, ITPAKO, ITRANS, JFAT, KCCMX, LCMPTR, LEVSTG, LLVFLG, NCMN, NCMX, NCONTR, NCONTS, NCONTT, NCRG, NDIA1, NENCL, NEWRFD, NFRC1, NFRC3, NITAV, NITMN, NITMX, NLOOPP, NOSETS, NSEND, NSEO, NSMN, NSMX, NSPL, NSPU, NSSO, NSTAB, NSTP, NVGRAV, NVPOW, OITMAX, SITMAX, STDYST, TRANSI, MOTSM, STATSM, SAXSM
INTEGER	DSTEP, OITMAX, SITMAX, STDYST, TRANSI, STATSM, SAXSM

REAL*8 VARIABLES:

CPUFLG	Option for eliminating the cpu time from being output to files TRCMSG and TRCOUT and the terminal so that a DIFF file comparison between TRAC-P versions will not include the cpu-time differences between calculations (Namelist variable).
DAMMC	Maximum gas-volume-fraction change during the timestep (not used).
DAMX	Error caused by the relative change in the gas volume fraction (not used).
DELT	Current timestep size for advancement in time of the finite-difference-equation solution.
DELTHT	Heat-transfer timestep size.
DIFMIN	Minimum diffusion number required for stability of the ROD or SLAB conduction solution.
DPRMX	Maximum pressure change during the timestep.
DTLMX	Maximum liquid-temperature change during the timestep.
DTMAX	Maximum allowable timestep size for the time interval.
DTMIN	Minimum allowable timestep size for the time interval.
DTO	Previous timestep size.
DTRAT	Ratio of the previous timestep size to the reduced timestep size that results in a trip (assigned special timestep data) crossing its setpoint at the end of the timestep.
DTRMX	Maximum ROD or SLAB temperature change during the timestep.
DTSMX	Maximum metal-temperature change during the timestep.
DTVMX	Maximum gas-temperature change during the timestep.

ENCMAX	Worst-case convection-power difference from a timestep.
EPS1	The lower-bound criterion for increasing the Kaganove-method integration timestep for solving the point-reactor kinetics equations.
EPS2	The upper-bound criterion for decreasing the Kaganove-method integration timestep for solving the point-reactor kinetics equations.
EPSO	Convergence criterion for the outer iteration.
ERCEMX	Worse-case convection-power difference during a calculation.
EPSS	Convergence criterion for the steady-state calculation.
ETIME	Current calculation time.
FRGH	Multiplier applied to the gravity-head term in all motion equations (Namelist variable; 1.0 default value).
HTLOSI	Wall inner-surface heat loss by 1D components only (total system heat loss from the fluid to the wall inner surface for 1D hydraulic components only).
HTLOSO	Wall outer-surface heat loss by 1D components only (total system heat loss from the wall outer surface to the exterior surroundings for 1D hydraulic components only).
ODELT	Previous timestep size.
POWERC	Maximum convection-power difference between what goes into the fluid and what comes from the wall in convection heat-transfer from HTSTRs.
PSSMN	Minimum steam-generator secondary-side pressure.
PSSMX	Maximum steam-generator secondary-side pressure.
RFAT	Maximum ratio of the interface flow area to the adjacent-mesh-cell average flow area.
RVMAX	Maximum ratio of the adjacent-mesh-cell average flow areas when their interface does not have an additive loss coefficient specified.
TEND	End time for the timestep data domain.
TERCMX	Time at which the worst-case power difference occurred during a calculation.
TIMEC	Clock time in seconds.
TIMET	Current calculation time.

TMMAX	Time at which the worse-case convection-power difference occurred during a calculation.
VARER	Variable error.
VCMN	Final convergence for component NCMN at step NSMN.
VCMX	Final convergence for component NCMX at step NSMX.
VMAXO	1D component maximum ratio of the Courant number to the timestep size at the beginning of the previous timestep.
VMAXT	1D component maximum ratio of the Courant number to the timestep size at the beginning of the present timestep.
VMAXT3	3D VESSEL component maximum ratio of the Courant number to the timestep size at the beginning of the present timestep.
VMCON	Net water mass (liquid plus vapor) convected into VESSEL component(s) during the time interval $t^{n+1} - t^n$.
VMNEW	VESSEL water mass (liquid plus vapor) at t^{n+1}.
VMOLD	VESSEL water mass (liquid plus vapor) at t^n.
VMXT3O	3D VESSEL component maximum ratio of the Courant number to the timestep size at the beginning of the previous timestep.
XTABLE	Abscissa-coordinate value from the last axial power-shape table evaluation.
X0SM	Special purpose DOE-model parameter.
Y0SM	Special purpose DOE-model parameter.
Z0SM	Special purpose DOE-model parameter.
OMSASM	Special purpose DOE-model parameter.
WSASM	Special purpose DOE-model parameter.
WDSASM	Special purpose DOE-model parameter.
T0SM(3)	Special purpose DOE-model parameter.
XVSM	Special purpose DOE-model parameter.
YVSM	Special purpose DOE-model parameter.
ZVSM	Special purpose DOE-model parameter.
RMATSM(3,3)	Special purpose DOE-model parameter.
OMSM(3)	Special purpose DOE-model parameter.
WSM(3)	Special purpose DOE-model parameter.
WDSM(3)	Special purpose DOE-model parameter.
ERRSM	Special purpose DOE-model parameter.
DTSM	Special purpose DOE-model parameter.

INTEGER VARIABLES:

DSTEP	Timestep number of the data dump to be used for the restart calculation.
IADDED	Number-of-timesteps interval for printing calculation summary to the terminal and TRCMSG file (0 suppresses this printout).
IBLAUS	Option to apply the Blasius interfacial-drag correlation in the downcomer and lower plenum of the VESSEL components (Namelist variable).
ICCMX	Component number in the IORDER array having the most severe timestep limit for numerical stability of the calculation.
ICMP	Component indicator.
ICMPMX	Component number in which the worse-case convection-power difference occurred during the timestep.
ICP	Temporary pointer to next free location in the dynamic computer-memory space for component data.
IDIAG	Option that defines different levels of debugging information of appropriate parameter values (Namelist variable).
IDIAGS	Option to select alternate variables to be written in a large edit to the TRCOUT file for 1D hydraulic components when IDIAG > 0 (Namelist variable).
IECCPX	Component number in which the worse-case convection-power difference occurred during the calculation.
IEOS	Air-water option 0 = steam water, noncondensable gas, and liquid water in fluid; 1 = noncondensable gas and liquid water (no steam water) in fluid.
IFF3D	Outer-iteration VESSEL-evaluation flag. 0 = evaluate the VESSEL-coefficient matrix equation; 1 = back-substitute the VESSEL matrix-equation solution.
IFPREP	Flag that indicates sections of PREPER to be executed (nonzero only for 1D cores).

IGEOM3 — VESSEL-geometry flag.

0 = flow areas between the downcomer and inside of the VESSEL set to zero (default value);

1 = flow areas between the downcomer and inside of the VESSEL maintained at the user input values.

Note: The vent-valve option overrides the IGEOM3 = 1 option in cells that have vent-valve connections.

IM100 — Flag that indicates if the back up occurred during previous timestep (used for mass check on logic).

IM100X — Flag that indicates whether the previous timestep that failed was obtained from a restart.

IMFR — Calculates the azimuthal-θ or y, axial-z, and radial-r or x mass flows for both liquid and gas, and outputs them to the TRCGRF graphics file (Namelist variable).

1 = outputs no phasic mass flows (default);

3 = outputs 3D VESSEL mass flow.

INVAN — Option to select either T_{CHF} or T_{sat} for control of the inverted annular-flow regime.

IOFFTK — Option to select the TEE offtake model.

0 = offtake model off;

1 = offtake model on.

IPAK — 1D hydraulic-component water-packing option.

0 = off;

1 = on.

IPAK3D — 3D VESSEL water-packing option.

0 = off;

1 = on.

IPAKON — Flag that indicates if water-packing logic is on during the timestep.

IPKPMP — Flag that indicates if water-packing corrections are made at a pump momentum-source interface.

0 = no (default);

1 = yes.

IRESET — Option to reinitialize the energy error to zero at the start of a restart calculation.

0 = no (allow the energy error to accumulate from the previous calculation.

1 = yes.

IRSFLG Composite number of the number of signal variables, control blocks, and trips that need to be read from the TRCRST restart file.

IRSTFL Variable not used.

ISOLUT Solute-tracking option for the entire system model.

0 = off;

1 = on.

ISSFLG Flag that controls steady-state convergence editing.

ISTDY Flag that indicates the type of calculation.

0 = transient;

1 = steady state.

ISTTC Static-check flag.

0 = normal mode;

1 = a static-balance check was requested when STDYST = 5 was input.

ITHD Option for inputting heat-transfer diameters for HTSTRs.

0 = no (heat-transfer diameters defined by hydraulic diameters);

1 = yes.

ITMIN Minimum stable-film-boiling option.

ITPAKO Iteration number at which water packing was detected.

ITRANS Value of TRANSI from the data-dump calculation read from file TRCRST or the value 0 for an initial calculation where all input data is read from file TRACIN.

JFAT Flow-area-ratio test-results flag.

0 = flow-area ratios are appropriate;

1 = one or more ratios of the interface flow area to the adjacent mesh-cell volume-average flow area are invalid.

2 = one or more ratios of the adjacent mesh-cell volume-average flow areas are invalid.

3 = one or more of both types of flow-area ratios are invalid.

KCCMX Component number of cell that limits stability.

LCMPTR Pointer to end of component data for last component read.

LEVSTG — Gas-volume-fraction averaging flag.

0 = no gas-volume-fraction averaging is performed in HTCOR for steam-generator secondaries (default value);

1 = special gas-volume-fraction averaging is performed in HTCOR for steam-generator secondaries.

LLVFLG — Switch that determines averaging procedure used in subroutine HTIF.

NCMN — Element number the in IORDER array for the component that was last to converge at timestep NSMN.

NCMX — Element number in the IORDER array for the component that was last to converge at timestep NSMX.

NCONTR — Number of constrained steady-state controllers.

NCONTS — Number of constrained steady-state controllers that adjust pumps or valves so that their coolant mass flow equals a monitored coolant mass flow elsewhere in the system.

NCONTT — Number of constrained steady-state controllers that adjust the flow resistance across the VESSEL (NCONTT = 0).

NCRG — Variable (not documented elsewhere and defaulted to zero) that could be used to force the input of ICRRG (see the VESSEL variable-length table, Appendix C). Logic is incomplete.

NDIA1 — Heat-transfer diameter input option for 1D components (Namelist variable).

1 = no heat-transfer diameter input for 1D components (default value);

2 = heat-transfer diameter input for 1D components.

NENCL — Total number of radiation enclosures in the radiation heat-transfer model (Namelist variable).

NEWRFD — Option that activates the reflood-model calculation for HTSTR components coupled to VESSEL components when internal test criteria are satisfied (Namelist variable).

0 = off;

1 = on.

NFRC1 — Additive-loss-coefficient defining form option for 1D components (Namelist variable).

1 = FRIC additive loss coefficients are input for both flow directions (default);

2 = FRIC and RFRIC forward- and reverse-flow additive loss coefficients are input.

NFRC3 Additive-loss-coefficient defining form option for VESSEL components (Namelist variable).

NITAV Average number of outer iterations since the last small edit.

NITMN Minimum number of outer iterations since the last small edit.

NITMX Maximum number of outer iterations since the last small edit.

NLOOPP Flag to indicate inconsistent source connections of a 1D component loop to different directional cell faces of VESSEL component(s).

NOSETS Option for evaluating the SETS3D equations for all VESSEL components (Namelist variable).

0 = evaluate the SETS3D equations when the timestep size exceeds 0.8 times the VESSEL Courant-limit timestep size;

1 = do not evaluate the SETS3D equations;

2 = evaluate the SETS3D equations every timestep (default).

NSEND End the calculation at this timestep number (Namelist variable).

NSEO Timestep number of last completed edit.

NSMN Last timestep at which NITMN outer iterations occurred.

NSMX Last timestep at which NITMX outer iterations occurred.

NSPL Debug print output if NSPL < NSTEP < NSPU.

NSPU Debug print output if NSPL < NSTEP < NSPU.

NSSO Timestep number of last completed small edit.

NSTAB SETS3D-equations evaluation flag for all VESSEL components.

NSTP Number of timesteps since the last edit.

NVGRAV Option to allow the orientation of each VESSEL component to be input specified (Namelist variable).

NVPOW Number of VESSEL components in the input and restart data files that are coupled to powered HTSTR components (Namelist variable).

OITMAX Maximum number of outer iterations for a transient calculation.

SITMAX Maximum number of outer iterations for a steady-state calculation.

STDYST Steady-state calculation indicator.

TRANSI Transient calculation indicator.

MOTSM Special purpose DOE-model parameter.

STATSM	Special purpose DOE-model parameter.
SAXSM	Special purpose DOE-model parameter.

D.18. COUPLE.H

COMMON/COUPLE/ RS, CCF, CCF1

COMMON/COUPLE/ ICUPLE, IEVEN, NT1, NT2

REAL*8 VARIABLES:

RS	Factor applied to radical velocity across the r = 0 cylindrical-geometry axis.
CCF	Cylindrical-geometry factor (CCF = 0.0 when NTSX is an even number and CCF = 1.0 when NTSX is an odd number).
CCF1	Cylindrical-geometry factor (CCF1 = 1.0 when NTSX is an even number and CCF1 = 0.0 when NTSX is an odd number).

INTEGER VARIABLES:

ICUPLE	Flag to indicate radial-direction convective coupling across the r = 0 cylindrical-geometry axis (not used).
IEVEN	Flag to indicate that the number of azimuthal sectors is an odd (0) or even (1) number.
NT1	Number of azimuthal sectors divided by 2 (NTSX/2).
NT2	NT1 + 1.

D.19. DAMPER.H

COMMON/DAMPER/ FIHT, IFRCR

REAL*8 VARIABLE:

FIHT	Wall-drag-coefficient adjustment factor (variable not used).

INTEGER VARIABLE:

IFRCR	Wall-drag evaluation option. 0 = no; 1 = yes.

D.20. DECAYC.H

COMMON/DECAYC/ FISPHI, FP235, FP238, FP239, QAVG, Q235, Q238, Q239, RANS, R239PF, TOPATE

COMMON/DECAYC/ IANS79

REAL*8 VARIABLES:

FISPHI	Fissions per initial fissile atom.

FP235 Fraction of reactor-core power from U^{235} fissions.
FP238 Fraction of reactor-core power from U^{238} fissions.
FP239 Fraction of reactor-core power from Pu^{239} fissions.
QAVG Average energy per fission.
Q235 Energy per fission from U^{235}.
Q238 Energy per fission from U^{238}.
Q239 Energy per fission from Pu^{239}.
RANS Multiplier applied to the ANS79 decay heat.
R239PF Atoms of U^{239} produced per fission.
TOPATE Four years in seconds units.

INTEGER VARIABLE:

IANS79 ANS79 decay-heat standard evaluation flag.
0 = not evaluated;
1 = evaluate the 69-group ANS79 decay-heat standard;
2 = evaluate the ANS79 decay-heat standard and the heavy-metal decay for U^{239} and Np^{239}.

D.21. DEFVAL.H

COMMON/DEFVAL/ ALPQ, HSTNQ, PQ, PAQ, QPPPQ, TLQ, TVQ, TWQ, VLQ, VVQ, CFZ3Q, HD3Q

COMMON/DEFVAL/ ISTOPT

REAL*8 VARIABLES:

ALPQ Default value for initial gas volume fractions input through NAMELIST and used to specify gas volume fractions when ISTOPT = 1 or 2.

HSTNQ Default value for initial HTSTR temperatures input through NAMELIST and used to specify the HTSTR temperatures when ISTOPT = 1 or 2.

PQ Default value for initial pressures input through NAMELIST and used to specify pressures when ISTOPT = 1 or 2.

PAQ Default value for initial noncondensable-gas partial pressures input through NAMELIST and used to specify noncondensable-gas partial pressures when STOPT = 1 or 2.

QPPPQ Default value for initial volumetric heat sources in flow channel walls input through NAMELIST and used to specify volumetric heat sources when ISTOPT = 1 or 2.

TLQ	Default value for initial liquid temperatures input through NAMELIST and used to specify liquid temperatures when ISTOPT = 1 or 2.
TVQ	Default value for initial gas temperatures input through NAMELIST and used to specify gas temperatures when ISTOPT = 1 or 2.
TWQ	Default value for initial wall temperatures input through NAMELIST and used to specify wall temperatures when ISTOPT = 1 or 2.
VLQ	Default value for initial liquid velocities input through NAMELIST and used to specify liquid velocities when ISTOPT = 1 or 2.
VVQ	Default value for initial gas velocities input through NAMELIST and used to specify gas velocities when ISTOPT = 1 or 2.
CFZ3Q	Default value for 3D VESSEL component additive loss coefficients input through NAMELIST and used to specify VESSEL additive loss coefficients when ISTOPT = 1 or 2.
HD3Q	Default value for 3D VESSEL component hydraulic diameters input through NAMELIST and used to specify VESSEL hydraulic diameters when ISTOPT = 1 or 2.

INTEGER VARIABLE:

ISTOPT	Option for defining thermal-hydraulic parameter default values through Namelist input (Namelist variable).

D.22. DETC.H

COMMON/DETC/ NDETC

INTEGER VARIABLE:

NDETC	Flag for generating debug printout from the outer-iteration cell-wise matrix definition.

D.23. DF1DC.H

COMMON/DF1DC/ ARDMIN, ARN, ARY, A11111, ALPST, C1A, C1AV, C2A, C2AV, CT, CTP, DVJP, FL1, FL2, FV1, FV2, HAVLV, QTP, R1L, R1V, R2L, R2V, SO1, SO2, SALT, SAVT, SSAC, SSE, SSMC, SSMOM, SSVC, SSVE, VJS, ZZZZZZ

COMMON/DF1DC/ IO1, IO2, IO3, IACC2, IBKS, ICME, ICORL, ICORU, IIO1, IIO2, IIO3, IL, IPHSEP, ISLB, ISRB, IVPVLV, JSTART, MSC, NC2, NSTG, NTEE, NJN, ISFLG, ICLFLG, LPINDX

REAL*8 VARIABLES:

ARDMIN	Minimum value of the difference between the flow-area ratios one mesh-cell distance from a junction interface with a PLENUM component and at the junction interface with a PLENUM component for flow from the PLENUM component.
ARN	No factor for applying flow-area ratios in the momentum-convection term. 0.0 = apply area ratios; 1.0 = do not apply area ratios.
ARY	Yes factor for applying flow-area ratios in the momentum-convection term. 1.0 = apply area ratios; 0.0 = do not apply area ratios.
A11111	Dummy variable that provides a known start to the COMMON block.
ALPST	The JCELL fluid gas volume fraction to be convected into the TEE side tube by the TEE offtake model.
C1A	Fraction of the liquid velocity at the left face of the TEE JCELL that contributes to momentum transfer into the TEE side tube.
C1AV	Fraction of the gas velocity at the left face of the TEE JCELL that contributes to momentum transfer into the TEE side tube.
C2A	Fraction of the liquid velocity at the right face of the TEE JCELL that contributes to momentum transfer into the TEE side tube.
C2AV	Fraction of the gas velocity at the right face of the TEE JCELL that contributes to momentum transfer into the TEE side tube.
CT	Momentum source coefficient.
CTP	AMAX1(0.0, COST).
DVJP	Pressure derivative of source velocity.
FL1	Temporary storage for liquid mass-flow corrections for mass-conservation checks at low-numbered cell face.
FL2	Temporary storage for liquid mass-flow corrections for mass-conservation checks at high-numbered cell face.

FV1	Temporary storage for gas mass-flow corrections for mass-conservation checks at low-numbered cell face.
FV2	Temporary storage for gas mass-flow corrections for mass-conservation checks at high-numbered cell face.
HAVLV	Temporary storage for the hydraulic diameter when the valve is open.
QTP	Total direct power input.
R1L	Coefficient of the SEPD or TEE side-tube coupled momentum-convection term at the left interface of JCELL for liquid.
R1V	Coefficient of the SEPD or TEE side-tube coupled momentum-convection term at the left interface of JCELL for gas.
R2L	Coefficient of the SEPD or TEE side-tube coupled momentum-convection term at the right interface of JCELL for liquid.
R2V	Coefficient of the SEPD or TEE side-tube coupled momentum-convection term at the right interface of JCELL for gas.
S01	Sign of IOU(1,current component).
S02	Sign of IOU(2,current component).
SALT	Source term to liquid for compressible work.
SAVT	Source term to gas for compressible work.
SSAC	Noncondensable-gas source.
SSE	Energy source.
SSMC	Mass source.
SSMOM	Momentum source to left-hand-cell boundary.
SSVC	Gas mass source.
SSVE	Gas energy source.
VJS	Source velocity.
ZZZZZZ	Dummy variable that provides a known end to the COMMON block.

INTEGER VARIABLES:

I01	ABS(IOU(1,current component)).
I02	ABS(IOU(2,current component)).
I03	IOU(3,current component) [always positive].
IACC2	Flag for PIPE to model an accumulator.
IBKS	Indicator for network solution.

ICME	Component index for referencing IOU array.
ICORL	Reactor-core region lower boundary.
ICORU	Reactor-core region upper boundary.
II01	I01 plus a displacement for the current loop.
II02	I02 plus a loop displacement.
II03	I03 plus a loop displacement.
IL	Loop number index.
IPHSEP	Phase-separation evaluation flag of the TEE offtake model.
ISLB	Left-hand boundary switch.
ISRB	Right-hand boundary switch.
IVPVLV	Interface number of the adjustable-valve flow area.
JSTART	Cell number at the left end of the 1D segment.
MSC	Cell number for the source terms.
NC2	Cell number that begins a SEPD or TEE side tube.
NSTG	Variable not used.
NTEE	Counter for a SPED or TEE.
NJN	Number of network matrix junctions.
ISFLG	Variable not used.
ICLFLG	Variable not used.
LPINDX	Loop index that indicates the loop in the system.

D.24. DIDDLE.H

COMMON/DIDDLE/ ALPCC, AFCT, ALPSHL, ALPSHU, ENCUT, ENFAC1, ENFAC2, ALW1, ALW2, FAREA1, FAREAH, FAREAV, FSE5, VRTCUT, ALPBCT, VECVCT, VECLCT, VINTF, ALPLVU, ALPLVL, CBMIN, CALV2, VRBCUT, VDRPF, VDRPMX, VLVCMX, ENMIN, SCINAN, TGRAV

COMMON/DIDDLE/ NIFSLB

REAL*8 VARIABLES:

ALPCC	Gas volume fraction that gives the minimum value for the bubble condensation rate.
AFCT	Area scaling for waves on inverted annular interface.
ALPSHL	Gas volume fraction below which the interface sharpener is off.
ALPSHU	Gas volume fraction above which the interface sharpener is on.
ENCUT	Minimum droplet-entrainment fraction.
ENFAC1	Scaling factor for minimum-entrainment velocity.

ENFAC2	Scaling factor for entrainment-correlation exponent.
ALW1	Gas volume fraction lower limit for transition from bubbly-slug (at ALW1= 0.5) to annular-mist (at ALW2 = 0.75) flow regimes.
ALW2	Gas volume fraction upper limit for transition from bubbly-slug (at ALW1 = 0.5) to annular-mist (at ALW2 = 0.75) flow regimes.
FAREA1	Scale factor for 1D plug-flow condensation area.
FAREAH	Scale factor for 3Dl separated plug-flow horizontal condensation area.
FAREAV	Scale factor for 3D separated plug-flow vertical condensation area.
FSE5	Scale factor for pool entrainment.
VRTCUT	Minimum allowed relative speed for computing interfacial coefficients (except during boiling).
ALPBCT	Lower bound on bubble gas volume fraction used to compute interfacial heat-transfer rates (and resulting Γ) when boiling.
VECVCT	Lowest allowed gas speed when computing interfacial coefficients.
VECLCT	Lowest allowed liquid speed when computing interfacial coefficients.
VINTF	Factor in expression for mean circulation speed in a moving drop.
ALPLVU	Highest value of the minimum adjacent gas volume fraction allowed for calculating a plug interfacial area.
ALPLVL	Lowest value of the maximum adjacent gas volume fraction allowed for calculating a plug interfacial area.
CBMIN	Minimum allowed mean bubble diameter.
CALV2	The liquid-side HTC for superheated drops.
VRBCUT	Minimum allowed relative velocity for computing ALV during boiling.
VDRPF	Scale factor in the expression for the limiting circulation velocity in a drop.
VDRPMX	Maximum allowed internal circulation velocity in a drop.
VLVCMX	Maximum liquid velocity used for the condensation ALV correlation.
ENMIN	Variable not used.
SCINAN	Scale factor for inverted annular drag.

TGRAV	Absolute value of GRAV (0.75) above which horizontal stratified flow cannot exist.

INTEGER VARIABLE:

NIFSLB	If nonzero, then slabs should be used to test for inverted annular flow.

D.25. DIDDLH.H

COMMON/DIDDLH/ ALP2, ALP3, ALPBR, ALPCUT, ALPCTR, ALOW, AUP, AFLML, AFLMU, ALPAG, ALPDF, FLILER, FLILES, FBER, FBEX, FDFHL, HGF, HGVMN, FACTHL3, FDALVA, FREQ1, FREQ2, FUDGE1, FUDGE2

COMMON/DIDDLH/ LIMFLG, IHTAV, IHTCN, NSHTCN

REAL*8 VARIABLES:

ALP2	Gas volume fraction (1.0) above which the gas is in forced convection.
ALP3	Gas volume fraction (0.9999) above which there is no liquid heat transfer.
ALPBR	Gas volume fraction (0.99) above which liquid convection is linearly faired off and gas convection is faired on.
ALPCUT	Gas volume fraction (0.98) above which nucleate boiling is not permitted (if other criteria are met).
ALPCTR	Gas volume fraction (0.995).
ALOW	Lowest value of the gas volume fraction in adjacent cells.
AUP	Highest value of the gas volume fraction in adjacent cells.
AFLML	Gas volume fraction below which Bromley film boiling contributes fully to the liquid.
AFLMU	Gas volume fraction above which no Bromley coefficient is added to the liquid.
ALPAG	Gas volume fraction at the agitated-to-post-agitated inverted-annular flow transition boundary.
ALPDF	Gas volume fraction describing the beginning of the highly dispersed inverted-annular flow.
FLILER	Constant used to adjust the wall-to-liquid HTC obtained by modified Bromley correlation in reflood.
FLILES	Same as FLILER for non-reflood cases.
FBER	Variable not used.

FBEX	Power of $(1 - \alpha)$ weighting of the Bromley correlation.
FDFHL	A scaling factor for the wall-to-droplet heat-transfer correlation.
HGF	Function of nucleate-boiling heat transfer, which contributed to subcooled boiling.
HGVMN	Cutoff velocity for condensation used to suppress subcooled nucleate boiling.
FACTHL3	Power that (ALOFF2–ALP)/(ALOFF2–ALPX2) is raised to.
FDALVA	Multiplier of DALVA(J), which is the rate of change of ALW with respect to gas volume fraction and is currently set to zero.
FREQ1	Time-constant constraint frequency for the maximum increase in interfacial heat-transfer and drag coefficients.
FREQ2	Time-constant constraint frequency for the maximum decrease in interfacial heat-transfer and drag coefficients.
FUDGE1	Time-constant constraint factor of maximum increase when the timestep size is 1.0/FREQ1.
FUDGE2	Time-constant constraint factor of maximum decrease when the timestep size is 1.0/FREQ2.

INTEGER VARIABLES:

LIMFLG	Flag for evaluating time-constant constraint of the evaporation and condensation rate coefficients. 0 = no; 1 = yes.
IHTAV	Variable is normally 1. When IHTAV is 0, there is no time averaging of HTCs.
IHTCN	Variable is normally 0. When IHTCN is 1, HTCs are forced to remain constant.
NSHTCN	Variable is normally 10 000 000. I f NSTEP > NSHTCN, then IHTCH is set to 1 (for debugging only).

D.26. DIDDLI.H

COMMON/DIDDLI/ SMIVX

COMMON/DIDDLI/ NSCOOL, IIABK

REAL*8 VARIABLE:

SMIVX	Constant value 1.5 (variable not used).

INTEGER VARIABLES:

NSCOOL	Flag (when having its default value of 1) that constrains the subcooled boiling heat flux to not exceed the wall heat flux to the liquid.
IIABK	Constant value 1 (variable not used).

D.27. DIMNSION.H

COMMON/DIMEN/ IFREE, JNVSSL, KVEL1T, KVEL2T, KVEL3T, LAST, LDIM, LENBD, LENDIM, LENTBL, LFREE, LLAST, LM1DP, LNLDPV, LSTART, LVER, MDIM, MEMFLG, MOFF, NCOMP, NCOMPT, NHTSTR, NJNMX, NJNT, NJUN, NLOOPS, NMVSSL, NPX, NSTGJ, NTHM, NUMTCR, NVCON, NVELX, NVELY, NVELZ, NWRDA

INTEGER VARIABLES:

IFREE	First free location in the dynamic computer-memory space.
JNVSSL	Maximum number of VESSEL junctions in a loop.
KVEL1T	Order of the r- or x-direction stabilizer motion-equation VESSEL matrix.
KVEL2T	Order of the θ- or y-direction stabilizer motion-equation VESSEL matrix.
KVEL3T	Order of the z-direction stabilizer motion-equation VESSEL matrix.
LAST	Last location in the dynamic computer-memory space.
LDIM	Maximum size order of the capacitance matrix.
LENBD	Length of boundary-data array for each junction.
LENDIM	Variable that dimensions the component variable-length tables.
LENTBL	Length of the fixed-length table.
LFREE	First free location in the computer-memory space.
LLAST	Last location in the computer-memory space.
LM1DP	Pointer variable for array that stores M1D input-data values for HTSTR components.
LNLDPV	Pointer variable for the network matrix equation right-hand side vector.
LSTART	First free location in the computer-memory space.
LVER	Location of version information data.
MDIM	Maximum order of the banded VESSEL matrix.
MEMFLG	Flag for monitoring dynamic computer-memory expansion.

MOFF	Array row number of the main diagonal elements from the banded VESSEL matrix.
NCOMP	Number of components.
NCOMPT	Total number of components.
NHTSTR	Total number of HTSTR components (Namelist variable).
NJNMX	Maximum number of network junctions.
NJNT	Total number of network junctions for all loops.
NJUN	Number of junctions.
NLOOPS	Number of 1D loops in the system model.
NMVSSL	Number of VESSELs.
NPX	Number of pointers in the PTRS COMMON block.
NSTGJ	Variable not used.
NTHM	Number of elements per cell in the DRIV array.
NUMTCR	Number of title cards.
NVCON	Total number of VESSEL connections.
NVELX	Order of the r- or x-direction stabilizer motion equation VESSEL matrix.
NVELY	Order of the θ- or y-direction stabilizer motion equation VESSEL matrix.
NVELZ	Order of the z-direction stabilizer motion equation VESSEL matrix.
NWRDA	Size of the A array under *IF DEF,ASIZE.

D.28. DLIMIT.H

COMMON/DLIM/	DELAMX, DELCMX, DELDMX, DELEMX, DELPMX, DELRMX, DELVMX, DELXMX, DPRMC, DTBKUP, DTLMC, DTRMC, DTSMC, DTVMC, FPMAX, FXMAX, GXMAX, SVMAXT, SVMXT3, VMERMX, VRMX
COMMON/DLIM/	MAXIT, MAXITB, MAXITC, MINDT, NLIM, NLIM2, NOBKUP
DIMENSION	NLIM(9), NLIM2(6)

REAL*8 VARIABLES:

DELAMX	Timestep limit caused by gas-volume-fraction change.
DELCMX	Timestep limit caused by maximum changes in pressures and temperatures.

DELDMX	Timestep limit caused by numerical considerations in the ROD and SLAB heat-transfer calculation.
DELEMX	Timestep limit caused by VESSEL mass errors.
DELPMX	Timestep limit that results in a maximum 10% change in reactor-core power.
DELRMX	Timestep limit caused by final value of the percentage variation in pressure from iteration to iteration.
DELVMX	Material Courant stability limit (computed only in VESSELs).
DELXMX	Timestep limit that results in the maximum allowed adjustment of VALVE components.
DPRMC	Maximum fractional change (0.5) in the pressure to control the timestep size.
DTBKUP	Timestep limit defined by DELPMX or DELXMX when a back up calculation is required after the prep-stage calculation.
DTLMC	Maximum change (20.0 K, 36.0 °F) in the liquid temperature to control the timestep size.
DTRMC	Maximum change (20.0 K, 36.0 °F) in the HTSTR-component wall inner- and outer-surface temperatures to control the timestep size.
DTSMC	Maximum change (20.0 K, 36.0 °F) in the 1D hydraulic-component wall outer-surface temperature to control the timestep size.
DTVMC	Maximum change (25.0 K, 45.0 °F) in the gas temperature to control the timestep size.
FPMAX	Maximum fractional change (0.1) in reactor-core power per timestep.
FXMAX	VALVE-adjustment algorithm parameter (0.4).
GXMAX	Minimum fractional change (0.05) in the VALVE maximum flow-area fraction change over a timestep.
SVMAXT	Reciprocal of the material Courant timestep size in all 1D hydraulic components.
SVMXT3	Reciprocal of the material Courant timestep size in all 3D VESSEL components.
VMERMX	VESSEL-component maximum fractional (10000.0) mass error to control the timestep size.

VRMX	Maximum fraction change (0.1) in the pressure to control the timestep size.

INTEGER VARIABLES:

MAXIT	Switch to continue the TRAC-P calculation without reducing the timestep size when outer-iteration convergence is not satisfied after OITMAX or SITMAX outer iterations.
MAXITB	Previous converged outer-iteration number below which the previous timestep size is increased by the multiplier DDI.
MAXITC	Previous converged outer-iteration number OITNO above which the previous timestep size is decreased by the multiplier MAXITC/OITNO.
MINDT	Switch to continue the TRAC-P calculation without reducing the timestep size below DTMIN, which would stop the calculation.
NLIM	Array that stores the number of timesteps that were constrained by each of the timestep limits since the last small or large edit.
NLIM2	Array that stores the number of timesteps that were constrained by each of the six different timestep limits defining DELCMX since the last small or large edit (the sum of all six NLIM2(I) equals NLIM(5), which is the number of times DELCMX controls the timestep size).
NOBKUP	Switch to continue the TRAC-P calculation without doing any timestep-reduction backup evaluations.

D.29. DMPCK.H

COMMON/DMPCK/ LVCK

INTEGER VARIABLE:

LVCK	Summed number of values over the VESSEL component that were written to the TRCDMP dump file (summed by subroutine DLEVEL but not used).

D.30. DMPCTRL.H

COMMON/CTRLDP/	DMPINT, LTDUMP, TDUMP
COMMON/CTRLDP/	DMPFLG, ICTRLD, NSDO
REAL*8	DMPINT, LTDUMP, TDUMP
INTEGER	DMPFLG, ICTRLD(8), NSDO

REAL*8 VARIABLES:

DMPINT	Dump interval for time domain.
LTDUMP	Cpu time when last data dump was taken.
TDUMP	Calculation time when next data dump will be taken.

INTEGER VARIABLES:

DMPFLG	Flag that signals whether the dump output file has been initialized. 0 = uninitialized; 1 = initialized.
ICTRLD	Array that contains buffering information about the dump output file.
NSDO	Timestep number of the last completed data dump.

D.31. DTINFO.H

COMMON/DTINFO/ DDI, DELTIT, DELTNC, DELV3X, DPRSV, DTDDI, DTFL, DTLSV, DTRSV, DTSSV, DTVSV, PRMXSV, SVDELC, TLMXSV, TRMXSV, TSMXSV, TVMXSV

COMMON/DTINFO/ ICSAVE, ICSRC, IDCDAL, IDCDAU, IDCOAL, IDCOAU, ITRPDT, IZDAL, IZDAU, IZOAL, IZOAU, JCSAVE, KCCMXT, KCSAVE, NCFACE

REAL*8 VARIABLES:

DDI	Timestep-size maximum-increase factor.
DELTIT	Timestep size from the iteration-count logic that results in scaling back the timestep size.
DELTNC	Timestep size from the iteration-count logic that results in no change to the timestep size.
DELV3X	Timestep size from the 3D material Courant limit in the VESSEL components.
DPRSV	Pressure change DELCMX limiter.
DTDDI	Timestep size from the iteration-count logic that results in an increase in the timestep size by the factor DDI.
DTFL	Value of 0.5*DELT used in the DELCMX timestep-size control logic.
DTLSV	Liquid-temperature change DELCMX limiter.
DTRSV	HTSTR-temperature change DELCMX limiter.
DTSSV	1D component wall-temperature change DELCMX limiter.

DTVSV	Gas-temperature change DELCMX limiter.
PRMXSV	Maximum pressure change used in the DELCMX logic.
SVDELC	Current value of DELCMX (before 0.5*DELT minimum applied).
TLMXSV	Maximum liquid-temperature change used in the DELCMX logic.
TRMXSV	Maximum HTSTR wall-temperature change used in the DELCMX logic.
TSMXSV	Maximum 1D component wall-temperature change used in the DELCMX logic.
TVMXSV	Maximum gas-temperature change used in the DELCMX logic.

INTEGER VARIABLES:

ICSAVE	Radial-r or x direction cell index for the 3D material Courant limit timestep-size diagnostic edit.
ICSRC	Flag to indicate that the 3D material Courant limit is at a 1D source connection to the VESSEL which is used for the timestep-size diagnostic edit.
IDCDAL	Cell number in the 3D level for gas-volume-fraction change timestep-size control variable DAL.
IDCDAU	Cell number in the 3D level for gas-volume-fraction change timestep-size control variable DAU.
IDCOAL	Cell number in the 3D level for gas-volume-fraction change timestep-size control variable OAL.
IDCOAU	Cell number in the 3D level for gas-volume-fraction change timestep-size control variable OAU.
ITRPDT	Flag to indicate that trip-controlled timestep-size logic was used in subroutine TRPSET.
IZDAL	3D level number for gas-volume-fraction change timestep-size control variable DAL.
IZDAU	3D level number for gas-volume-fraction change timestep-size control variable DAU.
IZOAL	3D level number for gas-volume-fraction change timestep-size control variable OAL.
IZOAU	3D level number for gas-volume-fraction change timestep-size control variable OAU.
JCSAVE	Azimuthal-θ or y direction cell index for the 3D material Courant limit timestep-size diagnostic edit.

KCCMXT	Cell-face-index absolute value for 1D material Courant limit logic.
KCSAVE	Axial-z direction cell index for the 3D material Courant limit timestep-size diagnostic edit.
NCFACE	3D cell-face number for the material Courant limit with a positive or negative value for a 1D source connection and a positive value for a 3D cell interface which is used for the timestep-size diagnostic edit.

D.32. DTPC.H

COMMON/DTPC/ DCNFACH, DCNFACL, DCSF1D, DCSF3DH, DDDI, DDPRMC, DDTLMC, DDTRMC, DDTSMC, DDTVMC, DFPMAX, DFXMAX, DGXMAX, DVMERMX, DVRMX, DXDAL, DXDAU, DXOAL, DXOAU

COMMON/DTPC/ IDTPC, NMAXIT, NMAXITB, NMAXITC, NMINDT, NNOBKUP

REAL*8 VARIABLES:

DCNFACH	Constant 0.8 used to initialize CNFACH in subroutine INPUT.
DCNFACL	Constant 0.75 used to initialize CNFACL in subroutine INPUT.
DCSF1D	Constant 1000.0 used to initialize CSF1D in subroutine INPUT.
DCSF3DH	Constant 1000.0 used to initialize CSF3DH in subroutine INPUT.
DDDI	Constant 1.05 used to initialize DDI in subroutine INPUT.
DDPRMC	Constant 0.5 used to initialize DPRMC in subroutine INPUT.
DDTLMC	Constant 20.0 used to initialize DTLMC in subroutine INPUT.
DDTRMC	Constant 20.0 used to initialize DTRMC in subroutine INPUT.
DDTSMC	Constant 20.0 used to initialize DTSMC in subroutine INPUT.
DDTVMC	Constant 25.0 used to initialize DTVMC in subroutine INPUT.
DFPMAX	Constant 0.1 used to initialize FPMAX in subroutine INPUT.
DFXMAX	Constant 0.4 used to initialize FXMAX in subroutine INPUT.
DGXMAX	Constant 0.05 used to initialize GXMAX in subroutine INPUT.
DVMERMX	Constant 10000.0 used to initialize VMERMX in subroutine INPUT.
DVRMX	Constant 0.1 used to initialize VRMX in subroutine INPUT.
DXDAL	Constant 0.2 used to initialize XDAL in subroutine INPUT.
DXDAU	Constant 0.05 used to initialize XDAU in subroutine INPUT.
DXOAL	Constant 1.0 used to initialize XOAL in subroutine INPUT.

DXOAU	Constant 1.0 used to initialize XOAU in subroutine INPUT.

INTEGER VARIABLES:

IDTPC	Option for inputting timestep-control parameter constants and switch variables to prevent timestep reductions (Namelist variable).
NMAXIT	Constant 0 used to initialize MAXIT in subroutine INPUT.
NMAXITB	Constant 4 used to initialize MAXITB in subroutine INPUT.
NMAXITC	Constant 5 used to initialize MAXITC in subroutine INPUT.
NMINDT	Constant 0 used to initialize MINDT in subroutine INPUT.
NNOBKUP	Constant 0 used to initialize NOBKUP in subroutine INPUT.

D.33. EDIFF.H

COMMON/EDIFF/ NTLTST, JTLTST, LTLTST, NTVTST, JTVTST, LTVTST, NTMTST, JTMTST, LTMTST, NPRTST, JPRTST, LPRTST, NDAMX, KDAMX, LDAMX, IDIAG2

INTEGER VARIABLES:

NTLTST	Component number that controls the timestep size due to the liquid-temperature change limit.
JTLTST	Horizontal-plane cell number of component NTLTST that controls the timestep size due to the liquid-temperature change limit.
LTLTST	Axial-z cell number of component NTLTST that controls the timestep size due to the liquid-temperature change limit.
NTVTST	Component number that controls the timestep size due to the gas-temperature change limit.
JTVTST	Horizontal-plane cell number of component NTVTST that controls the timestep size due to the gas-temperature change limit.
LTVTST	Axial-z cell number of component NTVTST that controls the timestep size due to the gas-temperature change limit.
NTMTST	Component number that controls the timestep size due to the metal-temperature change limit.
JTMTST	Structure node number of component NTMTST that controls the timestep size due to the "metal"-temperature change limit.
LTMTST	Axial-z cell number of component NTMTST that controls the timestep size due to the "metal"-temperature change limit.

NPRTST	Component number that controls the timestep size due to the pressure change limit.
JPRTST	Horizontal-plane cell number of component NPRTST that controls the timestep size due to the pressure change limit.
LPRTST	Axial-z cell number of component NPRTST that controls the timestep size due to the pressure change limit.
NDAMX	Variable not used.
KDAMX	Variable not used.
LDAMX	Variable not used.
IDIAG2	Flag that allows skipping of certain diagnostics generated in subroutine NEWDLT by the IDIAG option (default is on).

D.34. ELVKF.H

COMMON/ELVKF/ IELV, IINL, IKFAC, MWFL, MWFV

INTEGER VARIABLES:

IELV	Option for inputting cell-centered elevations to the gravity array (Namelist variable).
IINL	Index for the two passes through INIT.
IKFAC	Option for inputting K-factors to the additive form-loss coefficient array (Namelist variable).
MWFL	Option for inputting wall-to-liquid wall-friction multiplier factors (Namelist variable).
MWFV	Option for inputting wall-to-gas wall-fraction multiplier factors (Namelist variable).

D.35. EMOT.H

COMMON/EMOT/ CNFACH, CNFACL, CSF, CSF1D, CSF3D, CSF3DL, CSF3DH, FNCIF

COMMON/EMOT/ IVMN, IVMX, JIV, NOLDV

REAL*8 VARIABLES:

CNFACH	Multiplier (0.8) applied to the 3D VESSEL-component material Courant limit that, when the timestep size equals or is greater than this timestep limit, starts the evaluation of the 3DSETS method when Namelist variable NOSETS = 0.
CNFACL	Multiplier (0.75) applied to the 3D VESSEL-component material Courant limit that, when the timestep size equals or is less than

this timestep limit, stops the evaluation of the 3DSETS method when Namelist variable NOSETS = 0.

CSF	A factor (1.0) applied to CSF1D and CSF3D to define the maximum material Courant number.
CSF1D	Maximum material Courant number (1000.0) for the 1D hydraulic components.
CSF3D	Maximum material Courant number (1000.0) for the 3D VESSEL components.
CSF3DL	Maximum material Courant number (1.0) for the 3D VESSEL components when the SET3D equations are not evaluated.
CSF3DH	Maximum material Courant number (1000.0) for the 3D VESSEL components when the SET3D equations are evaluated.
FNCIF	Constant 0.7 (variable not used).

INTEGER VARIABLES:

IVMN	Minimum timestep number for debug outputting interface JIV velocities.
IVMX	Maximum timestep number for debug outputting interface JIV velocities.
JIV	Mesh-cell interface number for debug outputting gas and liquid tilde and basic velocities in subroutine TF1DS1.
NOLDV	Flag for setting the beta factor in the momentum-convection term to zero. 0 = no; 1 = yes.

D.36. ERRCON.H

COMMON/ERRCON/	ANTEST, ATEST1, DARA, DARL, DARV, DDVL, DDVV, DTLL, DTLLM, DTLU, DTLUM, DTVL, DTVLM, DTVU, DTVUM, TIMDL, TIMDU, TSDLT, TSDUT
COMMON/ERRCON/	IATEST, ICHGA, ILREIT, IPTEST, IVTEST, JATEST, JDARA, JDARL, JDARV, JDDVL, JDDVV, JDTLL, JDTLU, JDTVL, JDTVU, JPTEST, JVTEST, KPTEST, NDARA, NDARL, NDARV, NDDVL, NDDVV, NDTLL, NDTLU, NDTVL, NDTVU, NPTEST, NSDL, NSDU, TSDLS, TSDUS
INTEGER	TSDLS, TSDUS

REAL*8 VARIABLES:

ANTEST	End-of-timestep gas volume fraction that is outside its 0.0 to 1.0 value range in mesh cell JATEST of component IATEST.
ATEST1	Beginning-of-timestep gas volume fraction in mesh cell JATEST of component IATEST.
DARA	Maximum change in $\alpha\rho_a$.
DARL	Measure of the maximum difference in $(1-\alpha)\rho_\ell$ between the basic and stabilizer steps.
DARV	Measure of the maximum difference in $\alpha\rho_g$ between the basic and stabilizer steps.
DDVL	Measure of the maximum difference in V_ℓ between the basic and stabilizer steps.
DDVV	Measure of the maximum difference in V_g between the basic and stabilizer steps.
DTLL	Largest decrease in T_ℓ from the current iteration.
DTLLM	DTVLM and DTLLM are limits on DTVL and DTLL beyond which another iteration must be performed.
DTLU	Largest increase in T_ℓ from the current iteration.
DTLUM	DTVLM and DTLLM are limits on DTVL and DTLL beyond which another iteration must be performed.
DTVL	Largest decrease in gas temperature in a given iteration.
DTVLM	DTVLM and DTLLM are limits on DTVL and DTLL beyond which another iteration must be performed.
DTVU	Largest increase in T_g from the current iteration.
DTVUM	DTVLM and DTLLM are limits on DTVL and DTLL beyond which another iteration must be performed.
TIMDL	If TIMDL $\leq$ TIMET $\leq$ TIMDU for the problem time, details of DARV, etc., should be output.
TIMDU	If TIMDL $\leq$ TIMET $\leq$ TIMDU for the problem time, details of DARV, etc., should be output.
TSDLT	Starting time at which detailed timestep-diagnostic information on the logic used to evaluate the timestep size is output to file TRCMSG.

TSDUT	Ending time at which detailed timestep-diagnostic information on the logic used to evaluate the timestep size is output to file TRCMSG.

INTEGER VARIABLES:

IATEST	Component number with an out-of-range gas-volume-fraction value.
ICHGA	Flag to print the maximum gas-volume-fraction changes to the TRCMSG file.
ILREIT	Flag that allows reiteration messages when equation set changes.
IPTEST	VESSEL radial-r or x direction mesh-cell number having maximum $\lvert \delta p/p \rvert$.
IVTEST	Component number having a velocity that changed its numerical sign during the last outer iteration.
JATEST	Mesh-cell number in component IATEST where the gas volume fraction has an out-of-range value.
JDARA	Cell number where DARA occurred.
JDARL	Cell number where DARL occurred.
JDARV	Cell number where DARV occurred.
JDDVL	Cell number where DDVL occurred.
JDDVV	Cell number where DDVV occurred.
JDTLL	Cell number where DTLL occurred.
JDTLU	Cell number where DTLU occurred.
JDTVL	Cell number where DTVL occurred.
JDTVU	Cell number where DTVU occurred.
JPTEST	VESSEL azimuthal-θ or y direction mesh-cell number or 1D component mesh-cell number with maximum $\lvert \delta p/p \rvert$.
JVTEST	Mesh-cell interface number in component IVTEST with a velocity that changed numerical sign during the last outer iteration.
KPTEST	VESSEL axial-z direction mesh-cell number with maximum $\lvert \delta p/p \rvert$.
NDARA	Component number where DARA occurred.
NDARL	Component number where DARL occurred.
NDARV	Component number where DARV occurred.
NDDVL	Component number where DDVL occurred.
NDDVV	Component number where DDVV occurred.

NDTLL	Component number where DTLL occurred.
NDTLU	Component number where DTLU occurred.
NDTVL	Component number where DTVL occurred.
NDTVU	Component number where DTVU occurred.
NPTEST	Component number with maximum $\lvert \delta p/p \rvert$.
NSDL	If NSDL $\leq$ NSTEP $\leq$ NSDU for the timestep number, a detailed diagnostic of DARV, etc., should be output to the TRCOUT file and IDIAG = 3 diagnostics to the TRCMSG file.
NSDU	If NSDL $\leq$ NSTEP $\leq$ NSDU for the timestep number, a detailed diagnostic of DARV, etc., should be output to the TRCOUT file and IDIAG = 3 diagnostics to the to TRCMSG.
TSDLS	First timestep number where detailed timestep-diagnostic information on the logic used to evaluate the timestep size is output to file TRCMSG.
TSDUS	Last timestep number where detailed timestep-diagnostic information on the logic used to evaluate the timestep size is output to file TRCMSG.

D.37. FILM.H

COMMON/FILM/ CONFLM, FILMU, FILML, XFDCON, FDMAX, ALPF1, FFUNH1, ALPF2, FFUNH2, XPFUNH

REAL*8 VARIABLES:

CONFLM	Constant used in film thickness calculation.
FILMU	Upper bound on film thickness.
FILML	Lower bound on film thickness.
XFDCON	Multiplier on wet-wall film drag.
FDMAX	Factor indicating how much bigger film drag may be than wall drag.
ALPF1	Gas volume fraction when factor indicating cross-channel cold-wall effect begins.
FFUNH1	Factor indicating minimum cross-channel cold-wall effect.
ALPF2	Gas volume fraction when factor indicating cross-channel cold-wall effect reaches full on.
FFUNH2	Factor indicating maximum cross-channel cold-wall effect.
XPFUNH	decay power for cross-channel cold-wall effect.

D.38. FIXEDLT.H

COMMON/FLTAB/	HTLSCI, HTLSCO, PINTEG, TITLE(4), TYPE
COMMON/FLTAB/	ID, IREST, LENARR, LENFV, LENFV2, LENPTR, LENVLT, LEXTRA, LFV, LFVN, ICFLG, NUMBM1, NUMBM2, NUMBM3, NUMBN1, NUMBN2, NUMBN3, NODES, NUM, NRVLT, NCELLT

REAL*8 VARIABLES AND PARAMETER CONSTANTS:

HTLSCI	HTLSCIIND=1	Component-wall inner-surface heat-transfer coefficient.
HTLSCO	HTLSCOIND=2	Component-wall outer-surface heat-transfer coefficient.
PINTEG	PINTEGIND=3	Energy (such as resistance heating) into a 1D hydraulic component wall.
TITLE(1)	TITLEIND=4	Component description.
TITLE(2)	at 5	
TITLE(3)	at 6	
TITLE(4)	at 7	
TYPE	TYPEIND=8	Component-type number.

INTEGER VARIABLES AND PARAMETER CONSTANTS:

ID	IDIND=–2	Component identification number.
IREST	IRESTIND=–3	Component restart indicator.
LENARR	LENARRIND=–4	Length of array block.
LENFV	LENFVIND=–5	Length of fundamental variables.
LENFV2	LENFV2IND=–6	Length of fundamental variables for which old-time and new-time values are the same at the start of the OUTER phase.
LENPTR	LENPTRIND=–7	Length of the pointer table.
LENVLT	LENVLTIND=–8	Length of the variable-length table.
LEXTRA	LEXTRAIND=–9	Length of nonstandard dump for components.
LFV	LFVIND=–10	Relative position of old-time fundamental variables.
LFVN	LFVNIND=–11	Relative position of new-time fundamental variables.
ICFLG	ICFLGIND=–1	Cell-edge choked-flow model option.

NUMBM1	NUMBM1IND=–13	Index to access large numerals for printing component NUM and IORDER.
NUMBM2	NUMBM2IND=–14	Index to access large numerals for printing component NUM and IORDER.
NUMBM3	NUMBM3IND=–15	Index to access large numerals for printing component NUM and IORDER.
NUMBN1	NUMBN1IND=–16	Index to access large numerals for printing component NUM and IORDER.
NUMBN2	NUMBN2IND=–17	Index to access large numerals for printing component NUM and IORDER.
NUMBN3	NUMBN3IND=–18	Index to access large numerals for printing component NUM and IORDER.
NODES	NODESIND=–19	Number of heat-transfer nodes.
NUM	NUMIND=–20	Component number.
NRVLT	NRVLTIND=–21	Number of real variables in each component's variable-length table COMMON block.
NCELLT	NCELLTIND=–12	Total number of cells.

D.39. FIXUM.H

COMMON/FIXUM/ NOAIR, NSMEC, NTHRMC, NVTC

INTEGER VARIABLES:

NOAIR	Variable that turns off noncondensable gas calculations.
NSMEC	Variable that turns off stabilizer mass and energy equations.
NTHRMC	Variable that turns off (debugs) basic equation set.
NVTC	Variable that turns off stabilizer motion equations.

D.40. FLUID.H

COMMON/FLUID/ LH2O, ID2O

LOGICAL LH2O

LOGICAL VARIABLE:

LH2O	Logic flag for fluid properties. .TRUE. = H2O properties; .FALSE. = D2O properties.

INTEGER VARIABLE:

ID2O	Option for fluid properties.
	1 = H2O properties;
	2 = D2O properties.

D.41. GENPT.H

PARAMETER	(IPTSIZ=320)
INTEGER	PT(IPTSIZ)
COMMON/PLTAB/	PT

INTEGER VARIABLE:

PT	Graphics pointer table.

D.42. GRAPHICS.H

DIMENSION	ICTRLG(8)
COMMON/GRAPH/	EDINT, GFINT, SEDINT, TEDIT, TGRAF, TSEDIT
COMMON/GRAPH/	IBUFF, ICTRLG, IPKG, KP, LCAT, LCMGCT, LENCAT, NCTX, NSGO, NWTX

REAL*8 VARIABLES:

EDINT	Large-edit interval for the time domain.
GFINT	Graphics-edit interval for the time domain.
SEDINT	Small-edit interval for the time domain.
TEDIT	Time of next large edit.
TGRAF	Time of next graphics edit.
TSEDIT	Time of next small edit.

INTEGER VARIABLES:

IBUFF	Length of graphics buffer.
ICTRLG	Array that contains buffering information about the graphics output file.
IPKG	Graphics file packing density.
KP	Pointer in graphics catalog block.
LCAT	Address of graphics catalog in SCM.
LCMGCT	Address of graphics catalog in LCM.
LENCAT	Number of words in each catalog entry.
NCTX	Number of graphics catalog entries.
NSGO	Timestep number of last completed graphics edit.
NWTX	Number of words written to disk per graphics edit.

D.43. H2FDBK.H

COMMON/H2FDBK/ IH2SRC

INTEGER VARIABLE:

IH2SRC	Hydrogen-source flag (Namelist variable; when nonzero, TRAC sets Namelist variables IGAS = 2 and NOAIR = 0).

D.44. HPSSD.H

COMMON/HPSSD/ NIC(200), IIC, LDCINF, LDCLOC, LDCOUF, LDCPWI, LDCPWO, LDDINF, LDDLOC, LDDOUF, LDDPWI, LDDPWO, LMASI, LMASM, LMASN, LMAST, LPHM, LPMVL, LPMVV, LPOWER, LPP, LPPA, LPTL, LPTV, NFPI, NPATHS, NTPI

INTEGER VARIABLES:

NIC	Component ID numbers that are not being initialized by the hydraulic-path steady-state initialization procedure.
IIC	Total number of components that are not being initialized by the hydraulic-path steady-state initialization procedure.
LDCINF	A-array pointer variable for the 1D component ID number of the hydraulic-path inflow location of the hydraulic path.
LDCLOC	A-array pointer variable for the 1D component ID number of the hydraulic-path condition location defining the hydraulic condition.
LDCOUF	A-array pointer variable for the 1D component ID number of the hydraulic-path outflow location of the hydraulic path.
LDCPWI	A-array pointer variable for the 1D component ID number of the hydraulic-path inflow location of the first cell having a heat source or sink.
LDCPWO	A-array pointer variable for the 1D component ID number of the hydraulic-path inflow location of the last cell having a heat source or sink.
LDDINF	A-array pointer variable for the interface number of the hydraulic-path inflow location of the hydraulic path.
LDDLOC	A-array pointer variable for the interface number of the hydraulic-path condition location defining the hydraulic condition.

LDDOUF A-array pointer variable for the interface number of the hydraulic-path outflow location of the hydraulic path

LDDPWI A-array pointer variable for the cell number of the hydraulic-path inflow location of the first cell having a heat source or sink.

LDDPWO A-array pointer variable for the cell number of the hydraulic-path inflow location of the last cell having a heat source or sink.

LMASI A-array pointer variable for the input-specified coolant mass in each hydraulically coupled region of the system model.

LMASM A-array pointer variable for the two-phase coolant mass based on $\alpha = \alpha_m$ and $T_G = T_{sat} = T_L$ in each hydraulically coupled region of the system model.

LMASN A-array pointer variable for the two-phase coolant mass based on $\alpha = \alpha_n$ and $T_G = T_{sat} = T_L$ in each hydraulically coupled region of the system model.

LMAST A-array pointer variable for the coolant mass based on $\alpha = 0, \alpha_n$, or 1 and T_G and T_L in each hydraulically coupled region of the system model.

LPHM A-array pointer variable for the initial liquid mass flow or velocity at the location defining the hydraulic condition.

LPMVL A-array pointer variable for the initial liquid mass flow or velocity at the interface location defining the hydraulic condition.

LPMVV A-array pointer variable for the initial gas mass flow or velocity at the interface location defining the hydraulic condition.

LPOWER A-array pointer variable for the total heat source or sink power between and including cells A(LDDINF) to A(LDDOUF).

LPP A-array pointer variable for the total pressure in all cells along the hydraulic path (when NTPI = 0).

LPPA A-array pointer variable for the noncondensable-gas pressure in all cells along the hydraulic path (when NTPI = 0).

LPTL A-array pointer variable for the initial liquid temperature at the donor-cell location defining the hydraulic condition.

LPTV A-array pointer variable for the initial gas temperature at the donor-cell location defining the hydraulic condition.

NFPI Mass-flow or velocity input option.
0 = input liquid and gas mass flows;

1 = input liquid and gas velocities.

NPATHS — Number of 1D hydraulic paths defined in the system model.

NTPI — Total pressure and noncondensable-gas pressure input option.
- 0 = input pressures for all hydraulic-path cells;
- 1 = define the hydraulic-condition donor-cell pressures for all hydraulic-path cells;
- 2 = define pressure from the component input data.

INTEGER IHPSIZE, AHPSIZE
PARAMETER (IHPSIZE=1000, AHPSIZE=1000)
COMMON/HPCOM/ AHP(AHPSIZE), IHP(IHPSIZE), IFREEHP

REAL*8 VARIABLE:

AHP — HPSSD real variable container array.

INTEGER VARIABLES:

IHP — HPSSD integer variable container array.

IFREEHP — Pointer for the next element of free storage space in the container A array for storing the masi, masm, masn, and mast arrays.

D.45. HTCAV.H

COMMON/HTCAV/ FHTCU, FHTCL, OWHTD

REAL*8 VARIABLES:

FHTCU — Maximum factor of increase (2.0) in the liquid and gas heat-transfer coefficients.

FHTCL — Minimum factor of decrease (0.0) in the liquid and gas heat-transfer coefficients.

OWHTD — Fraction (0.55) of the previous time-averaged liquid or gas heat-transfer coefficient that is averaged together with the fraction 1.0 – OWHTD (0.45) of the present coefficient to define the present time-averaged value.

D.46. HTCREF1.H

COMMON/HTCREF1/ ALPAG2(NXRYT), ALPCF2(NXRYT), ALPRW(NXRYT), ALPSM(NXRYT), ALPTB(NXRYT), FUNH(NXRYT), ZAGS(NXRYT), ZCHFL(NXRYT), ZDFS(NXRYT),

ZRWS(NXRYT), ZSMS(NXRYT), ZTB(NXRTY), QCHF, ZSLAB

COMMON/HTCREF1/ IJ, NNODES, NHSCA(NXRYT)

REAL*8 VARIABLES:

ALPG2	Array of gas volume fractions at the top of the agitated section for a given (r, θ) or (x, y) cell.
ALPCF2	Array of gas volume fractions at the CHF location for a given (r, θ) or (x, y) cell.
ALPRW	Array of gas volume fractions at the top of the rough wavy section for a given (r, θ) or (x, y) cell.
ALPSM	Array of gas volume fractions at the top of the smooth section for a given (r, θ) or (x, y) cell.
ALPTB	Array of gas volume fractions at the transition boiling location for a given (r, θ) or (x, y) cell.
FUNH	Array of the fraction of each heat-structure surface that is unheated.
ZAGS	Array of the elevation where agitated inverted annular flow ends for a given (r, θ) or (x, y) cell.
ZCHFL	Array of the elevation of the CHF point for a given (r, θ) or (x, y) cell.
ZDFS	Array of the elevation where highly dispersed flow begins for a given (r, θ) or (x, y) cell.
ZRWS	Array of the elevation where rough-wavy inverted annular flow ends for a given (r, θ) or (x, y) cell.
ZSMS	Array of the elevation where smooth inverted annular flow ends for a given (r, θ) or (x, y) cell.
ZTB	Array of the elevation of the transition boiling point for a given (r, θ) or (x, y) cell.
QCHF	Critical heat flux (CHF).
ZSLAB	Elevation of the heat-transfer node being considered.

INTEGER VARIABLES:

IJ	(r,θ) or (x, y) horizontal-plane hydraulic-cell number.
NNODES	Number of nodes in a given ROD or SLAB.
NHSCA	Array of HTSTR-component numbers that defines the principal powered RODs or SLABs.

D.47. HTCREF2.H

COMMON/HTCREF2/ TVZ(NZFMX), TWZ(NZFMX), ZNODES(NZFMX)

REAL VARIABLES:

TVZ	Array of gas temperatures for a given ROD or SLAB.
TWZ	Array of wall temperatures for a given ROD or SLAB.
ZNODES	Array of node-center elevations.

D.48. HTCREF3.H

COMMON/HTCREF3/ IFREZ, NREFLD(NXRYT)

INTEGER VARIABLES:

IFREZ	Flag used to turn interfacial vapor heat transfer off; i.e., freeze the drop size.
NREFLD	Flag indicating the reflood model is on (set in subroutine CORE1).

D.49. HTCS.H

COMMON/HTCS/ HVAP, HLIQ, SLIP, QSTEAM, HTCWL, HTCWV, ICONHT, MHTLI, MHTLO, MHTVI, MHTVO

REAL*8 VARIABLES:

HVAP	Enthalpy of the gas.
HLIQ	Enthalpy of the liquid.
SLIP	Slip ratio between phasic velocities.
QSTEAM	Wall-to-wall heat flux.
HTCWL	Constant wall-to-liquid HTC (Namelist variable).
HTCWV	Constant wall-to-gas HTC (Namelist variable).

INTEGER VARIABLES:

ICONHT	Heat-transfer option (Namelist variable). 0 = HTCs evaluated (default); 1 = constant HTCs defined by Namelist variables.
MHTLI	Option for inputting wall-to-liquid heat-transfer multiplier factors for the inner surface of all HTSTRs (Namelist variable).
MHTLO	Option for inputting wall-to-liquid heat-transfer multiplier factors for the outer surface of all HTSTRs (Namelist variable).
MHTVI	Option for inputting wall-to-gas heat-transfer multiplier factors for the inner surface of all HTSTRs (Namelist variable).

MHTVO   Option for inputting wall-to-gas heat-transfer multiplier factors for the outer surface of all HTSTRs (Namelist variable).

D.50. IFCRS.H

COMMON/IFCRS/   ALMAX, ALMIN, ALPBCD, ALPBCH, ALPBCW, ALPDCH, ALPGS, ALPMCT, ALPTS1, ALPTS2, ALPTP, ALPTM, ALPVS, ALVCN, ALVCN1, ALVCN2, ALVEFX, ALVEV, ALVEV1, ALVEV2, ALVFAX, AL01, AUPCT, AUPDRG, CCFLL, CCFVLM, CCFUL, CHTABH, CHTACC, CHTAFX, CHTCN1, CHTCN2, CHTEV1, CHTEV2, CHTFAX, CHTICN, CHTIEV, CNDBS, CNDFL, CNDPL, CNDRO, CNDST, CHTINV, CHTANV, D1X, D2X, DCALW1, DCALW2, DCDGM1, DCDGM2, DTVHT, EPMAX, EPMIN, EVFAX, F2MX, FCSUB, FDIS1, FDIS2, FDISV1, FDISV2, FIFAM, FIFBL, FIFBS, FIFCR, FIFEP, FIFST, FIFWL, FISHI, FLMIN, FLSHF, FLSH1, FLSH2, FRI1, FRI2, FUI1, FUI2

COMMON/IFCRS/   FSB, FFS, FSM, FRW, FMDIS, FCDROP, FDIS, FFD, VOIDS1, VOIDS2, VOIDS3, VOIDD1, VOIDD2, XMDIS, XNB, XHVDIS, HARMX, HAMIN, HCMIN, HCAMIN, HDMAX, HFVL, HFVU, HIMFAC, H0, PC24, PCRIT, REGMN, RDMAX, RDMIN, SLP1, STFRL, STFRU, STSTRT, TLGTS, TVLTL, TVLTS, TWDFAC, TWDFAK, VLACC, VLMAX, VRCMIN, VRFMIN, VR2MIN

COMMON/IRCRS/   IEPRI, IWILS, IHOTP, IPDRGX

REAL*8 VARIABLES:

ALMAX   Maximum gas volume fraction (0.9999) to use in calculation of interfacial drag.

ALMIN   Minimum gas volume fraction (0.00001) to use in calculation of interfacial drag.

ALPBCD   Minimum gas volume fraction (0.00001) to use in calculation of bubbly-interfacial drag.

ALPBCH   Minimum gas volume fraction (0.00001) to use in calculation of bubbly-interfacial heat transfer.

ALPBCW   Minimum gas volume fraction (0.00001) to use in the Wilson model (upper-plenum) calculation of interfacial drag.

ALPDCH	Maximum gas volume fraction (0.9995) for calculation of droplet diameter in the annular-mist regime.
ALPGS	Variable not used.
ALPMCT	Variable not used.
ALPTS1	Variable not used.
ALPTS2	Variable not used.
ALPTP	Minimum gas volume fraction (0.9) in cell above for vertical stratified flow in the VESSEL.
ALPTM	Maximum gas volume fraction (0.1) in cell below for vertical stratified flow in the VESSEL.
ALPVS	Gas volume fraction constant (0.3) in model for bubbly flow below a stratified level.
ALVCN	Time constant in rate model for change in condensing ALVE.
ALVCN1	Constant used in determining the upper bound at the liquid-side HTC for subcooled liquids.
ALVCN2	Constant used in determining the lower bound of the liquid-side HTC for subcooled liquids.
ALVEFX	Constant used in determining the limit of the liquid-side HTC between timesteps.
ALVEV	Constant used in determining the limit of the liquid-side HTC between timesteps.
ALVEV1	Constant used in determining the upper bound of the liquid-side HTC for saturated or superheated liquids.
ALVEV2	Constant used in determining the lower bound of the liquid-side HTC for saturated or superheated liquids.
ALVFAX	Constant used in determining the limit of flashing HTC between timesteps.
AL01	Constant in subcooled boiling model.
AUPCT	Constant in stratified flow model for ACCUMs (variable not used).
AUPDRG	Constant in stratified flow model for ACCUMs (variable not used).
CCFLL	Variable not used.
CCFVLM	Variable not used.
CCFUL	Variable not used.
CHTABH	Gas-side HTC for noncondensable gas in bubbly-slug flow.

CHTACC	Gas-side HTC for ACCUMs (variable not used).
CHTAFX	Constant used in determining the limit of the gas-side HTC for noncondensable gas.
CHTCN1	Constant used in determining the upper bound of the gas-side HTC for subcooled vapor.
CHTCN2	Constant used in determining the lower bound of the gas-side HTC for subcooled vapor.
CHTEV1	Constant used in determining the upper bound of the gas-side HTC for saturated or superheated vapor.
CHTEV2	Constant used in determining the lower bound of the gas-side HTC for saturated or superheated vapor.
CHTFAX	Constant used in determining the limit of the gas-side HTC between timesteps.
CHTICN	Constant used in determining the limits of the gas-side HTC.
CHTIEV	Constant used in determining the limits of the gas-side HTC.
CNDBS	Constant to adjust the interfacial area for condensing bubble.
CNDFL	Constant to adjust the liquid-side HTC in annular-mist flow.
CNDPL	Constant to adjust the liquid-side HTC for condensation in plug flows.
CNDRO	Constant to adjust the liquid-side HTC in annular-mist flows.
CNDST	Multiplier for stratified-flow condensation interfacial heat transfer.
CHTINV	Liquid-side HTC for smooth, rough-wavy, and agitated inverted-annular flows.
CHTANV	Liquid-side HTC of noncondensable gas for smooth, rough-wavy, and agitated inverted-annular flow.
D1X	Constant in EPRI model.
D2X	Constant in EPRI model.
DCALW1	Minimum gas volume fraction (0.0001) for averaging the bubbly-flow-regime axial drag and separated-flow-regime Blausis axial drag in the downcomer and lower plenum.
DCALW2	Maximum gas volume fraction (0.05) for averaging the bubbly-flow-regime axial drag and separated-flow-regime Blausis axial drag in the downcomer and lower plenum.
DCDGM1	Multiplier (0.5) for Blausis axial drag in the downcomer and lower plenum for the bubbly-flow regime.

DCDGM2	Multiplier (0.5) for Blausis axial drag in the downcomer and lower plenum for the separated-flow regime.
DTVHT	Variable not used.
EPMAX	Maximum drag on EPRI model for CORE-component interfacial drag (variable not used).
EPMIN	Minimum drag on EPRI model for CORE-component interfacial drag (variable not used).
EVFAX	Constant in the evaporation model.
F2MX	Factor in the droplet vapor to interface heat-transfer model.
FCSUB	Multiplication constant in the subcooled-boiling condensation model.
FDIS1	Constant in the dispersed-droplet interfacial-drag model.
FDIS2	Constant in the dispersed-droplet interfacial-drag model.
FDISV1	Constant in the dispersed-droplet interfacial-drag model.
FDISV2	Constant in the dispersed-droplet interfacial-drag model.
FIFAM	Factor (1.0) applied to annular-mist interfacial drag.
FIFBL	Multiplier for downcomer interfacial-drag model.
FIFBS	Factor (1.0) applied to bubbly-slug interfacial drag.
FIFCR	Variable not used.
FIFEP	Multiplier (1.0) in EPRI interfacial-drag model (variable not used).
FIFST	Factor (1.0) applied to stratified-flow interfacial drag.
FIFWL	Multiplier (1.0) for Wilson-model interfacial drag in the upper plenum of the VESSEL.
FISHI	Variable not used.
FLMIN	Minimum film thickness for annular flow.
FLSHF	Multiplier on liquid superheat for flashing.
FLSH1	Maximum-flash multiplier.
FLSH2	Minimum-flash multiplier.
FRI1	Time constant for rate of decrease in C_i.
FRI2	Time constant for rate of increase in C_i.
FUI1	Minimum allowed change in C_i.
FUI2	Maximum allowed change in C_i.
FSB	Constant to adjust the interfacial-drag coefficient for the subcooled nucleate-boiling regime.

FFS	Constant to adjust the interfacial-drag coefficient for the free-stream contribution in subcooled nucleate-boiling regime.
FSM	Constant to adjust the interfacial-drag coefficient for the smooth inverted-annular flow.
FRW	Constant to adjust the interfacial-drag coefficient for the rough-wavy inverted-annular flow.
FMDIS	Constant to adjust the interfacial-drag coefficient for the post-agitated inverted-annular flow.
FCDROP	Constant to adjust the droplet interfacial-drag coefficient for lightly dispersed inverted-annular flow.
FDIS	Constant to adjust the interfacial-drag coefficient for highly dispersed inverted-annular flow.
FFD	Constant to adjust the interfacial-drag coefficient for liquid flow in highly dispersed inverted-annular flow.
VOIDS1	Lower limit of the gas volume fraction (0.05) for smooth inverted-annular flow.
VOIDS2	Upper limit on the gas volume fraction for rough-wavy inverted-annular flow.
VOIDS3	Upper limit on the gas volume fraction (0.3) for smooth inverted-annular flow.
VOIDD1	Variable not used.
VOIDD2	Gas volume fraction limit in the rough-wavy reflood flow regime.
XMDIS	Constant to adjust the weighting for the post-agitated inverted-annular-flow interfacial-drag coefficient.
XNB	Constant to adjust the weighting for the bubbly-flow interfacial-drag coefficient in the intermediate gas-volume-fraction region.
XHVDIS	Constant to adjust the weighting for rough-wavy- and smooth-inverted-annular-flow interfacial-drag coefficient in the intermediate gas-volume-fraction region.
HARMX	Variable not used.
HAMIN	Product of the heat-transfer coefficient times the interfacial area.
HCMIN	Variable not used.
HCAMIN	Variable not used.
HDMAX	Hydraulic diameter used in the VESSEL component if user-input hydraulic diameter is $<10^{-5}$.

HFVL	Constant used in dispersed-droplet interfacial-drag model.
HFVU	Constant used in dispersed-droplet interfacial-drag model.
HIMFAC	Multiplication constant used in the calculation of the minimum gas-interface heat-transfer rate.
H0	Constant in subcooled-boiling condensation model.
PC24	Pressure constant (1.95187E+15 Pa^2).
PCRIT	Critical-point pressure (2.209E+07 Pa).
REGMN	Minimum Reynold's number in stratified flow.
RDMAX	Maximum droplet radius in annular-mist flow.
RDMIN	Minimum droplet radius in annular-mist flow.
SLP1	Constant in subcooled-boiling model.
STFRL	Stratified-flow lower-velocity limit multiplier.
STFRU	Stratified-flow upper-velocity limit multiplier.
STSTRT	Multiplier (1.0) on stratified-flow interfacial heat transfer.
TLGTS	Maximum liquid superheat used to calculate limit on interfacial heat transfer.
TVLTL	Maximum gas temperature less than liquid temperature used to calculate limit on interfacial heat transfer.
TVLTS	Maximum vapor subcooling used to calculate limit on interfacial heat transfer.
TWDFAC	Constant in subcooled-boiling model.
TWDFAK	Constant in subcooled-boiling model.
VLACC	Maximum liquid velocity in ACCUM for calculation of interfacial heat transfer (variable not used).
VLMAX	Maximum liquid velocity in annular film for calculation of interfacial heat transfer.
VRCMIN	Minimum relative velocity (0.1) used to calculate a run.
VRFMIN	Minimum relative velocity to be used in the bubbly-slug C_i calculation.
VR2MIN	Variable not used.

INTEGER VARIABLES:

IEPRI	EPRI interfacial-drag-model flag used for rod bundles in the core region when set to 1.
IWILS	Wilson interfacial-drag-model flag for use in the upper plenum when set to 1.
IHOTP	Hot-patch modeling option.

0 = off;

1 = on.

IPDRGX Lower-plenum Blasius interfacial-drag off flag.

0 = apply Blasius interfacial drag in the lower plenum when IBLAUS = 1.

1 = do not apply Blasius interfacial drag in the lower plenum when IBLAUS = 1.

D.51. IFDPTR.H

INTEGER	TMPVL
COMMON/IFDPTR/	TMPVL
COMMON/IFDPTR/	IWRTPT, LASTP1, NTMPV
COMMON/IFDPTR/	LVT1, LVT2, LVT3, LVT4, LVT5, LVT6, LVT7, LVT8, LVT9, LVT10, LVT11, LVT12, LVT13, LVT14, LVT15, LVT16, LVT17, LVT18, LVT19, LVT20, LVT21, LVT22, LVT23, LVT24, LVT25, LVT26, LVT27, LVT28, LVT29, LVT30, LVT31, LVT32, LVT33, LVT34, LVT35, LVT36, LVT37, LVT38, LVT39, LVT40, LVT41, LVT42, LVT43, LVT44, LVT45, LVT46, LVT47, LVT48, LVT49, LVT50, LVT51, LVT52, LVT53, LVT54, LVT55, LVT56, LVT57, LVT58, LVT59, LVT60, LVT61, LVT62, LVT63, LVT64, LVT65, LVT66, LVT67, LVT68, LVT69, LVT70, LVT71, LVT72, LVT73, LVT74, LVT75, LVT76, LVT77, LVT78, LVT79, LVT80, LVT81, LVT82, LVT83, LVT84, LVT85, LVT86, LVT87, LVT88, LVT89, LVT90, LVT91, LVT92, LVT93, LVT94, LVT95, LVT96, LVT97, LVT98, LVT99, LVT100, LVT101, LVT102, LVT103, LVT104, LVT105, LVT106, LVT107, LVT108, LVT109, LVT110, LVT111, LVT112, LVT113, LVT114, LVT115, LVT116, LVT117, LVT118, LVT119, LVT120, LVT121, LVT122, LVT123, LVT124, LVT125, LVT126, LVT127, LVT128, LVT129, LVT130

INTEGER VARIABLES:

TMPVL Number of calculative mesh cells in the 3D VESSEL component.

IWRTPT Flag to set up temporary pointers for subroutines PREFWD and PREIFD.

LASTP1	LAST+1 pointer for the first free location in memory for the temporary storage arrays set up to vectorize the 3D wall-shear and interfacial-drag coefficient evaluations.
NTMPV	Number of temporary storage arrays (130) in subroutines PREFWD and PREIFD set up to vectorize the 3D wall-shear and interfacial-drag coefficient evaluations.
LVT#	Pointer variable for the #th (# = 1 to 130) temporary storage array set up to vectorize the 3D wall-shear and interfacial-drag coefficient evaluations.

D.52. INFOHL.H

COMMON/INFOHL/ DROPD, FHLF, QDEN, QFR, QTOTAL, QWEBB, VR2

REAL*8 VARIABLES:

DROPD	Calculated drop diameter used in the Forsland-Rohsenow correlation.
FHLF	Factor carried along to separate the Denham and Forsland-Rohsenow regions.
QDEN	Heat flux calculated using the Denham correlation.
QFR	Heat flux calculated using the Forsland-Rohsenow correlation.
QTOTAL	Total heat flux calculated including radiation.
QWEBB	Heat flux calculated using the Webb-Chen correlation.
VR2	Local relative velocity minus quench-front relative velocity.

D.53. IOUNITS.H

COMMON/UNITS/	IBFADD, IBFADG, IBFADR, IBFLND, IBFLNG, IBFLNR, IDOUT, IEEEG, IGOUT, IKEYBD, IMOUT, IN, INLAB, INPROC, IOALL, IODONE, IOERR, IOGRF, IOINP, IOLAB, IOOUT, IOSKIP, IOUT, IRSTRT, ITTY, IUNLAB, IUNOUT, LCMCPD, NITTAB, NPWTAB, NRDY
COMMON/CUNITS/	CARD
CHARACTER*100	CARD

INTEGER VARIABLES:

IBFADD	Pointer to the beginning of dump LCM buffer.
IBFADG	Pointer to the beginning of graphics LCM buffer.
IBFADR	Pointer to the beginning of restart LCM buffer.
IBFLND	Length of dump buffer.

IBFLNG	Length of graphics buffer.
IBFLNR	Length of restart buffer.
IDOUT	I/O unit number for dump output file (currently set to unit 12).
IEEEG	Option to specify the format of the TRCGRF file for graphics output (Namelist variable). 0 = unformatted binary file; 1 = IEEE standard-format binary file.
IGOUT	I/O unit number for graphics output file (currently set to unit 11).
IDEYBD	I/O unit number for the terminal keyboard (currently set to unit 59 for –DEF,IBM and unit 5 for DEF,IBM).
IMOUT	I/O unit number for warning messages (currently set to unit 7).
IN	I/O unit number for input to TRAC-P (initially set to unit 5 to point to file TRCINP; if the input does not invoke free format, IN is changed to 1 to point to file TRACIN).
INLAB	I/O unit number for TRAC to generate a labeled input-data file (currently set to unit 3).
INPROC	Flag set during input to indicate whether component data are being processed.
IOALL	IOALL = |IOGRF| + |IOINP| + |IOLAB| + |IOOUT|.
IODONE	Flag that indicates if the current input card has been read.
IOERR	Input error flag.
IOGRF	SI/English-units flag for writing graphics data to file TRCGRF (Namelist variable).
IOINP	SI/English-units flag for reading input data from file TRACIN (Namelist variable).
IOLAB	SI/English-units flag for writing comment-labeled input data to file INLAB (Namelist variable).
IOOUT	SI/English-units flag for echoing input and restart data and writing small and large edits to file TRCO"UT and writing calculative information to file TRCMSG and the terminal (Namelist variable).
IOSKIP	Flag that turns input processing off and on.
IOUT	I/O unit number for the printed-output file (currently set to unit 6).
IRSTRT	I/O unit number for the restart-input file (currently set to unit 13).

ITTY	I/O unit number for terminal output (currently set to unit 59).
IUNLAB	Option for inputting user-defined units-name labels required for defining the units of control block or trip-signal-expression parameters (Namelist variable).
IUNOUT	Option for writing SI/English units to file TRCOUT (Namelist variable).
LCMCPD	Storage for the beginning address for reading from or writing to LCM with calls to subroutines RDLCM and WRLCM.
NITTAB	Flag for printing the timestep data table heading label to the terminal (–1) or message file (–2) because a warning message(s) has been printed since the last table values were printed.
NPWTAB	Flag for printing the power/reactivity feedback table heading label to the message file (–1) because a warning message(s) has been printed since the last table values were printed.
NRDY	Flag for reading the input-data files. 0 = reading the TRCRST restart-data file; 1 = reading the TRACIN user-input-data file.

LOGICAL VARIABLE:

CARD	Variable that contains the current input-card data.

D.54. ITERSTAT.H

COMMON/ISTAT/	VARERM, VERR
COMMON/ISTAT/	IOTT, NSTEP, OITNO
INTEGER	OITNO

REAL*8 VARIABLES:

VARERM	Maximum variable error.
VERR	Velocity error at component junction.

INTEGER VARIABLES:

IOTT	Temporary storage for IITNO.
NSTEP	Number of timesteps evaluated during the TRAC-P calculation.
OITNO	Outer-iteration number.

D.55. JUNCTION.H

COMMON/JUNCT/ JPTR, JMATCH

INTEGER VARIABLES:

JPTR	Number of junction-component pairs.

JMATCH	Number of bad junction numbers detected during the network trace in SRTLP.

D.56. LABELV.H

COMMON/LABELV1/ LABUN(150)
COMMON/LABELV2/ LUNCB(2,150)
COMMON/LABELV3/ LUPCB(2,150)
COMMON/LABELV4/ RUNCB(2,150)
COMMON/LABELV5/ LABSV(2,150)
COMMON/LABELV/ FACTOR(150), OFFSET(150)
COMMON/LABELV/ IH(26), ITLS(777), ITSV(105), ILS, ILU, ILUN, IOLD
COMMON/LABELV/ LABELS(777)
COMMON/LABELV/ ALPBET, LUAR, LUCP, LUD, LUDH, LUE, LUEN, LUH, LUHA, LUHX, LUID, LUIS, LUM, LUMF, LUP, LUPD, LUPH, LUPT, LUPW, LUR, LUS, LUSP, LUSZ, LUT, LUTC, LUTM, LUTP, LUV, LUVF, LUVO, LUZ

NOTE: 777 ---> 806 with pending KAPL updates.

LOGICAL VARIABLES:

LABUN	CHARACTER*8 names of the units-name labels.
LUNCB	CHARACTER*13 left-justified names of the SI-units and English-units symbols of the units-name labels.
LUPCB	CHARACTER*13 names surrounded by parentheses of the SI-units and English-units symbols of the units-name labels.
RUNCB	CHARACTER*12 right-justified names of the SI-units and English-units symbols of the units-name labels.
LABSV	CHARACTER*14 names of the signal-variable parameters.

REAL*8 VARIABLES:

FACTOR	Conversion factor from SI units to English units.
OFFSET	Offset shift value from SI units to English units after the conversion factor is applied to the SI-units value.

INTEGER VARIABLES:

IH	$1 \leq IH(I) \leq 777$ defines the index of the first FORTRAN I/O real-variable name in TRAC-P beginning with the Ith letter of the alphabet.

ITLS	$1 \leq$ ITLS(J) ≤ 150 defines the index of the units-name label defining the units of the Jth FORTRAN I/O real-variable name in TRAC-P.
ITSV	$1 <$ ITSV(K) ≤ 150 defines the index of the units-name label defining the units of the Kth signal-variable parameter.
ILS	Total number of FORTRAN I/O real-variable names in TRAC-P.
ILU	Index of the last units-name label defined internally in TRAC-P.
ILUN	Index of the last units-name label in TRAC-P after user-defined units-name labels are input.
IOLD	Index of the units-name label for the last FORTRAN I/O real-variable name processed for possible units conversion by subroutine UNCNVT.

LOGICAL VARIABLES:

LABELS	CHARACTER*8 names of the FORTRAN I/O real variables in TRAC-P.
ALPBET	CHARACTER*26 string of the 26 letters of the alphabet.
LUAR	Commonly used CHARACTER*4 units symbol ' m2 ' or ' ft2' for area.
LUCP	Commonly used CHARACTER*10 units symbol ' w*s/kg/k ' or ' btu/lbm/f' for specific heat.
LUD	Commonly used CHARACTER*2 units symbol ' –' for a dash indicating no units.
LUDH	Commonly used CHARACTER*19 units symbol '——— w/m2/k ———' or '— btu/ft2/f/hr —' with dashes on each side for a heat-transfer coefficient.
LUE	Commonly used CHARACTER*4 units symbol ' w*s' or ' btu' for energy.
LUEN	Commonly used CHARACTER*7 string ' end ' or 'end (s)'.
LUH	Commonly used CHARACTER*13 units symbol ' w/m2/k ' or ' btu/ft2/f/hr' for a heat-transfer coefficient.
LUHA	Commonly used CHARACTER*9 units symbol ' w/k ' or ' btu/f/hr' for interfacial heat transfer total flux.
LUHX	Commonly used CHARACTER*12 units symbol ' w/m2 ' or ' btu/ft2/hr ' for heat flux.
LUID	Commonly used CHARACTER*8 units symbol ' kg/m4 ' or ' lbm/ft4' for interfacial drag.

LUIS	Commonly used CHARACTER*6 units symbol ' rad/s' or ' rpm ' for the pump-impeller rotational speed.
LUM	Commonly used CHARACTER*4 units symbol ' kg ' or ' lbm' for mass.
LUMF	Commonly used CHARACTER*7 units symbol ' kg/s ' or ' lbm/hr' for mass flow.
LUP	Commonly used CHARACTER*5 units symbol ' pa ' or ' psia' for absolute pressure.
LUPD	Commonly used CHARACTER*5 units symbol ' pa ' or ' psid' for a pressure difference.
LUPH	Commonly used CHARACTER*11 units symbol ' m2/s2 ' or ' lbf*ft/lbm' for pump head.
LUPT	Commonly used CHARACTER*7 units symbol ' pa*m3 ' or ' lbf*ft' for torque.
LUPW	Commonly used CHARACTER*7 units symbol ' w ' or ' btu/hr' for power.
LUR	Commonly used CHARACTER*8 units symbol ' kg/m3 ' or ' lbm/ft3' for density.
LUS	Commonly used CHARACTER*2 units symbol ' s' for time.
LUSP	Commonly used CHARACTER*8 string ' step ' or 'step (s)'.
LUSZ	Commonly used CHARACTER*8 string ' size ' or 'size (s)'
LUT	Commonly used CHARACTER*3 string ' **'
LUTC	Commonly used CHARACTER*12 units symbol ' w/m/k ' or ' btu/ft/f/hr' for thermal conductivity.
LUTM	Commonly used CHARACTER*8 string ' time ' or 'time (s)'.
LUTP	Commonly used CHARACTER*2 units symbol ' k' or ' f' for temperature.
LUV	Commonly used CHARACTER*5 units symbol ' m/s ' or ' ft/s' for velocity.
LUVF	Commonly used CHARACTER*5 units symbol ' m3/s' or ' gpm ' for volumetric flow.
LUVO	Commonly used CHARACTER*4 units symbol ' m3 ' or ' ft3' for volume.
LUZ	Commonly used CHARACTER*3 units symbol ' m ' or ' ft' for length.

D.57. MASSCK.H

COMMON/MASSCK/ NSTABO

INTEGER VARIABLE:

NSTABO	Old value of NSTAB (flag for evaluating the SETS3D equations) from the previous timestep.

D.58. MELFLG.H

COMMON/MELFLG/ MELTRC

INTEGER VARIABLE:

MELTRC	Flag to indicate whether subroutine THERMO is called from TRAC-P components or MELVSL (necessary due to differing convention on mixture properties). 0 = call is from MELVSL; 1 = call is from TRAC-P.

D.59. MEMORY.H

COMMON/TIMER/ ADATE, ATIME, CPUT, TIMCPU, TIMEI, TIMIOM, TIMSYS, TIMTOT

COMMON/TIMER NSTEPT

REAL*8 VARIABLES:

ADATE	Date obtained from calling system routine DATE.
ATIME	Time obtained from calling system routine DATE.
CPUT	Cumulative CPU time from previous jobs in a restarted series of calculations; CPUT is set to 0.0 s at time 0.0 s.
TIMCPU	CPU time obtained from calling system routine TIMING.
TIMEI	Time limit of the current job obtained from calling system routine GETJTL.
TIMIOM	I/O time obtained from calling system routine TIMING.
TIMSYS	System time obtained from calling system routine TIMING.
TIMTOT	Total of CPU, I/O, and system times obtained from calling system routine TIMING.

INTEGER VARIABLE:

NSTEPT	Cumulative number of timesteps from previous jobs in a restarted series of calculations; NSTEPT is set to 0 at time 0.0 s.

D.60. NAVGN.H

COMMON/NAVGN/ NAVG1

INTEGER VARIABLE:

NAVG1	Value defined to IDALPI in subroutine TF1DS when the interface is a junction connected to a BREAK component with flow into the BREAK.

D.61. NMFAIL.H

COMMON/NMFAIL IFTP, ITFL1, NFL1, NFL3

INTEGER VARIABLES:

IFTP	Flag that prevents thermal failure messages if a message has come from TF1SD3 or FF3D.
ITFL1	Iteration number of the last TF1DS3 failure.
NFL1	Total number of TF1DS3 failures in the current timestep.
NFL3	Total number of FF3D failures in the current timestep.

D.62. NRCMP.H

COMMON/NRCMP/ NCMPMX, NHTSMN, NRCOMP

INTEGER VARIABLES:

NCMPMX	Maximum hydraulic-component number.
NHTSMN	Minimum heat-structure component number.
NRCOMP	Number of components defined from the TRCRST restart-data file.

D.63. OVLI.H

COMMON/OVLI/ JFLAG, ISTORE

INTEGER VARIABLES:

JFLAG	Flag that is set to 1 when an input-data error is encountered and TRAC-P is to abort the calculation after all input data have been processed.
ISTORE	Pointer variable for the A array where unused computer-memory space starts.

D.64. PMPSTB.H

COMMON/PMPSTB/ FWPA

COMMON/PMPSTB/ IPMPCN

REAL*8 VARIABLE:

FWPA — Fraction 0.1 of the present donor-celled gas volume fraction across the pump-impeller interface that is averaged with the fraction (1.0 – FWPA = 0.9) of its previous gas volume fraction average to define the gas volume fraction for evaluating the PUMP-curve HDM table.

INTEGER VARIABLE:

IPMPCN — Flag for not defining the donor-celled mixture density and gas volume fraction across the pump-impeller interface.

D.65. POINTERS.H

INTEGER PTRSSIZE

PARAMETER (PTRSSIZE=75) —Size of Common Block PTRS—

—General Pointers—

COMMON/PTRS/ LBD, LCNTL, LCOMPT, LCONTP, LCONTR, LDRA, LDRC, LICVS, LDPMAX, LIJVS, LILCMP, LIOU, LISVF, LIVCON, LIVLJN, LJOUT, LJSEQ, LJUN, LLCMHS, LLCON, LLOOPN, LMATB, LMCMSH, LMSCT, LNBR, LNJN, LNSIG, LNSIGP, LNVCNL, LORDER, LPRPTB, LPTBLN, LTITLE, LVSI, LWP

—Network Solution Pointers—

COMMON/PTRS/ LAOL, LAOU, LAOV, LDPVC, LDPVCV, LDREL, LDREV, LDRL, LDRV, LDVB, LIDPCV, LOD, LVRH, JAOL, JAOV, JDRV, JDRL, JOD, JDREL, JDREV, JDRA, JDRC, JNJUN, LILPRB, LIVLFC, LIVVTO, LIVLTO

—Combination of Unshifted Pointers and Array Lengths—

COMMON/PTRS/ LBVEC, LBW, LDMAT, LEMAT, LENFXD, LFXD, LRMAT, LVMAT, LVSSC, LVSSIP, NCLEAR, NMAT, NVCELL

GENERAL POINTERS:

LBD — Boundary-data array pointer.

LCNTL — A-array pointer for the signal-variable, control-block, and trip control-parameter data.

LCOMPT — Component LCM pointers stored in the order used for iteration.

LCONTP — Number of constrained steady-state controllers that adjust a VALVE to achieve a desired upstream pressure.

LCONTR	Pointer to the location where the first parameter of constrained steady-state parameter data is stored in the A array.
LDRA	Storage for right-hand side of the noncondensable stabilizer mass equation.
LDRC	Pointers for network variables for the solute-tracking option.
LICVS	Pointer for a temporary array that contains a list of all VESSEL composite-cell numbers that have a source connection to one of their cell faces.
LDPMAX	Pointer for an array saving the maximum pressure error for each hydraulic component during the last outer iteration.
LIJVS	Pointer for a temporary array that contains a list of all junction numbers that link to a VESSEL.
LILCMP	Component LCM pointers stored in the order in which the components were read.
LIOU	Network junction numbers for the junctions of all components excluding BREAKs and FILLs.
LISVF	Pointer to an array of flags indicating whether or not a particular component is used to evaluate one or more signal variables (–1, no signal variable; +1, signal variable); this array uses the same order in which the component data are processed.
LIVCON	Pointer to network junction numbers that connect to a VESSEL.
LIVLJN	IVLJN(I) is the VESSEL junction number that corresponds to the network junction number given by IVCON(I).
LJOUT	Storage area for pointers that locate the beginning of each system loop within data for IOU.
LJSEQ	Junction numbers in the order in which junctions occur in the junction-component array.
LJUN	Junction-component pair array pointer.
LLCMHS	Pointer to define the starting address for the fixed-length table of each heat-structure component.
LLCON	Number of times each component was the last to converge since the last edit.
LLOOPN	IA(LLOOPN+IL–1) gives the element of the IORDER array that begins the ILth loop pass.
LMATB	Pointer for additional material-property ID numbers.

LMCMSH	Storage for number of coarse-mesh VESSEL source cells or absolute cell index if direct VESSEL solution is used.
LMSCT	Temporary storage for VESSEL pressure changes adjacent to sources.
LNBR	Component numbers stored in the order in which components were read.
LNJN	NJN(IL) is the number of network junctions in loop IL.
LNSIG	NSIG(IL) is the total number of components excluding BREAKs and FILLs in a loop.
LNSIGP	NSIGP(IL) is NSIG(IL).
LNVCNL	IA(LNVCNL+IL–1) points to the elements of IVCON and IVLJN that begin the IL^{th} loop.
LORDER	Component numbers stored in the order used for iteration.
LPRPTB	Pointer to user-defined material-property tables.
LPTBLN	Pointer for the number of entry groups in the user-defined material-property table.
LTITLE	Problem title and version information (stored using only the first four bytes of each word).
LVSI	Junction flow reversal indicators in the order in which junctions occur in the junction-component array.
LWP	Pointer for the composite location numbers of hydraulic cells coupled to a heat-structure component surface.

NETWORK SOLUTION POINTERS:

LAOL	Variable to rework solution of ARL, AREL, and VLT (contains rework matrix).
LAOU	Network junction coefficient matrix.
LAOV	Variable to rework solution of ARV, AREV, and VVT (contains rework matrix).
LDPVC	Locator that shows the beginning of coefficients to evaluate the derivatives of junction velocities with respect to VESSEL pressures.
LDPVCV	Pointer for reordered coupling coefficients between the VESSEL and the 1D network solution.
LDREL	Storage for right-hand side of the liquid stabilizer equation.
LDREV	Storage for right-hand side of the vapor stabilizer equation.

LDRL — Variable to rework solution of ARL and VLT (contains right-hand side of linear equations).

LDRV — Variable to rework solution of ARV and VVT (contains right-hand side of linear equations).

LDVB — Storage for the right-hand side of the network junction equations or the changes in junction velocities.

LIDPCV — Pointers to coefficients stored in DPCV.

LOD — Temporary storage for intercomponent coupling information.

LVRH — Storage for explicit information to evaluate equations of motion at network junctions.

JAOL — Variable not used.

JAOV — Variable not used.

JDRV — Variable not used.

JDRL — Variable not used.

JOD — Variable not used.

JDREL — Variable not used.

JDREV — Variable not used.

JDRA — Variable not used.

JDRC — Variable not used.

JNJUN — Temporary storage location used to define the number of junctions in the current network solution procedure.

LILPRB — Pointer for the A array which defines if each hydraulic loop has VESSEL predictor velocities coupled in different directions.

LIVLFC — Pointer for the A array which defines the face-connect number for all junction connections to VESSELs for a given hydraulic loop.

LIVVTO — Pointer for the A array which defines the gas tilde velocity at a source-connection junction to a VESSEL for a hydraulic loop.

LIVLTO — Pointer for the A array which defines the liquid tilde velocity at a source-connection junction to a VESSEL for a hydraulic loop.

COMBINATION OF UNSHIFTED POINTERS AND ARRAY LENGTHS:

LBVEC — Pointer for storing in the A array the capacitance-matrix equation right-hand-side vector.

LBW — Number of element rows in the array that stores the VESSEL banded coefficient matrix.

LDMAT	Pointer for storing in the A array the capacitance coefficient matrix.
LEMAT	Pointer for storing in the A array the E matrix of the capacitance-matrix method.
LENFXD	Length of data that always remains in the SCM array A.
LFXD	First word address in the A array of the data defined by LENFXD.
LRMAT	Pointer for storing in the A array the R matrix of the capacitance-matrix method.
LVMAT	VESSEL matrix storage for coarse-mesh rebalance or direct inversion.
LVSSC	Right-hand side of equation associated with LVMAT.
LVSSIP	Pivoting information for LVMAT.
NCLEAR	Number of values in the A (LVMAT) array storing the VESSEL banded coefficient matrix.
NMAT	Number of additional material-property tables provided by the user through input.
NVCELL	Total number of cells in all VESSELs.

D.66. PSE.H

COMMON/PSE/ NPICMP, NPSE, NPSE1, NPSE3, NPSIZ, NPSJ, NPSK, NPSV1, NPSHTI

INTEGER VARIABLES:

NPICMP	Component number in TF1DS if NSTEP = NPSE1 and in HTIF if NSTEP = NPSHTI that causes a pause.
NPSE	Pause in TRANS if NSTEP = NPSE.
NPSE1	Pause in TF1DS if NSTEP = NPSE1; the cell number is NPSJ, and the component number is NPICMP.
NPSE3	Pause in TF3DS if NSTEP = NPSE3; the cell index K is NPSK, and the second level is NPSIZ.
NPSIZ	Pause in TF3DS for each level if NSTEP = NPSE3.
NPSJ	Pause in TF1DS for each cell if NSTEP = NPSE1.
NPSK	Pause in TF3DS for each cell if NSTEP = NPSE3.
NPSV1	Pause in TF1DS1 if NSTEP = NPSE1; the cell number is NPSJ and the component number is NPICMP.
NPSHTI	Pause in HTIF if NSTEP = NPSHTI.

D.67. RADATA.H

COMMON/RADATA/ ALPR1, ALPR2, CRAD1, DDRMIN, DDRMAX, RADC1, RADC2, RADGC1, RADGC2, RADGC3, RADGC4, RADGC5, RADGC6, RADGC7, RADGC8, RADGC9, RADG10

REAL*8 VARIABLES:

ALPR1	Gas volume fraction below which the liquid absorbs all the radiant energy.
ALPR2	Gas volume fraction above which the liquid and gas absorb all the radiant energy.
CRAD1	Exponent-power constant used in subroutine RADFP for liquid property.
DDRMIN	Minimum drop size in the radiation model.
DDRMAX	Maximum drop size in the radiation model.
RADC1	Constant used in subroutine RADFP.
RADC2	Constant used in subroutine RADFP.
RADGC1	Constant used in subroutine RADFP for gas property.
RADGC2	Constant used in subroutine RADFP for gas property.
RADGC3	Constant used in subroutine RADFP for gas property.
RADGC4	Constant used in subroutine RADFP for gas property.
RADGC5	Constant used in subroutine RADFP for gas property.
RADGC6	Constant used in subroutine RADFP for gas property.
RADGC7	Constant used in subroutine RADFP for gas property.
RADGC8	Constant used in subroutine RADFP for gas property.
RADGC9	Constant used in subroutine RADFP for gas property.
RADG10	Constant used in subroutine RADFP for gas property.

D.68. RADNEL.H

COMMON/RADMEM/ NFIX1, NFIX2, NFIX3, NFQUAN, MTNFCE, MTNHYD, MNTFCE, MXTFCE, NUTFCE, MAXHZS, MFIXLD, MVARLV, MVARLD

COMMON/RADPTB/ LENCLI, LENCLO, LTOTHL, LTOTZS, LTOTRF, LNPMF, LUTOTF, LTFOS, LMTFOS, LTMPE1

COMMON/RADPTR/ LZFACP, LPMPTR, LHTSCM, LHTSND, LINOUT, LHTSCB, LRODNM, LHSARA, LEMCO1, LEMCO2,

	LEMCO3, LEMISN, LRGMRD, LRODTP, LQRADG, LQRADL, LQRAD, LQRADP
COMMON/RADPTR/	LGVF, LPATHYL, LEMITG, LASBG, LPROBG, LTAUG, LEMITL, LASBL, LPROBL, LTAUL, LTAU
COMMON/RADPTR/	LENCO, LHYDEO, LFACEM, LHYDMU, LHYDM, LVLOS
COMMON/RADPTH/	LHYDOS, LHYDCM, LHYDCL, LRODID, LHYALP, LHYP, LHYPA, LHYROL, LHYROV, LHYMUL, LHYMUV, LHYSIG, LHYTL, LHYTV, LHYQRV, LHYQRL, LHYHD, LHYVL, LHYVV, LHYPMD, LHYRGM, LHYQDG, LHYQDL, LENDRD

INTEGER VARIABLES:

INFORMATION NEEDED TO SET UP RADIATION DATA STORAGE

NFIX1	Number of pieces of information related to each enclosure needed to initially determine computer-memory requirements.
NFIX2	Number of single information entries for a single face of any enclosure.
NFIX3	Number of hydraulic-level information entries for a single hydraulic level.
NFQUAN	Number of arrays stored for each face which are 'the total number of faces' long.
MTNFCE	Total number of radiation faces for all levels of all enclosures, which is equal to the sum of the number of faces times 2 times the number of hydraulic levels.
MTNHYD	Total number of radiation hydraulic levels for all enclosures.
MNTFCE	Minimum number of faces for any radiation enclosure.
MXTFCE	Maximum number of faces for any radiation enclosure.
NUTFCE	Number of unique total number of faces for all enclosures.
MAXHZS	Maximum number of hydraulic levels for any enclosure.
MFIXLD	Total computer memory required for single-length variables.
MVARLV	Length of the vector required to store a single set of information, which is dependent on the total number of faces over all radiation levels.
MVARLD	Total computer memory required for information associated with the total number of faces over all radiation levels.

BASIC INFORMATION FOR EACH RADIATION ENCLOSURE

LENCLI — A-array pointer where the basic enclosure information with respect to the enclosure number is stored (these are TRAC's reordered numbers).

LENCLO — A-array pointer where the basic enclosure information with respect to the enclosure number is stored (these are the user's original enclosure numbers).

LTOTHL — A-array pointer where the basic enclosure information with respect to the total number of enclosure faces is stored.

LTOTZS — A-array pointer where the basic enclosure information with respect to the total number of hydraulic levels is stored.

LTOTRF — A-array pointer where the basic enclosure information with respect to the total number of enclosure levels is stored.

LNPMF — A-array pointer where the basic enclosure information with respect to whether the radiation level has a participating medium or not is stored.

NUTOTF — A-array pointer where the basic enclosure information with the unique total number of faces in increasing order.

LTFOS — A-array pointer where the basic enclosure information with the off set in variable length data corresponding to where the total number of faces changes.

LMTFOS — A-array pointer where the basic enclosure information with the m-number corresponding to a change in the total number of faces takes place.

LTMPE1 — A-array pointer where the scratch vector in basic enclosure information is located.

INFORMATION NEEDED TO SET UP AND LOCATE THE MAJOR PORTION OF THE RADIATION MODEL'S DATA

LZFACP — A-array pointer where a vector 'MTNFCE' long exists that contains the face number of each radiation level for all radiation levels is stored.

LPMPTR — Pointer where flag for whether the face as a participating media is stored within the face-related information in the A array.

LHTSCM — Pointer where the heat-structure component number for a given face is stored within the face-related information in the A array.

LHTSND — Pointer where the heat-structure node/elevation number for a given face is stored within the face-related information in the A array.

LINOUT — Pointer where information as to whether the radiation node is on the inner or outer surface of the heat structure is stored in the A array.

LHTSCB — Pointer where information as to whether this m-number face has its wall heat flux combined with another for the conduction solution is stored in the A array.

LRODNM — Pointer where information as to the heat-structure rod number for a given radiation face is stored in the A array.

LHSARA — Pointer where information as to the heat-structure surface area for a given radiation face is stored in the A array.

LEMCO1 — A-array pointer where the first coefficient for a quadratic fit to emissivity vs temperature for the radiation model begins.

LEMCO2 — A-array pointer where the second coefficient for a quadratic fit to emissivity vs temperature for the radiation model begins.

LEMCO3 — A-array pointer where the third coefficient for a quadratic fit to emissivity vs temperature for the radiation model begins.

LEMISN — A-array pointer where the radiation-level surface emissivity is stored.

LRGMRD — A-array pointer where the radiation flow regime for each radiation face is stored.

LRODTP — A-array pointer where addresses of the wall temperatures for each radiation face are stored.

LQRADG — A-array pointer where phasic radiation heat flux for gas for each radiation face is stored.

LQRADL — A-array pointer where phasic radiation heat flux for liquid for each radiation face is stored.

LQRAD — A-array pointer where the radiation heat flux for each radiation face is stored.

LQRADP — A-array pointer where addresses of the radiation wall heat fluxes for the proper heat-structure node for each radiation face are stored.

INFORMATION NEEDED TO SET UP AND LOCATE STORAGE NEEDED FOR THE RADIATION MODEL

LGVF	A-array pointer where geometric view-factor storage for the radiation model begins.
LPATHL	A-array pointer where the path-length storage for the radiation model begins.
LEMITG	A-array pointer where the gas-emittance storage for the radiation model begins.
LASBG	A-array pointer where the gas-absorbence storage for the radiation model begins.
LPROBG	A-array pointer where the gas-probability storage for the radiation model begins.
LTAUG	A-array pointer where the gas-transmittance storage for the radiation model begins.
LEMITL	A-array pointer where the liquid-emittance storage for the radiation model begins.
LASBL	A-array pointer where the liquid-absorptance storage for the radiation model begins.
LPROBL	A-array pointer where the liquid-probability storage for the radiation model begins.
LTAUL	A-array pointer where the liquid-transmittance storage for the radiation model begins.
LTAU	A-array pointer where the total-transmittance storage for the radiation model begins.

INFORMATION ASSOCIATED WITH THE RADIATION MODEL'S DATA

LENCO	A-array pointer where the integer vector of the basic enclosure information that is NENCL long is stored.
LHYDEO	A-array pointer where the integer vector of the basic enclosure information that is NENCL long is stored.
LFACEM	A-array pointer where a vector MTNFCE long is stored that contains m-numbers ordered in terms of a 4D array where the m-numbers are a function of the face number, radiation-level number, hydraulic-cell number, and enclosure number.
LHYDMU	A-array pointer where a vector MTNFCE long is stored that contains a hydraulic-level number for each m-number, which is

an assigned number that always increases independent of what enclosure is involved.

LHYDM A-array pointer where a vector MTNFCE long is stored that contains a hydraulic-level number for each m-number, which is an assigned number that is relative to the enclosure being considered.

LVLOS A-array pointer where an integer vector MTNFCE long is stored that contains offset points for variable length data associated with each enclosure's m numbers.

INFORMATION, ASSOCIATED WITH EACH RADIATION HYDRAULIC LEVEL, THAT SETS UP AND LOCATES RADIATION-MODEL STORAGE

LHYDOS A-array pointer where off set data associated with radiation hydraulic-level information begins.

LHYDCM A-array pointer where the hydraulic-component number (with which a given face might communicate) is stored within the hydraulic-level related information.

LHYDCL A-array pointer where the hydraulic-component cell number (with which a given face might communicate) is stored within the hydraulic-level related information.

LRODID A-array pointer where information as to the rod ID number or the hydraulic r-theta (with which a given face might communicate) is stored within the hydraulic-level related information.

LHYALP A-array pointer where indices for the hydraulic-cell gas volume fraction for a given hydraulic-level is stored within the hydraulic-level related information.

LHYP A-array pointer where indices for the hydraulic-cell pressure for a given hydraulic-level is stored within the hydraulic-level related information.

LHYPA A-array pointer where indices for the hydraulic-cell non-condensable-gas pressure for a given hydraulic-level is stored within the hydraulic-level related information.

LHYROL A-array pointer where indices for the hydraulic-cell liquid density for a given hydraulic-level is stored within the hydraulic-level related information.

LHYROV A-array pointer where indices for the hydraulic-cell gas density for a given hydraulic-level is stored within the hydraulic-level related information.

LHYMUL A-array pointer where indices for the hydraulic-cell liquid viscosity for a given hydraulic-level is stored within the hydraulic-level related information.

LHYMUV A-array pointer where indices for the hydraulic-cell gas viscosity for a given hydraulic-level is stored within the hydraulic-level related information.

LHYSIG A-array pointer where indices for the hydraulic-cell surface tension for a given hydraulic-level is stored within the hydraulic-level related information.

LHYTL A-array pointer where indices for the hydraulic-cell liquid temperature for a given hydraulic-level is stored within the hydraulic-level related information.

LHYTV A-array pointer where indices for the hydraulic-cell gas temperature for a given hydraulic-level is stored within the hydraulic-level related information.

LHYQRV A-array pointer where indices for the hydraulic-cell gas radiation heat flux for a given hydraulic-level is stored within the hydraulic-level related information.

LHYQRL A-array pointer where indices for the hydraulic-cell liquid radiation heat flux for a given hydraulic-level is stored within the hydraulic-level related information.

LHYHD A-array pointer where indices for the hydraulic-cell hydraulic diameter for a given hydraulic-level is stored within the hydraulic-level related information.

LHYVL A-array pointer where indices for the hydraulic-cell liquid velocity for a given hydraulic-level is stored within the hydraulic-level related information.

LHYVV A-array pointer where indices for the hydraulic-cell gas velocity for a given hydraulic-level is stored within the hydraulic-level related information.

LHYPMD A-array pointer where the flag for whether the hydraulic cell has a participating media is stored within the hydraulic-related information.

LHYRGM	A-array pointer where the radiation flow-regime indicator is stored within the hydraulic-related information.
LHYQDG	A-array pointer where the phasic radiation heat flux for the gas is stored within the hydraulic-related information.
LHYQDL	A-array pointer where the phasic radiation heat flux for the liquid is stored within the hydraulic-related information.
LENDRD	A-array pointer where radiation data storage ends.

D.69. RADTMP.H

COMMON/RADTMP/ LTMPI1, LTMPI2, LTMPR1, LTMPR2, LTMPR3, LTMPR4, LTMPL1, LTMPL2

INTEGER VARIABLES:

LTMPI1	A-array pointer to a vector for temporary storage of integers.
LTMPI2	A-array pointer to a vector for temporary storage of integers.
LTMPR1	A-array pointer to a vector for temporary storage of reals.
LTMPR2	A-array pointer to a vector for temporary storage of reals.
LTMPR3	A-array pointer to a vector for temporary storage of reals.
LTMPR4	A-array pointer to a vector for temporary storage of reals.
LTMPL1	A-array pointer to a vector for temporary storage of long vectors.
LTMPL2	A-array pointer to a vector for temporary storage of long vectors.

D.70. REFHTI.H

COMMON/REFHTI/ AGALP, AGSZ, CHFALP, CHFHV, CHFZ, DFALP, DFSZ, RWALP, RWSZ, SMALP, SMSZ, TBALP, TBZ, UNHF, CAFJ, VLAG, VVAG

REAL*8 VARIABLES:

AGALP	Gas volume fraction at the agitated section of inverted-annular flow.
AGSZ	Elevation of the agitated section of inverted-annular flow.
CHFALP	Gas volume fraction at the CHF point.
CHFHV	Vapor heat transfer at CHF.
CHFZ	Elevation of CHF.
DFALP	Gas volume fraction at the highly dispersed section elevation.
DFSZ	Elevation of highly dispersed section of inverted-annular flow.
RWALP	Gas volume fraction of rough-wavy section elevation.
RWSZ	Elevation of rough-wavy section of inverted-annular flow.

SMALP	Gas volume fraction at the smooth section elevation.
SMSZ	Elevation of smooth section of inverted-annular flow.
TBALP	Gas volume fraction at transition-boiling point.
TBZ	Elevation of transition boiling.
UNHF	Fraction of heated surface that is unheated.
CAFJ	Capillary number.
VLAG	Liquid velocity at the agitated level.
VVAG	Gas velocity at the agitated level.

D.71. REFHTI2.H

COMMON/REFHTI2/ ALPTSL, ALPTRL, ALPTAL, ALPTSU, ALPTRU, ALPTAU

REAL*8 VARIABLES:

ALPTSL	Minimum gas volume fraction allowed for the end of the smooth-inverted flow regime.
ALPTRL	Minimum gas volume fraction allowed for the end of the rough-wavy-inverted flow regime.
ALPTAL	Minimum gas volume fraction allowed for the end of the agitated-inverted flow regime.
ALPTSU	Maximum gas volume fraction allowed for the end of the smooth-inverted flow regime.
ALPTRU	Maximum gas volume fraction allowed for the end of the rough-wavy-inverted flow regime.
ALPTAU	Maximum gas volume fraction allowed for the end of the agitated-inverted flow regime.

D.72. RESTART.H

INTEGER DLNFLT, DNCOMP, ICTRLR(8)

COMMON/RSTART/ DDATE, DDTIME

COMMON/RSTART/ DLNFLT, DNCOMP, ICTRLR

REAL*8 VARIABLES:

DDATE	Date the restart file was created.
DDTIME	Time the restart file was created.

INTEGER VARIABLES:

DLNFLT	Length of the fixed-length tables read from the restart file.

DNCOMP	Number of components in the restart file.
ICTRLR	Array that contains buffering information about the restart file.

D.73. ROWS.H

COMMON/ROWS/ ISCL

INTEGER VARIABLE:

ISCL	Flag that has TRAC-P divide by the largest matrix element in each matrix row for all 4 or 5 matrix elements and 3 right-hand-side elements in each row of the 4 x 4 or 5 x 5 outer-iteration mesh-cell matrix equation. 0 = no; 1 = yes.

D.74. RSPARM.H

COMMON/RSPARM/ DTSTRT

COMMON/RSPARM/ ICDELT

REAL*8 VARIABLE:

DTSTRT	Timestep that can be set as the initial timestep size for a restart calculation (Namelist variable; –1.0 default value).

INTEGER VARIABLE:

ICDELT	Option that overrides the evaluation of DELT at the beginning of an initial calculation. 0 = DELT is set to DTMIN; 1 = DELT is evaluated.

D.75. SEPCB.H

COMMON/SEPCB/ ALPSPC, ALPDRC, DPSEPC

COMMON/SEPCB/ ISEPCB, IDRYCB, NCSEPC, NDRYRC, NSEPSC, ISTAGC

REAL*8 VARIABLES:

ALPSPC	Separator gas volume fraction.
ALPDRC	Gas volume fraction to be convected from the dryer.
DPSEPC	Separator pressure drop.

INTEGER VARIABLES:

ISEPCB	Separator flag.
IDRYCB	Dryer flag.
NCSEPC	Cell number for separator.

NDRYRC	Cell number for dryer.
NSEPSC	Number of separators modeled.
ISTAGC	Separator-option type.

D.76. SIGNAL.H

DIMENSION CPV(42), DSV(2)
COMMON/SIGNAL/ CPV, DSV

REAL*8 VARIABLES:

CPV	Control-panel vector for storing the values of signal-variable parameter numbers 1 through 6 for the global parameters and 7 through 15 for up to four coolant loops (variable not used).
DSV	Dummy signal-variable vector for storing the values of signal-variable parameter numbers 16 and 17 (variable not used).

D.77. SOLCON.H

COMMON/SOLCON/ CNT, CNC, CNTLMN, CNMIN, CNTLMX, CNMAX

REAL*8 VARIABLES:

CNT	Coefficient of liquid temperature (kg solute/kg liquid K, lb_m solute/lb_m liquid F) in linear fit to solubility.
CNC	Constant term (kg solute/kg liquid, lb_m solute/lb_m liquid) in linear fit to solubility.
CNTLMN	Minimum liquid temperature (K, F) of linear fit.
CNMIN	Solubility (kg solute/kg liquid, lb_m solute/lb_m liquid) when the liquid temperature is at or below CNTLMN.
CNTLMX	Maximum liquid temperature (K, F) of linear fit.
CNMAX	Solubility (kg solute/kg liquid, lb_m solute/lb_m liquid) when the liquid temperature is at or above CNTLMX.

D.78. STDYERR.H

DIMENSION	FMAX(7), LOK(7,2)
REAL*8	MAXFLN
COMMON/SSCON/	CF, EPS, EPSPOW, FFLW, FMAX, MAXFLN, RPCF, RTWFP, STIME, TPOWR
COMMON/SSCON/	IPOVEL, IPOWR, ISSCVT, LOK, NCORES, NEF, NET, NOPOW

REAL*8 VARIABLES:

CF	Fluid mass flow through the reactor-core region.
EPS	Tolerance on calculation time for editing and terminating the problem.
EPSPOW	Convergence criterion on the fractional change in liquid velocity per second for setting on the steady-state power when all reactor-core inlet interfaces satisfy this criterion.
FFLW	Fraction of the steady-state power level that the coolant mass flow through the reactor core times RPCF defines.
FMAX	Array of maximum normalized errors.
MAXFLN	Maximum 1D mass flow at this steady-state convergence test.
RPCF	Ratio of reactor-core power to coolant mass flow based on the difference in internal energies from the core inlet and outlet temperatures that are input.
RTWFP	Ratio of heat-transfer to fluid-dynamics timestep sizes.
STIME	Steady-state calculation time.
TPOWR	Steady-state calculation time when the reactor-core power is set on.

INTEGER VARIABLES:

IPOVEL	Number of reactor-core inlet interfaces that satisfy the EPSPOW criterion based on the date-of-change of the liquid velocity.
IPOWR	Flag that turns on the steady-state power.
ISSCVT	Option for evaluating the EPSS steady-state convergence test during a TRANSI = 1 transient calculation (Namelist variable).
LOK	Array of locations of maximum normalized errors.
NCORES	Total number of reactor-core region inlet interfaces.
NEF	Number of timesteps (100) between steady-state convergence check printouts to the terminal and message files.
NET	Number of timesteps (5) between steady-state convergence checks.
NOPOW	Steady-state power flag. 0 = on; 1 = off.

D.79. STNCOM.H

COMMON/STNCOM/ STNMAX, TMSTNU, TLDMIN, TMTLD
COMMON/STNCOM/ ISTNU, JSTNU, KSTNU, NSTNU, ITLDM, JTLDM, KTLDM, NTLDM

REAL*8 VARIABLES:

STNMAX	Largest Stanton number evaluated in this calculation.
TLDMIN	The minimum liquid temperature (for any heat structure) when subcooled boiling begins based on the Saha-Zuber correlation.
TMTLD	Time when TLDMIN was found.
TMSTNU	Time when STNMAX was evaluated.

INTEGER VARIABLES:

ISTNU	3D r- or x-cell number where STNMAX was evaluated.
JSTNU	3D θ- or y-cell number where STNMAX was evaluated.
KSTNU	3D z-level number where STNMAX was evaluated.
NSTNU	Component number where STNMAX was evaluated.
ITLDM	Variable not used.
JTLDM	Axial node number where TLDMIN was found.
KTLDM	Variable not used.
NTLDM	Component number where TLDMIN was found.

D.80. STRTNT.H

COMMON/STRTNT/ SDTINT, STFVL, STFVU, STFLL, STFLU, FSTRV, FSTRL

REAL*8 VARIABLES:

SDTINT	Variable not used.
STFVL	Variable not used.
STFVU	Variable not used.
STFLL	Constant used to determine stratified-flow weighting factors.
STFLU	Constant used to determine stratified-flow weighting factors.
FSTRV	Variable not used.
FSTRL	Multiplier on the liquid velocity check for stratified flow in subroutine CELLA3.

D.81. SUPRES.H

COMMON/SUPRES/ S

REAL*8 VARIABLE:

S	Factor in nucleate-boiling heat-transfer coefficient evaluation in subroutine CHEN.

D.82. SYSSUM.H

COMMON/SYSSUM/ ALQCOR, ALQPRZ, ALQUP, CORWM, PMX, TLMX, TLNCOR, TSHCOR, TSNCOR, TVMX, VOLCOR, XLQCOR, XTSHCR

COMMON/SYSSUM/ JPMX, JTLMX, JTVMX, NPMX, NTLMX, NTVMX

REAL*8 VARIABLES:

ALQCOR	Core-region mean liquid volume fraction.
ALQPRZ	PRIZER (pressurizer) mean liquid volume fraction.
ALQUP	Upper-plenum mean liquid volume fraction (evaluated only for 3D VESSELs).
CORWM	Core-region water mass.
PMX	Maximum pressure.
TLMX	Maximum liquid temperature.
TLNCOR	Core-region mean liquid temperature.
TSHCOR	Core region mean superheat.
TSNCOR	Core-region mean saturation temperature.
TVMX	Maximum gas temperature.
VOLCOR	Core-region volume.
XLQCOR	Minimum core-region liquid volume fraction.
XTSHCR	Maximum core-region superheat.

INTEGER VARIABLES:

JPMX	Cell number for the maximum pressure.
JTLMX	Cell number for the maximum liquid temperature.
JTVMX	Cell number for the maximum gas temperature.
NPMX	Component number for the maximum pressure.
NTLMX	Component number for the maximum liquid temperature.
NTVMX	Component number for the maximum gas temperature.

D.83. TEEOPT.H

COMMON/TEEOPT/ NOSRCE

INTEGER VARIABLE:

NOSRCE	Option to turn off momentum-source coupling between the main tube and side tube of a SEPD or TEE component.

0 = evaluate momentum-source coupling;
1 = turn off momentum-source coupling.

D.84. TF3DC.H

INTEGER	ORG
COMMON/TF3DC/	INSCT, IZ, KABSO, KCMSH, KL, KLEV, KU, ORG, KVEL1, KVEL2, KVEL3

INTEGER VARIABLES:

INSCT	Variable used to obtain a displacement into network arrays involving VESSEL junctions when there is more than one VESSEL.
IZ	VESSEL level number currently being evaluated.
KABSO	Storage offset to obtain an absolute cell number when multiple VESSELs are used.
KCMSH	Offset for coarse-mesh indexing with multiple VESSELs.
KL	Displacement of level (IZ–1) from level (IZ) in A-array storage for the VESSEL 3D data array.
KLEV	VESSEL component axial-direction K index [the axial-level number IZ plus NZBCM (for two lower pseudo-cell levels)].
KU	Displacement of level (IZ+1) from level (IZ) in A-array storage for the VESSEL 3D data array.
ORG	Starting location of the 3D VESSEL-component IZ level data in the A array.
KVEL1	Order of the r- or x-direction stabilizer motion-equation matrix for the present VESSEL component.
KVEL2	Order of the θ- or y-direction stabilizer motion-equation matrix for the present VESSEL component.
KVEL3	Order of the z-direction stabilizer motion-equation matrix for the present VESSEL component.

D.85. THERM.H

REAL*8	NTC
COMMON/THERM/	NTC, DIATC, ATC, VTC, AW, ATW, CKW
COMMON/THERM/	ITTC

REAL*8 VARIABLES:

NTC	Number of thermocouples per ROD or SLAB element.

DIATC	Diameter of thermocouple.
ATC	Area per unit length of thermocouple.
VTC	Volume per unit length of thermocouple.
AW	Area of ROD or SLAB element to thermocouple weld.
ATW	Thickness of ROD or SLAB element to thermocouple weld.
CKW	Thermal conductivity of the ROD or SLAB element-to-thermocouple weld.

INTEGER VARIABLE:

ITTC:	Thermocouple flag. 0 = no thermocouple on ROD or SLAB element; 1 = thermocouple present on ROD or SLAB element.

D.86. THERMV.H

COMMON/THERMV/ ISTRT3, IEND3, NVTHM, NDIMV1, NIXNJ, NSTHM

INTEGER VARIABLES:

ISTRT3	First cell number (ICO) in the VESSEL component r- or x-direction.
IEND3	Last cell number (ICX) in the VESSEL component r- or x-direction.
NVTHM	Number of different array parameters in the EQUIV common block for a VESSEL component.
NDIMV1	NVTHM times the total number of r- or x-direction calculation plus pseudo cells dimensioned for.
NIXNJ	NDIMV1 times the total number of θ- or y-direction calculation plus pseudo cells dimensioned for.
NSTHM	NI*NJ*NK stride between derivative pointer variables for a VESSEL component.

D.87. TMP.H

COMMON/TMP/	AFLUX(NK), ARLCK(NK), ARVCK(NK), S2A(NK), S2B(NK), S2C(NK), S2D(NK), S3A(NK), S3B(NK), S3C(NK), S3D(NK), S5A(NK), STDER(NK), STPRS(NK), XVOLL(NK), XVOLV(NK)
COMMON/TMP	LIFEQ(NK)
LOGICAL	LIFEQ

REAL*8 VARIABLES:

AFLUX	Net noncondensable-gas mass flow into the NK–NZBCM level mesh cell.
ARLCK	Net liquid mass flow into the NK–NZBCM level mesh cell.
ARVCK	Net gas mass flow into the NK–NZBCM level mesh cell.
S2A	Vectorization mask factor for defining the gas mass equation.
S2B	Vectorization mask factor for defining the gas volume fraction equal to 1.0.
S2C	Vectorization mask factor for defining the gas volume fraction equal to 0.0.
S2D	Vectorization mask factor for defining the vapor pressure equal to the saturation pressure based on the gas temperature.
S3A	Vectorization mask factor for defining the gas energy equation.
S3B	Vectorization mask factor for defining the liquid temperature equal to the gas temperature.
S3C	Vectorization mask factor for defining the liquid temperature equal to the saturation temperature based on the vapor pressure.
S3D	Vectorization mask factor for defining the gas temperature equal to the saturation temperature based on the vapor pressure.
S5A	Vectorization mask factor for defining the noncondensable-gas mass equation.
STDER	Derivative of the saturation temperature with respect to the total pressure based on the saturation temperature and saturation pressure.
STPRS	Saturation pressure based on the liquid temperature.
XVOLL	Fluid volume (NSTAB=0) or fluid volume minus liquid volume outflow during the timestep (NSTAB=1) in the NK–NZBCM level mesh cell.
XVOLV	Fluid volume (NSTAB=0) or fluid volume minus gas volume outflow during the timestep (NSTAB=1) in the NK–NZBCM level mesh cell.

LOGICAL VARIABLE:

LIFEQ	Fluid-phase flag that is false when two-phase fluid may become single phase. If this flag is false on the second pass through the linearization, the cell will be relinearized.

D.88. TOTALS.H

COMMON/TOTALS/ TLEN, TVOL

REAL*8 VARIABLES:

TLEN	Total length of a component.
TVOL	Total fluid volume of a component.

D.89. TSATCN.H

DIMENSION CEOSLP(40)

COMMON/TSATCN/ AEOS14, CEOS1, CEOS2, CEOS3, CEOSLP

COMMON/TSATCN/ IGAS, ILIQ

REAL*8 VARIABLES:

AEOS14	Constant in expression for saturation-temperature calculation at intermediate pressures (defined in subroutine THERMO).
CEOS1	First constant in expression for saturation-temperature calculation at intermediate pressures (defined in subroutine THERMO).
CEOS2	Second constant in expression for saturation-temperature calculation at intermediate pressures (defined in subroutine THERMO).
CEOS3	Third constant in expression for saturation-temperature calculation at intermediate pressures (defined in subroutine THERMO).
CEOSLP	Equation-of-state array for low pressures (defined in subroutine SETEOS).

INTEGER VARIABLES:

IGAS	Noncondensable-gas type option (Namelist variable). 1 = air; 2 = hydrogen; 3 = helium (ideal gas); 4 = helium (nonideal gas).
ILIQ	Condensable-fluid type option (variable not used).

D.90. TST3D.H

COMMON/TST3D/ CCIF

COMMON/TST3D/ I1D, NIFHT, NIFSH, NOBOIL, NOIMP, NWSH, IMOML

REAL*8 VARIABLE:

CCIF	Constant value for the interfacial-drag coefficient when NIFSH = 1 (Namelist variable).

INTEGER VARIABLES:

I1D	Flag to convert mean-mass and gas-mass equations to gas-mass and liquid-mass equations for evaluation by subroutine TF3DS.
NIFHT	Flag for defining a constant 10.0 value to the ALVE, CHTI, ALV, and CHTIA evaporation and condensation coefficients.
NIFSH	Interfacial-shear (drag) option flag (Namelist variable).
NOBOIL	Flag for not evaluating evaporation and condensation when IEOS = 0.
NOIMP	Flag for not evaluating the gas volume fraction time-derivative term in the motion equation. 0 = evaluate the gas volume fraction time-derivative term; 1 = do not evaluate the gas volume fraction time-derivative term (default).
NWSH	Flag for defining the gas FRIC by its gas-field value rather than by the liquid-field value.
IMOML	Option to improve momentum conservation where the gas volume fraction gradient is large. 0 = no (default); 1 = yes.

D.91. TWOSTEP.H

COMMON/TWOSTP/ NPSFE, NPSME, NTSPRN

INTEGER VARIABLES:

NPSFE	Pause in FEMOM and CIF3 if the timestep number NSTEP = NPSFE. The cell number is NPSJ or the level number is NPSIZ and the component number is NPICMP.
NPSME	Pause in STBME and STBMPL if the timestep number NSTEP = NPSME. The cell number is NPSJ and the component number is NPICMP.
NTSPRN	Flag for printing extra thermal-hydraulic parameter information to file TRCOUT.

D.92. VCKDAT.H

COMMON/VCKDAT/ DONTOL

COMMON/VCKDAT/ IPRVCK, ISKIP, ITVKMX

REAL*8 VARIABLE:

DONTOL — Tolerance for density difference requiring redonor celling in the VESSEL.

INTEGER VARIABLES:

IPRVCK — Flag to print information about redonor celling in the VESSEL (normally set to 0 for no print).

ISKIP — Flag to skip redonor-cell logic in the VESSEL component (normally set to 0 for no skip).

ITVKMX — Maximum iteration count to check for need to redonor cell in the VESSEL.

D.93. VDVMOD.H

COMMON/VDVMOD/ IVDVS1, IVDVS2

INTEGER VARIABLES:

IVDVS1 — Flag for scaling the $V \cdot \nabla V$ terms.
0 = no;
1 = yes.

IVDVS2 — Flag for scaling the $\beta V \cdot \nabla V$ terms.
0 = no;
1 = yes.

D.94. VELLIM.H

COMMON/VELLIM/ VVUB, VVLB, VLUB, VLLB, DFVUB, DFVLB, DFLUB, DFLLB

COMMON/VELLIM/ JVLIM

REAL*8 VARIABLES:

VVUB — Pump-impeller interface gas velocity upper-limit value.

VVLB — Pump-impeller interface gas velocity lower-limit value.

VLUB — Pump-impeller interface liquid velocity upper-limit value.

VLLB — Pump-impeller interface liquid velocity lower-limit value.

DFVUB — Derivative of the pump-impeller interface gas velocity (at its upper limit) with respect to total pressure.

DFVLB — Derivative of the pump-impeller interface gas velocity (at its lower limit) with respect to total pressure.

DFLUB Derivative of the pump-impeller interface liquid velocity (at its upper limit) with respect to total pressure.

DFLLB Derivative of the pump-impeller interface liquid velocity (at its lower limit) with respect to total pressure.

INTEGER VARIABLE:

JVLIM: For PUMP type IPMPTY = 0, the pump-impeller interface number (JVLIM = 2) when the PUMP component-action table defines the fluid velocity.

D.95. WEBNUM.H

COMMON/WEBNUM/ ALVFCP, ALVFCS, BMIN, CHTFCP, CHTFCS, CHTIBC, CHTIBH, CNDFC, DMIN, PENTL, PENTU, VLSPR, VVLOW, VVUP, WEB, WED, WEDU

COMMON/WEBNUM/ ICHVOL

REAL*8 VARIABLES:

ALVFCP Multiplier on ALV for low-velocity vertical components.

ALVFCS Multiplier on ALV under spray conditions.

BMIN Minimum allowed bubble size.

CHTFCP Multiplier on CHTI for low-velocity vertical components.

CHTFCS Multiplier on CHTI under spray conditions.

CHTIBC Vapor-bubble interfacial HTC when TV > TSAT.

CHTIBH Vapor-bubble interfacial HTC when TV < TSAT.

CNDFC Condensation-rate scaling factor.

DMIN Minimum allowed drop size.

PENTL Lower bound on entrained gas volume fraction α.

PENTU Upper bound on entrained gas volume fraction α.

VLSPR Lower limit on the quantity $(1-\alpha)V_\ell$ at the top of the cell above which the spray condition is assumed to exist.

VVLOW Lower limit on gas velocity for special condensation model for low-velocity vertical components.

VVUP Upper limit on gas velocity for special condensation model for low-velocity vertical components. Note: For liquid velocity greater than VLUP, the regular condensation model is used. For liquid velocity less than VLLOW, the special condensation model is used. For liquid velocity between VLLOW and VLUP, a linear interpolation between the two models is used.

WEB Bubble Weber number.

WED Droplet Weber number.

WEDU Droplet Weber number during core-region upflow (not implemented).

INTEGER VARIABLE:

ICHVOL: Flag that invokes a minimum value on the interfacial HTC.
0 = no effect (default);
1 = sets the minimum value to the cell volume times 1.0 x 10^7.

D.96. XTVCOM1.H

```
INTEGER                NVNAME1, NVNAME3, NVNAMEH, NVNAMEP
PARAMETER              (NVNAME1=26, NVNAME3=35, NVNAMEH=12,
                       NVNAMEP=10)
COMMON/XTVCOMC/VNAME1(NVNAME1), VNAME3(NVNAME3),
                       VNAMEH(NVNAMEH), VNAMEP(NVNAMEP)
```

LOGICAL VARIABLES:

VNAME1 XTV-graphics CHARACTER*30 variable names for 1D hydraulic components.

VNAME3 XTV-graphics CHARACTER*30 variable names for 3D VESSEL components.

VNAMEH XTV-graphics CHARACTER*30 variable names for heat-structure ROD or SLAB components.

VNAMEP XTV-graphics CHARACTER*30 variable names for PLENUM components.

D.97. XVOL.H

```
COMMON/XVOL/     BGSS, DAWL, DAXVL, DAXVU, DGSS, FREV
COMMON/XVOL/     IFVT, IFVTU, LDAX
```

REAL*8 VARIABLES:

BGSS Limits on special gas volume fraction prediction logic.

DAWL Weighting factors in special TF1DS flux logic.

DAXVL Lower-velocity limit on special TF1DS flux logic.

DAXVU Upper-velocity limit on special TF1DS flux logic.

DGSS Limits on special gas volume fraction prediction logic.

FREV Sensitivity level for reiteration on flow reversal.

INTEGER VARIABLES:

IFVT	Flag for setting velocities passed to TF1DS for special flux logic.
IFVTU	Time-of-velocity controller. 0 = XVSET logic uses the old-time velocity; 1 = XVSET logic uses the new-time velocity.
LDAX	Bypass switches on special TF1DS flux logic.

APPENDIX E

EXAMPLE OF MAKING CHANGES TO TRAC-M

E.1. INTRODUCTION

Programming changes are made to TRAC-M under the RCS supervised by the CVS. CVS is the front end of the RCS, which extends the notion of revision control from a collection of files in a single directory to a hierarchical collection of directories consisting of revision controlled files. These directories and files can be combined together to form a software release. CVS provides the functions necessary to manage these software releases and to control the concurrent editing of source files among multiple software developers. CVS keeps a single copy of the master sources. This copy is called the source repository. It contains all the information to permit extracting previous software releases based on either a symbolic revision tag or a date in the past.

Three commands under CVS enable a developer to make changes to TRAC-M.

- The developer uses the checkout (CO) command to create a copy of include and routine files from the CVS source-file repository. Then the developer makes programming changes to these files in a subdirectory of his working directory.
- The developer uses the commit (CI) command to save his programming changes to the include and routine files back into the CVS source-file repository after his programming changes have been tested, reviewed, and accepted.
- The developer uses the update command to merge his programming changes with the concurrent programming changes of other developers who have already committed their changes to the CVS source-file repository. Each developer is responsible to test his changes after the merge with the concurrent changes committed to the repository by the other developers.

This appendix discusses the following example of making changes to TRAC-M under the programming-change label UP1DPTR. We will be programming five new variable arrays DNEW, DNEWN, HYNEW, HTNEW, and INEW in TRAC-M for all 1D hydraulic components. These arrays are stored in the A array of blank common with integer pointer variables defining the element number of the A array where the first element of each of the new arrays is stored. Each pointer variable name begins with the letter L followed by its array name and is stored in COMMON /PTAB/. The UP1DPTR changes reprogram portions of COMMON/PTAB/ in the include files DUALPT.H, HYDROPT.H, HEATPT.H, and INTPT.H where LDNEW and LDNEWN, LHYNEW, LHTNEW, and LINEW are stored, respectively. Space for the number of elements in each of the new arrays is reserved in subroutine S1DPTR of file S1DPTR.H. Arrays DNEW, DNEWN, HTNEW, and INEW have NCELLS elements for their cell-centered parameters, whereas array HYNEW has NFACES = NCELLS+1 elements for its cell-edged parameter. These number of elements are reserved between the LDNEW, LDNEWN, LHYNEW, LHTNEW, and LINEW pointer values, and the values of the pointers of the arrays that follow them when the UP1DPTR changes to subroutine S1DPTR reserve their space in the A array.

Arrays DNEW, DNEWN, and HYNEW are assumed to be evaluated in subroutine FEMOM, whereas arrays HTNEW and INEW are assumed to be evaluated in subroutine CYLHT. Actual working equations for their evaluation will not be programmed by the UP1DPTR changes. They are defined by arbitrary REAL*8 constant values in this example as a substitute for their evaluation. These new arrays are passed to subroutines FEMOM and CYLHT through their argument list. The UP1DPTR changes program them in the argument lists of subroutines FEMOM and CYLHT as well as in the CALL FEMOM argument list in subroutine PREPER and in the CALL CYLHT argument list in subroutine POSTER.

DNEW and DNEWN are assumed to be the old-time and new-time derivative of density with respect to pressure, HYNEW is assumed to be reciprocal pressure, and HTNEW is assumed to be thermal conductivity. For real-valued variables that are input and/or output, TRAC-M needs to know their units internally for inputting/outputting their values in SI or English units. Units information on real-valued variables is programmed in TRAC-M by include file LABELV.H and subroutine file BLKDAT2.F. Files LABLEV.H and BLKDAT2.F are output by FORTRAN 77 program LABPRG.F based on current-version units information in file LABIN and new-update units information in file LABNEW that are input. File LABNEW input data for LABPRG.F, required by the UP1DPTR changes to TRAC-M, is listed in Section E.2. Program LABPRG.F, and its input/output files are described in Appendix F.

The DNEWN, HYNEW, HTNEW, and INEW results by the UP1DPTR changes are programmed in subroutines DCOMP, RECOMP, ECOMP, and SVSET1. For the restart capability of TRAC-M, subroutines DCOMP and RECOMP are programmed to write and read DNEWN, HYNEW, HTNEW, and INEW array data to the TRCDMP and from the TRCRST files, respectively. Subroutine ECOMP is programmed to output DNEWN, HYNEW, and HTNEW values to the TRCOUT file for each large edit. Subroutine SVSET1 is programmed to output a signal-variable parameter value from array DNEWN for the control procedure of TRAC-M. TRAC-M does this for all 1D hydraulic components.

Section E.3 shows a listing of a portion of the include and subroutine files with UP1DPTR changes to be programmed in TRAC-M. Changes to the TRAC-M manuals because of these programming changes are commented on initially. Note that in the programming changes to subroutine ECOMP, subroutine UNCNVT is called to convert a parameter's TRAC-internal SI-units values (stored temporarily in array TMP) to English units if IOOUT = 1 before the array values are output to the TRCOUT file with the parameter's units symbol LUNCB(IOOUT+1, ITLS(IOLD)).

E.2. INPUT-DATA FILE LABNEW FOR LABPRG.F

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345

  1  1  4  labnew data for up1dptr
luddendp    (kg/m3/pa) (lbm/ft3/psi)  4.30425636E+02  0.00000000E+00
d/p (kg/m3/pa) d/p (lb/ft3/p) luddendp
 dnew      luddendp
dnewn     luddendp
hynew     lurpress
htnew     luthcond
  0  0  0  labnew data end
```

E.3. UP1DPTR CHANGES TO TRAC-M

The output of new array variables DNEWN, HYNEW, HTNEW, and INEW to files TRCDMP and TRCOUT does not require documentation in the TRAC-M manuals. Appendix C of the TRAC-M Programmers Guide needs to have the five new pointer-variable names and their descriptions added to include files DUALPT.H, HYDROPT.H, HEATPT.H, and INTPT.H.

Section C.1.1. DUALPT.H

Name	Array	Dimension	Description
LDNEW	DNEW	NCELLS	Old-time derivative of density with respect to pressure.
LDNEWN	DNEWN	NCELLS	New-time derivative of density with respect to pressure.

Section C.1.2. HYDROPT.H

Name	Array	Dimension	Description
LHYNEW	HYNEW	NCELLS+1	Reciprocal of the pressure.

Section C.1.4. HEATPT.H

Name	Array	Dimension	Description
LHTNEW	HTNEW	NCELLS	Thermal conductivity.

Section C.1.3. INTPT.H

Name	Array	Dimension	Description
LINEW	INEW	NCELLS	Boundary-layer form index having a REAL*8 value.

The following programming changes, shown in bold type, need to be made to files DUALPT.H, HYDROPT.H, HEATPT.H, INPTP.H, LABELV.H, BLKDAT2.F, S1DPTR.F, DCOMP.F, RECOMP.F, ECOMP.F, SVSET1.F, FEMOM.F, PREPER.F, CYLHT.F, and POSTER.F. These files need to be (1) checked out of the CVS repository with the CVS CO "list of file names" command, (2) modified with the bold-type changes shown, and (3) committed back to (checked in to) the CVS repository with the CVS CI "list of file names" command.

Include file DUALPT.H changes
Add new old-time LDNEW and new new-time LDNEWN pointers

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345

      INTEGER   lalp,lalpn,lalpd,lalpdn,lalv,lalvn,lalve,lalven,          &
     & lara,laran,larel,lareln,larev,larevn,larl,larln,larv,larvn,lbit,   &
```

```
    & lbitn,lchti,lchtin,lchtia,lchtan,lcif,lcifn,lconc,lconcn,ld,ldn,   &
    & ldnew,ldnewn,lea,lean,lel,leln,lev,levn,lgam,lgamn,lhig,lhigo,    &
    & lhil,lhilo,lhiv,lhivo,lp,lpn,lpa,lpan,lqppc,lqppco,lroa,lroan,     &
    & lrol,lroln,lrov,lrovn,ls,lsn,ltce,ltcen,ltd,ltdn,ltl,ltln,ltv,     &
    & ltvn,ltw,ltwn,ltwa,ltwan,ltwe,ltwen,lvl,lvln,lvlt,lvlto,lvm,       &
    & lvmn,lvv,lvvn,lvvt,lvvto
!
      COMMON /ptab/lalp,lalpn,lalpd,lalpdn,lalv,lalvn,lalve,lalven,      &
    & lara,laran,larel,lareln,larev,larevn,larl,larln,larv,larvn,lbit,   &
    & lbitn,lchti,lchtin,lchtia,lchtan,lcif,lcifn,lconc,lconcn,ld,ldn,   &
    & ldnew,ldnewn,lea,lean,lel,leln,lev,levn,lgam,lgamn,lhig,lhigo,    &
    & lhil,lhilo,lhiv,lhivo,lp,lpn,lpa,lpan,lqppc,lqppco,lroa,lroan,     &
    & lrol,lroln,lrov,lrovn,ls,lsn,ltce,ltcen,ltd,ltdn,ltl,ltln,ltv,     &
    & ltvn,ltw,ltwn,ltwa,ltwan,ltwe,ltwen,lvl,lvln,lvlt,lvlto,lvm,       &
    & lvmn,lvv,lvvn,lvvt,lvvto
```

Include file HYDROPT.H changes

Add new hydrodynamic-calculation LHYNEW pointer

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345

      INTEGER    lalpmn,lalpmx,lalpo,lam,larc,lcfz,lcl,lcpl,lcpv,lcv,     &
    & ldalva,ldfldp,ldfvdp,ldriv,                                       &
    & ldtsdp,ldeldp,ldevdp,ldeldt,ldevdt,ldrolp,ldrovp,ldrolt,ldrovt,   &
    & lhvst,lhlst,ldhvsp,ldhlsp,ldtssp,ldevat,ldevap,ldrvap,ldrvat,     &
    & ldx,lelev,lfa,lfavol,lfinan,lfric,                                &
    & lfsmlt,lgrav,lgrvol,lh,lhd,lhdht,lhfg,lhgam,lhla,lhlatw,lhva,     &
    & lhvatw,lhynew,lqrl,lqrv,lqp3f,lqppp,lregnm,lrhs,lrmem,lrmvm,      &
    & lrarl,lrarv,lxsm,lysm,lzsm,lrsm,lr0sm,lnfvsm,lnflsm,luvsm,         &
    & lnfcvsm,lnfclsm,lvvsm,lvlsm,lnf1sm,lnf2sm,lnf3sm,lnfv4sm,lnfl4sm, &
    & lrom,lrvmf,lsig,ltrid,ltsat,ltssn,lvisl,lvisv,lvlalp,lvlvc,       &
    & lvlvol,lvlx,lvol,lvr,lvrv,lvvvol,lvvx,lwa,lwat,lwfl,lwfv,lwfmfl,  &
    & lwfmfv
!
      COMMON /ptab/lalpmn,lalpmx,lalpo,lam,larc,lcfz,lcl,lcpl,lcpv,lcv,  &
```

```
      & ldalva,ldfldp,ldfvdp,ldriv,                                       &
      & ldtsdp,ldeldp,ldevdp,ldeldt,ldevdt,ldrolp,ldrovp,ldrolt,ldrovt,  &
      & lhvst,lhlst,ldhvsp,ldhlsp,ldtssp,ldevat,ldevap,ldrvap,ldrvat,    &
      & ldx,lelev,lfa,lfavol,lfinan,lfric,                                &
      & lfsmlt,lgrav,lgrvol,lh,lhd,lhdht,lhfg,lhgam,lhla,lhlatw,lhva,    &
      & lhvatw,lhynew,lqrl,lqrv,lqp3f,lqppp,lregnm,lrhs,lrmem,lrmvm,     &
      & lrarl,lrarv,lxsm,lysm,lzsm,lrsm,lr0sm,lnfvsm,lnflsm,luvsm,       &
      & lnfcvsm,lnfclsm,lvvsm,lvlsm,lnf1sm,lnf2sm,lnf3sm,lnfv4sm,lnfl4sm, &
      & lrom,lrvmf,lsig,ltrid,ltsat,ltssn,lvisl,lvisv,lvlalp,lvlvc,      &
      & lvlvol,lvlx,lvol,lvr,lvrv,lvvvol,lvvx,lwa,lwat,lwfl,lwfv,lwfmfl, &
      & lwfmfv
      & lhvatw,lqrl,lqrv,lqp3f,lqppp,lregnm,lrhs,lrmem,lrmvm,lrarl,      &
      & lrarv,lxsm,lysm,lzsm,lrsm,lr0sm,lnfvsm,lnflsm,luvsm,lnfcvsm,     &
      & lnfclsm,lvvsm,lvlsm,lnf1sm,lnf2sm,lnf3sm,lnfv4sm,lnfl4sm,lrom,   &
      & lrvmf,lsig,ltrid,ltsat,ltssn,lvisl,lvisv,lvlalp,lvlvc,lvlvol,    &
      & lvlx,lvol,lvr,lvrv,lvvvol,lvvx,lwa,lwat,lwfl,lwfv,lwfmfl,lwfmfv
```

Include file HEATPT.H changes
Add new heat-calculation LHTNEW pointer

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345

       COMMON /ptab/lcpw,lcw,ldr,lemis,lhol,lhov,lhtnew,lrn,lrn2,        &
      & lrow,ltchf,ltol,ltov
!
       INTEGER      lcpw,lcw,ldr,lemis,lhol,lhov,lhtnew,lrn,lrn2,        &
      & lrow,ltchf,ltol,ltov
```

Include file INTPT.H changes
Add new integer LINEW pointer

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345
```

```
      COMMON /ptab/lidr,linew,lmatid,lnff,llccfl
!
      INTEGER      lidr,linew,lmatid,lnff,llccfl
```

Include file LABELV.H changes
See the include file LABELV.H listing in App. F, Sec. F.6 for the full listing of common block LABELV generated by program LABPRG.F.

Subroutine file BLKDAT2.F changes
See the subroutine file BLKDAT2.F listing in App. F, Sec. F.7 for the full listing of subroutine BLOCK DATA BLKDAT2 generated by program LABPRG.F.

Subroutine file S1DPTR.F changes
Initialize the newly added pointers in subroutine S1DPTR. Increment LENPTR by one for each pointer added in the appropriate section of S1DPTR. Adjust the length of the pointer initialized directly after each of the new pointers is added to reflect correct lengths.

```
         1         2         3         4         5         6         7
123456789012345678901234567890123456789012345678901234567890123456789012345

      ld(3)=ld(2)+0
      ldnew=ld(3)+0
       lea=ldnew+ncells
      lel=lea+ncells

      ldn(3)=ldn(2)+0
      ldnewn=ldn(3)+0
       lean=ldnewn+ncells
      leln=lean+ncells

      lnxt=ltcen+1
!      lenptr=86  before
      lenptr=88
!
```

```
!     set time update pointers

      lregnm=lwat+ncells
      lhynew=lregnm+nfaces
       lxsm=lhynew+nfaces
      lysm=lxsm+ncells

      lnxt=lvlsm+3*nfaces
!      lenptr=lenptr+82  before
      lenptr=lenptr+83
!     add pointers for radiation phasic heat fluxes.

        llccfl=lnff+nfaces
        linew=llccfl+nfaces
         lnxt=linew+ncells
!        lenptr=lenptr+4  before
        lenptr=lenptr+5
!

            ltov=ltol+ncells
            lhtnew=ltov+ncells
             lnxt=lhtnew+ncells
          ENDIF
!          lenptr=lenptr+12  before
           lenptr=lenptr+13
        ENDIF
```

Subroutine file DCOMP.F changes

Output the new variables to be dumped to the dump/restart file. Increment LVCNTR and LVEDGE by the number of cell-center and cell-edge variables being dumped, respectively.

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345

! lvcntr is the number of pointers for cell-center variables.
! lvedge is the number of pointers for cell-edge variables.
```

```
!
!       lvcntr=25  before
        lvcntr=28
!       lvedge=15  before
        lvedge=16
        IF (isolut.NE.0) lvcntr=lvcntr+2
```

```
        CALL bfout(a(ltcen),1,ictrld)
          CALL  bfout(a(ldnewn),ncellt,ictrld)
         CALL   bfout(a(lhynew),ncellt+1,ictrld)
         CALL   bfout(a(lhtnew),ncellt,ictrld)
         CALL   bfout(a(linew),ncellt,ictrld)
        IF (isolut.NE.0) THEN
```

Subroutine file RECOMP.F changes
Input the new variables from the dump/restart file in the same order that they were output.

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345
```

```
        CALL bfin(a(ltcen),1,ictrlr)
         CALL   bfin(a(bump+ldnewn),ncells,ictrlr)
         CALL   bfin(a(bump+lhynew),ncells+1,ictrlr)
         CALL   bfin(a(bump+lhtnew),ncells,ictrlr)
         CALL   bfin(a(bump+linew),ncells,ictrlr)
        IF (isolut.NE.0) THEN
```

Subroutine file ECOMP.F changes
Output the new variables to the TRCOUT file.

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345
```

```
          ENDIF
        ENDIF
!
```

```
!      print out hydraulic parameters dnewn and hynew
!
      jn=0
      DO n=1,nn
          j1=jstrt+(n-1)*10
          j2=min0(j1+9,jstop)
        j0=j1-1
        j3=j2-j0
        DO j=j1,j2
          jj=j-j0
          jm1=j-1
            tmp(jj,1)=a(ldnewn+jm1)
            tmp(jj,2)=a(lhynew+jm1)
        ENDDO
          CALL uncnvt('dnewn',tmp(1,1),j3,1,-ioout)
        i1=iold
          CALL uncnvt('hynew',tmp(1,2),j3,1,-ioout)
        IF (n.EQ.1) THEN
           IF (iunout.EQ.0) THEN
             WRITE (iout,450)
          ELSE
             WRITE (iout,450) luncb(ioout+1,itls(i1)),
     &                         luncb(ioout+1,itls(iold))
          ENDIF
 450       FORMAT (/9x,'ddendp',6x,'hynew'/' cell   ',2a)
        ENDIF
        j1=j1-j0
        j2=j2-j0
          WRITE (iout,455) (jn+j,(tmp(j,k),k=1,2),j=j1,j2)
 455     FORMAT (1x,i3,1x,1p,2e11.3)
        jn=jn+10
      ENDDO
      tmp(1,2)=a(lhynew+jstop)
      CALL uncnvt('hynew',tmp(1,2),1,1,-ioout)
      jn=jstop-jstrt+2
      WRITE (iout,460) jn,tmp(1,2)
```

```
 460  FORMAT  (1x,i3,12x,1p,e11.3)
!
!      print  out  heat-transfer  parameter  htnew
!
       IF  (nodes.NE.0)  THEN
         jn=0
         DO  n=1,nn
             j1=jstrt+(n-1)*10
             j2=min0(j1+9,jstop)
           j0=j1-1
           j3=j2-j0
           DO  j=j1,j2
             jj=j-j0
             jm1=j-1
               tmp(jj,1)=a(lhtnew+jm1)
           ENDDO
               CALL  uncnvt('htnew',tmp(1,1),j3,1,-ioout)
            IF  (n.EQ.1)  THEN
               IF  (iunout.EQ.0)  THEN
                 WRITE  (iout,470)
              ELSE
                   WRITE  (iout,470)  luncb(ioout+1,itls(iold))
              ENDIF
 470            FORMAT  (/9x,'htnew'/'  cell    ',a)
           ENDIF
           j1=j1-j0
           j2=j2-j0
             WRITE  (iout,475)  (jn+j,tmp(j,1),j=j1,j2)
  475       FORMAT  (1x,i3,1x,1p,e11.3)
           jn=jn+10
         ENDDO
      ENDIF
      RETURN
      END
```

Subroutine file SVSET1.F changes

Define DNEWN to be signal-variable parameter number 105.

```
          1         2         3         4         5         6         7
 12345678901234567890123456789012345678901234567890123456789012345

                                 ELSEIF (nsvn.EQ.7) THEN
!
!      isvn=103 : slab outer-surface heat loss (w)
!                 from the wall outside surface
!
                                   act(kpt+7)=htlsco
                                   GOTO 980
                                 ELSEIF (nsvn.EQ.8) THEN
!
!      isvn=104 : cell mixture temperature (k)
!
                                   IF (nstep.EQ.0) THEN

                                   ENDIF
                                   l=lvv-1
                                   DO i=1,ncellt

                                   ENDDO
                                   GOTO 880
                                   ELSEIF (nsvn.EQ.9) THEN
!
!      isvn=105 : cell d(density)/d(pressure) (kg/m**3/pa)
!
                                    l=ldnewn-1
                                    GOTO 880
                                 ENDIF
                               ELSEIF (nsvn.EQ.1) THEN
```

Subroutine file FEMOM.F changes

Assuming that DNEW, DNEWN, and HYNEW are all evaluated in FEMOM, add them to the argument list of FEMOM, add dimension statements in FEMOM, and perform their evaluation by assigning constants.

```
          1         2         3         4         5         6         7
 1234567890123456789012345678901234567890123456789012345678901234567890123456

      SUBROUTINE femom(alpo,alp,rov,rol,visv,visl,vl,vv,vlt,vvt,arv,      &
     & arl,vln,vvn,p,tsn,tln,tvn,dx,hd,fa,vol,wfv,wfl,cif,grav,bd1,bd2,  &
     & trid,aol,aov,drv,drl,od,kjn,ncells,vlto,vvto,dfldp,pa,tssn,sigm,  &
      & gam,rarl,rarv,nff,tchf,lccfl,xvlr,wfmfl,wfmfv,dnew,dnewn,hynew)
      IMPLICIT REAL*8 (a-h,o-z)

      REAL*8 nff(*),lccfl(*)
      DIMENSION alpo(1),alp(1),rov(1),rol(1),visv(1),visl(1),vl(1),vv     &
     & (1),vlt(1),vvt(1),arv(1),arl(1),vln(1),vvn(1),p(1),tsn(1),tln(1)  &
     & ,tvn(1),dx(1),hd(1),fa(1),vol(1),wfv(1),wfl(1),cif(1),grav(1),    &
     & bd1(nbd),bd2(nbd),trid(6,20),aol(kjn,1),aov(kjn,1),drv(1),drl(1)  &
     & ,od(4,20),vlto(1),vvto(1),dfldp(1),pa(1),tssn(1),sigm(1),gam(1),  &
      & rarl(1),rarv(1),tchf(1),wfmfl(1),wfmfv(1),dnew(1),dnewn(1),      &
     & hynew(1),xvlr(*)
!

      x4v=0.d0
      x4l=0.d0
!
!      define some values to dnew, dnewn, and hynew
!
        DO j=jstart,ncells
          dnew(j)   = 1.0d0
          dnewn(j)  = 2.0d0
        ENDDO
        DO j=jstart,ncp
          hynew(j)  = 3.0d0
        ENDDO
!
!   loop to set up the tridiagonal systems for the two
```

```
!   equations of motion, and perform the initial elimination.
!
      DO j=jstart,ncp
```

Subroutine file PREPER.F changes
Change the CALL FEMOM statement to include DNEW, DNEWN, and HYNEW in the argument list.

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345

      CALL femom(a(lalpo),a(lalp),a(lrov),a(lrol),a(lvisv),a(lvisl),a    &
     & (lvl),a(lvv),a(lvlt),a(lvvt),a(larv),a(larl),a(lvln),a(lvvn),a    &
     & (lp),a(ltsat),a(ltln),a(ltvn),a(ldx),a(lhd),a(lfa),a(lvol),a      &
     & (lwfv),a(lwfl),a(lcifn),a(lgrav),bd1,bd2,a(ltrid),a(jaol),a       &
     & (jaov),a(jdrv),a(jdrl),a(jod),jnjun,istop,a(lvlto),a(lvvto),a     &
     & (ldfldp),a(lpa),a(ltssn),a(lsig),a(lgam),a(lrarl),a(lrarv),a      &
      & (lnff),tchf,a(llccfl),a(lvr),a(lwfmfl),a(lwfmfv),a(ldnew),a      &
      &  (ldnewn),a(lhynew))
!***check sign convention here
      IF (.NOT.(nvcon.EQ.0.OR.nstab.EQ.0)) THEN
```

Subroutine file CYLHT.F changes
Similarly, assuming that HTNEW and INEW are evaluated in subroutine CYLHT, add them to the argument list of CYLHT, add dimension statements in CYLHT, and perform their evaluation by assigning constants.

```
         1         2         3         4         5         6         7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345

      SUBROUTINE cylht(t,tn,rn,rn2,dr,hil,hiv,til,tiv,hol,hov,tol,tov,    &
     & row,cpw,cw,qppp,a,b,nodes,ndm1,ncells,dt,istdy,qp3f,htnew,inew)
      IMPLICIT REAL*8 (a-h,o-z)
!
!     cylht calculates temperature fields in the radial direction
!     for cylindrical geometries
```

```
!      istdy=1 - implicit boundary conditions
!      istdy=0 - explicit boundary conditions
!
!
       INCLUDE 'constant.h'
!
       DIMENSION t(nodes,ncells),tn(nodes,ncells),rn(nodes),rn2(ndm1),dr   &
      & (ndm1),hil(ncells),hiv(ncells),til(ncells),tiv(ncells),hol         &
      & (ncells),hov(ncells),tol(ncells),tov(ncells),row(ndm1,ncells),     &
      & cpw(ndm1,ncells),cw(ndm1,ncells),qppp(nodes,ncells),qp3f(ncells),  &
       &  htnew(ncells),inew(ncells)
       DIMENSION a(nodes,3),b(nodes)
        REAL*8  inew
!
!       define  some  values  to  htnew  and  inew
!
        DO  j=1,ncells
           htnew(j)  =  4.0do
           inew(j)   =  5.0do
        ENDDO
       fts=zero
       fss=one
```

Subroutine file POSTER.F changes
Change the CALL CYLHT statement to include HTNEW and INEW in the argument list.

```
         1         2         3         4         5         6         7
12345678901234567890123456789012345678901234567890123456789012345678901234

          ENDIF
          CALL cylht(a(ltw+nods*(istm1)),a(ltwn+nods*(istm1)),a(lrn        &
     &     +ioff*nods),a(lrn2+ioff*ndm1),a(ldr+ioff*ndm1),a(lhil           &
     &     +istm1),a(lhiv+istm1),a(ltln+istm1),a(ltvn+istm1),a(lhol        &
     &     +istm1),a(lhov+istm1),a(ltol+istm1),a(ltov+istm1),a(lrow        &
     &     +istm1*ndm1),a(lcpw+istm1*ndm1),a(lcw+istm1*ndm1),a(lqppp       &
     &     +istm1*nods),a(ldum1),a(ldum2),nods,ndm1,ncells,deltht,         &
```

```
     &        istdy,a(lqp3f+istm1),a(lhtnew(istm1),a(linew+istm1))
!
!    readjust prizer hiv back to its original value.
```

APPENDIX F

LABPRG FOR UPDATING UNITS LABELS IN TRAC-M

F.1. INTRODUCTION

FORTRAN 77 (F77) program LABPRG.F reads information from input-data files LABIN and LABNEW on current and new real-valued variables and their associated SI/English units for TRAC-M I/O. LABPRG.F writes the combined current and new real-valued variables and their units information to a new input-data file LABINN and writes replacement coding in file LABELV.H for INCLUDE LABELV.H and in file BLKDAT2.F for subroutine BLOCK DATA BLKDAT2.

F.2. LABPRG.F INPUT DATA

The real-valued variables and their SI/English units information for the current version of TRAC-M are input to LABPRG.F by file LABIN. A listing of file LABIN for Version 1.10+ of TRAC-M is provided in Section F.5. New real-valued variables and their SI/English units information required by new I/O statements in TRAC-M are input to LABPRG.F by input-data file LABNEW. The file-LABNEW input-data format for the new real-valued variables for TRAC-M I/O follows.

Card Number 1. (3I3, 2X, A37) NNUL, NNSV, NNVN, Message

Columns	Variable	Description
1–3	NNUL	Number of new units-label names that need their SI and English units symbols, SI-to-English factor, and SI-to-English shift offset values defined.
4–6	NNSV	Number of new signal variables that need their descriptive labels having SI- and English-unit symbols and their units-label names defined.
7–9	NNVN	Number of new real-valued FORTRAN variable names involved in TRAC I/O and their units-label names that need to be defined.

12–48	Message	"labnew data required by uuuuuu"

The message "labnew data required by uuuuuu" after the three values identifies this block of LABNEW data as being required by programming changes labeled by the name uuuuuu.

Input Card Number 2 for I = 1, 2, ..., NNUL (Omit Card Number 2 if NNUL = 0).

Card Number 2. (A8, 1X, A13, 1X, A13, 1X, E15.8, 1X, E15.8) LABC(I), LUPCB(1,I), LUPCB(2,I), FACTOR(I), OFFSET(I)

Columns	Variable	Description
1–8	LABC(I)	New units-label name with the form LUxxxxxx that is not already defined in Table 6-2 of the TRAC-M Users Guide. This units-label name is required by the update to define the SI/English units of a new signal-variable parameter or real-valued variable name required for TRAC I/O. (The first two letters of the name must be LU, and the last one-to-six letters are the update developer's choice but must be different from the letters already used by units-label names in Table 6-2.)
10–22	LUPCB(1,I)	SI-units symbol within parentheses and right justified.
24–36	LUPCB(2,I)	English-units symbol within parentheses and right justified.
38–52	FACTOR(I)	Factor value applied to the SI-units value to convert it to its English-units value (before the translational shift).
54–68	OFFSET(I)	Translational-shift offset value applied to the FACTOR(I) times SI-units value to convert it to its English-units value.

Input Card Number 3 for I = 1, 2, ..., NNSV (Omit Card Number 3 if NNSV = 0).

Card Number 3. (A14, 1X, A14, 1X, A8) SV(1,I), SV(2,I), LABS(I)

Columns	Variable	Description
1–14	SV(1,I)	Signal-variable parameter descriptive label with its SI-units symbol in parentheses and left justified.
16—29	SV(2,I)	Signal-variable parameter descriptive label with its English-units symbol in parentheses and left justified.
31–38	LABS(I)	Units-label name with form LUxxxxxx defining the units of the signal-variable parameter.

Input Card Number 4 for I = 1, 2, ..., NNVN (Omit Card Number 4 if NNVN=0).

Card Number 4. (A8, 1X, A8, 1X, I1) LABELS(I), TLABELS(I), IDEL

Columns	Variable	Description
1–8	LABELS(I)	Real-valued variable name involved in TRAC-M I/O that needs to have its SI and English units defined in TRAC-M for the coding changes of the update (real-valued variable names presently not involved in TRAC-M I/O but having units that need to be documented to better understand the coding in TRAC-M should be defined as well).
10–17	TLABELS(I)	Units-label name with form LUxxxxxx defining the units of the real-valued variable name LABELS(I).
19	IDEL	Option parameter for deleting a real-valued variable name from the variable names already defined in file LABIN [when IDEL = 1, define LABELS(I) and TLABELS(I) the same as they are defined in file LABIN in order to delete the variable name from file LABINN; when IDEL = 0, 0 doesn't need to be input specified because LABPRG will read a blank field in column 19 as a 0]. 0 = no; 1 = yes.

Card Numbers 1 to 4 define a block of variable-units information for each set of programming changes made to the current version of TRAC-M. The following Last Card is defined after zero or more of such data blocks.

Last Card (3I3, 2X, A15) IZ, IZ, IZ, labnew data end
where IZ = 0 with message "labnew data end".

The following listing is an example of file LABNEW data for programming changes made in App. E with the labeled name UP1DPTR. One new units-label name, one new signal-variable parameter, and 4 new real-valued variable names are defined. LABPRG.F adds this information to file LABIN to create file LABINN.

```
                   1         2         3         4         5         6         7         8
  12345678901234567890123456789012345678901234567890123456789012345678901234567890

1   1  1  4  labnew data required by up1dptr
2 luddendp    (kg/m3/pa) (lbm/ft3/psi)  4.30425636D+02  0.00000000D+00
3 d/p (kg/m3/pa) d/p (lb/ft3/p) luddendp
4 dnew      luddendp
5 dnewn     luddendp
6 hynew     lurpress
7 htnew     luthcond
8   0  0  0  labnew data end
```

F.3. LABPRG.F OUTPUT DATA

F77 program LABPRG.F begins by reading file LABIN to obtain all currently defined variable-units information for the current version of TRAC-M. LABPRG.F then reads file LABNEW and incorporates its new variable-units information from programming changes into the data from file LABIN. The LABNEW data are checked for appropriateness, and warning messages are written to file LABELV.H if errors are detected. If one or more LABNEW input-data errors are detected, the execution of LABPRG.F ends. If no LABNEW input-data errors are detected, LABPRG.F continues and writes replacement coding in file LABELV.H for INCLUDE LABELV.H and in file BLKDAT2.F for subroutine BLOCK DATA

BLKDAT2. Listings of files LABELV.H and BLKDAT2.F, based on file LABIN data from Version 1.10+ of TRAC-M and file LABNEW data from UP1DPTR, are provided in Sections F.6 and F.7, respectively.

LABPRG.F writes the combined variable-units information from files LABIN and LABNEW to file LABINN (LABIN New). File LABINN is the pending replacement input-data file for file LABIN for input to LABPRG.F for the next version of TRAC-M based on UP1DPTR programming changes. The listing of file LABINN is similar to the listing for file LABIN in Section F.5 except for:

```
file-LABNEW line 2       is inserted after file-LABIN line  50,
file-LABNEW line 3       is inserted after file-LABIN line 156,
file-LABNEW line 4 and 5 are inserted after file-LABIN line 301,
file-LABNEW line 6       is inserted after file-LABIN line 460, and
file-LABNEW line 7       is inserted after file-LABIN line 452.
```

LABPRG.F keeps the real-valued variable names in alphabetical order for output to files LABINN and BLKDAT2.F.

F.4. ARCHIVE FILES

The current files LABPRG.F, LABPRG.XCRAY (LABPRG executable on a Cray computer), LABPRG.XSUN (LABPRG executable on a SUN workstation), LABIN, LABNEW, LABELV.H, and BLKDAT2.F are stored in archive file ARLAB77 in the Los Alamos Common File System (CFS). Blocks of LABNEW data from each developer's programming changes are added successively to the LABNEW file. For version control and quality assurance of this TRAC-M support software and data when a release version of TRAC-M is generated, the TRAC-code custodian replaces file LABIN with file LABINN renamed LABIN, archives the release version's LABNEW file, replaces the LABNEW file with a file named LABNEW having the single record `0    0    0 LABNEW DATA END`, and resaves files LABIN and LABNEW to archive file ARLAB77.

The current TRAC-M source files are stored in their own subdirectory at this CFS location by a an RCS supervised by a CVS. Each developer needs to do the following to program-changed SI/English units information in TRAC-M when making programming changes to the TRAC-M source files. The LABELV.H and BLKDAT2.F files in the CVS-repository subdirectory are removed by the CO command, these files in your local directory are replaced by the LABELV.H and BLKDAT2.F files output by LABPRG.F, and then these revised files are saved in the CVS-repository subdirectory using the CI command.

F.5. LISTING OF FILE LABIN

```
          1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890

 1 lunounit            (-)            (-)  1.00000000D+00  0.00000000D+00
 2 lutime              (s)            (s)  1.00000000D+00  0.00000000D+00
 3 lutemp              (k)            (f)  1.80000000D+00 -4.59670000D+02
 4 lutempd             (k)            (f)  1.80000000D+00  0.00000000D+00
 5 lulength            (m)           (ft)  3.28083990D+00  0.00000000D+00
 6 luarea             (m2)          (ft2)  1.07639104D+01  0.00000000D+00
 7 luvolume           (m3)          (ft3)  3.53146667D+01  0.00000000D+00
 8 luvel             (m/s)         (ft/s)  3.28083990D+00  0.00000000D+00
 9 luacc            (m/s2)        (ft/s2)  3.28083990D+00  0.00000000D+00
10 lupumphd        (m2/s2)  (lbf*ft/lbm)  3.34552563D-01  0.00000000D+00
11 luvolflw         (m3/s)          (gpm)  1.58503222D+04  0.00000000D+00
12 luspvol         (m3/kg)      (ft3/lbm)  1.60184634D+01  0.00000000D+00
13 lumass             (kg)          (lbm)  2.20462262D+00  0.00000000D+00
14 lumassfw         (kg/s)       (lbm/hr)  7.93664144D+03  0.00000000D+00
15 lumfwrat        (kg/s2)       (lbm/s2)  2.20462262D+00  0.00000000D+00
16 lumassfx      (kg/m2/s)   (lbm/ft2/hr)  7.37338117D+02  0.00000000D+00
17 luvapgen      (kg/m3/s)   (lbm/ft3/hr)  2.24740658D+02  0.00000000D+00
18 luden           (kg/m3)      (lbm/ft3)  6.24279606D-02  0.00000000D+00
19 luddendt      (kg/m3/k)    (lbm/ft3/f)  3.46822003D-02  0.00000000D+00
20 luidrag         (kg/m4)      (lbm/ft4)  1.90280424D-02  0.00000000D+00
21 lupressa           (pa)         (psia)  1.45037738D-04  0.00000000D+00
22 lupressd           (pa)         (psid)  1.45037738D-04  0.00000000D+00
23 luprsrat         (pa/s)        (psi/s)  1.45037738D-04  0.00000000D+00
24 luminert        (kg*m2)      (lbm*ft2)  2.37303604D+01  0.00000000D+00
25 lutorque        (pa*m3)       (lbf*ft)  7.37562149D-01  0.00000000D+00
26 lubtork    (pa*m3*s/rad)  (lbf*ft/rpm)  7.72373277D-02  0.00000000D+00
27 luctork   (pa*m3*s2/r2) (lbf*ft/rpm2)  8.08827404D-03  0.00000000D+00
```

```
28 lupower            (w)        (btu/hr)  3.41214163D+00  0.00000000D+00
29 lupowrat         (w/s)      (btu/hr/s)  3.41214163D+00  0.00000000D+00
30 lulinhts         (w/m)     (btu/ft/hr)  1.04002077D+00  0.00000000D+00
31 luheatfx        (w/m2)    (btu/ft2/hr)  3.16998331D-01  0.00000000D+00
32 luvolhts        (w/m3)    (btu/ft3/hr)  9.66210912D-02  0.00000000D+00
33 luthcond       (w/m/k)   (btu/ft/f/hr)  5.77789317D-01  0.00000000D+00
34 luhtc         (w/m2/k)   (btu/ft2/f/h)  1.76110184D-01  0.00000000D+00
35 luihttf          (w/k)      (btu/f/hr)  1.89563424D+00  0.00000000D+00
36 luenergy         (w*s)           (btu)  9.47817120D-04  0.00000000D+00
37 luspener      (w*s/kg)       (btu/lbm)  4.29922614D-04  0.00000000D+00
38 luspheat    (w*s/kg/k)     (btu/lbm/f)  2.38845897D-04  0.00000000D+00
39 lurtime          (1/s)           (1/s)  1.00000000D+00  0.00000000D+00
40 lurtemp          (1/k)           (1/f)  5.55555556D-01  0.00000000D+00
41 lurmass         (1/kg)         (1/lbm)  4.53592370D-01  0.00000000D+00
42 lurpress        (1/pa)         (1/psi)  6.89475729D+03  0.00000000D+00
43 luspeed        (rad/s)           (rpm)  9.54929659D+00  0.00000000D+00
44 luradacc      (rad/s2)         (rpm/s)  9.54929659D+00  0.00000000D+00
45 luangle          (rad)           (deg)  5.72957795D+01  0.00000000D+00
46 luburnup     (mwd/mtu)       (mwd/mtu)  1.00000000D+00  0.00000000D+00
47 luenfiss    (mev/fiss)      (mev/fiss)  1.00000000D+00  0.00000000D+00
48 lugapgas     (g-moles)       (g-moles)  1.00000000D+00  0.00000000D+00
49 lurtmsq         (1/k2)          (1/f2)  3.08641975D-01  0.00000000D+00
50 lunitnam           (*)             (*)  1.00000000D+00  0.00000000D+00
51 luserdef           (*)             (*)  1.00000000D+00  0.00000000D+00
52 time (s)        time (s)        lutime
53 core power (w) power (btu/hr) lupower
54 pri press (pa) p press (psia) lupressa
55 pzr press (pa) pz pres (psia) lupressa
56 prizr temp (k) prizr temp (f) lutemp
57 pz liq lev (m) pz lq lev (ft) lulength
58 tk liq lev (m) tk lq lev (ft) lulength
59 hot-l temp (k) hot-l temp (f) lutemp
60 cld-l temp (k) cld-l temp (f) lutemp
61 p mflow (kg/s) mflow (lbm/hr) lumassfw
62 ecc mfw (kg/s) eccmf (lbm/hr) lumassfw
63 sec press (pa) sc pres (psia) lupressa
```

64	st mflw (kg/s)	st mf (lbm/hr)	lumassfw
65	fw mflw (kg/s)	fw mf (lbm/hr)	lumassfw
66	afw mfw (kg/s)	af mf (lbm/hr)	lumassfw
67	sc liq lev (m)	sc lq lev (ft)	lulength
68	user defined	user defined	lunounit
69	user defined	user defined	lunounit
70	core power (w)	power (btu/hr)	lupower
71	pwr period (s)	pwr period (s)	lutime
72	liq level (m)	liq level (ft)	lulength
73	pressure (pa)	press (psia)	lupressa
74	gas temp (k)	gas temp (f)	lutemp
75	liq temp (k)	liq temp (f)	lutemp
76	in sf temp (k)	in sf temp (f)	lutemp
77	surf temp (k)	surf temp (f)	lutemp
78	htstr temp (k)	htstr temp (f)	lutemp
79	void fraction	void fraction	lunounit
80	yt v mf (kg/s)	y vmf (lbm/hr)	lumassfw
81	z vp mf (kg/s)	z vmf (lbm/hr)	lumassfw
82	xr v mf (kg/s)	x vmf (lbm/hr)	lumassfw
83	yt l mf (kg/s)	y lmf (lbm/hr)	lumassfw
84	z lq mf (kg/s)	z lmf (lbm/hr)	lumassfw
85	xr l mf (kg/s)	x lmf (lbm/hr)	lumassfw
86	yt g vel (m/s)	y g vel (ft/s)	luvel
87	z gs vel (m/s)	z g vel (ft/s)	luvel
88	xr g vel (m/s)	x g vel (ft/s)	luvel
89	yt l vel (m/s)	y l vel (ft/s)	luvel
90	z lq vel (m/s)	z l vel (ft/s)	luvel
91	xr l vel (m/s)	x l vel (ft/s)	luvel
92	dis solute/liq	dis solute/liq	lunounit
93	pm spd (rad/s)	pm speed (rpm)	luspeed
94	valve farea fr	valve farea fr	lunounit
95	valve stem pos	valve stem pos	lunounit
96	mult cnst keff	mult cnst keff	lunounit
97	prog reac	prog reac	lunounit
98	tot fdbk reac	tot fdbk reac	lunounit
99	fuel temp reac	fuel temp reac	lunounit

```
100 cool temp reac cool temp reac lunounit
101 void frac reac void frac reac lunounit
102 solute reac    solute reac    lunounit
103 av fl temp (k) av fl temp (f) lutemp
104 av cl temp (k) av cl temp (f) lutemp
105 avg void fr    avg void fr    lunounit
106 avg sol (ppm)  avg sol (ppm)  lunounit
107 trp signal (*) trp signal (*) luserdef
108 trp set status trp set status lunounit
109 prompt pwr (w) pt pw (btu/hr) lupower
110 decayh pwr (w) dh pw (btu/hr) lupower
111 a mx sf tp (k) a mx sf tp (f) lutemp
112 s mx sf tp (k) s mx sf tp (f) lutemp
113 pmp hd (m2/s2) ph(lbf*ft/lbm) lupumphd
114 torque (pa*m3) torq (lbf*ft)  lutorque
115 p msou (m2/s2) ms(lbf*ft/lbm) lupumphd
116 vlve h dia (m) vlv h dia (ft) lulength
117 yt hyd dia (m) y hyd dia (ft) lulength
118 z hyd diam (m) z hyd dia (ft) lulength
119 xr hyd dia (m) x hyd dia (ft) lulength
120 yt m mf (kg/s) y mmf (lbm/hr) lumassfw
121 z m mfw (kg/s) z mmf (lbm/hr) lumassfw
122 xr m mf (kg/s) x mmf (lbm/hr) lumassfw
123 yt m vel (m/s) y m vel (ft/s) luvel
124 z mx vel (m/s) z m vel (ft/s) luvel
125 xr m vel (m/s) x m vel (ft/s) luvel
126 vp den (kg/m3) v dn (lbm/ft3) luden
127 lq den (kg/m3) l dn (lbm/ft3) luden
128 mi den (kg/m3) m dn (lbm/ft3) luden
129 ng den (kg/m3) ng d (lbm/ft3) luden
130 ngas mass (kg) ng mass (lbm)  lumass
131 ng press (pa)  ng pres (psia) lupressa
132 ng ie (w*s/kg) ng e (btu/lbm) luspener
133 vp ie (w*s/kg) v ie (btu/lbm) luspener
134 lq ie (w*s/kg) l ie (btu/lbm) luspener
135 sat temp s (k) sat temp s (f) lutemp
```

```
136 sat temp t (k) sat temp t (f) lutemp
137 vcv (w*s/kg/k) vc (btu/lbm/f) luspheat
138 lcv (w*s/kg/k) lc (btu/lbm/f) luspheat
139 ht vp (w*s/kg) htvp (btu/lbm) luspener
140 shloss vap (w) shlsv (btu/hr) lupower
141 shloss liq (w) shlsl (btu/hr) lupower
142 inf ht flw (w) ihtfw (btu/hr) lupower
143 v htc (w/m2/k) (btu/ft2/f/hr) luhtc
144 l htc (w/m2/k) (btu/ft2/f/hr) luhtc
145 v htc (w/m2/k) (btu/ft2/f/hr) luhtc
146 l htc (w/m2/k) (btu/ft2/f/hr) luhtc
147 ia*vhtc (w/k)  avh (btu/f/hr) luihttf
148 ia*lhtc (w/k)  alh (btu/f/hr) luihttf
149 yt idc (kg/m4) yidc (lbm/ft4) luidrag
150 z idc (kg/m4)  zidc (lbm/ft4) luidrag
151 xr idc (kg/m4) xidc (lbm/ft4) luidrag
152 ps den (kg/m3) ps d (lbm/ft3) luden
153 vgen (kg/m3/s) vg(lbm/ft3/hr) luvapgen
154 is ht loss (w) is hl (btu/hr) lupower
155 os ht loss (w) os hl (btu/hr) lupower
156 c mix temp (k) c mix temp (f) lutemp
157 signalvariable signalvariable lunounit
158 ads       luarea
159 aeffmi    luminert
160 alp       lunounit
161 alpha     lunounit
162 alphap    lunounit
163 alpin     lunounit
164 alpn      lunounit
165 alpoff    lunounit
166 alptb     lunounit
167 alreac    lunounit
168 alv       luihttf
169 alve      luihttf
170 alven     luihttf
171 am        lumass
```

172	amh2	lumass
173	amncss	luserdef
174	amxcss	luserdef
175	angl	luangle
176	apowr	lulinhts
177	area	luarea
178	arln	luden
179	arvn	luden
180	atork	lutorque
181	atw	lulength
182	avent	luarea
183	avlve	luarea
184	aw	lulength
185	bcr0	luden
186	bcr1	luden
187	beffmi	luminert
188	belv	lulength
189	beta	lunounit
190	bpp0	luden
191	bpp1	luddendt
192	bsa	lumass
193	bsmass	lumass
194	btork	lubtork
195	burn	luburnup
196	bxa	lumassfw
197	bxmass	lumassfw
198	bxsm	lulength
199	bysm	lulength
200	bzsm	lulength
201	cb	luserdef
202	cbcon1	luserdef
203	cbcon2	luserdef
204	cbdt	lutime
205	cbeta	lunounit
206	cbftab	luserdef
207	cbgain	luserdef

208	cbtau	lutime
209	cbwt	lunounit
210	cbxmax	luserdef
211	cbxmin	luserdef
212	ccflc	lunounit
213	ccflm	lunounit
214	ccif	luidrag
215	cdgn	lupower
216	cdhn	lupower
217	ceffmi	luminert
218	cemfr	lumassfw
219	cener	luenergy
220	cepwn	lupower
221	cfmass	lumass
222	cfrlxr	lunounit
223	cfrlyt	lunounit
224	cfrlz	lunounit
225	cfrvxr	lunounit
226	cfrvyt	lunounit
227	cfrvz	lunounit
228	cfz	lunounit
229	cfz3	lunounit
230	cfzlxr	lunounit
231	cfzlyt	lunounit
232	cfzlz	lunounit
233	cfzvxr	lunounit
234	cfzvyt	lunounit
235	cfzvz	lunounit
236	chm12	lunounit
237	chm13	lunounit
238	chm14	lunounit
239	chm15	lunounit
240	chm22	lunounit
241	chm23	lunounit
242	chm24	lunounit
243	chm25	lunounit

244	chti	luihttf
245	chtia	luihttf
246	chtin	luihttf
247	cif	luidrag
248	cifn	luidrag
249	cifxr	luidrag
250	cifyt	luidrag
251	cifz	luidrag
252	cimfrl	lumassfw
253	cimfrv	lumassfw
254	cks	lunounit
255	ckw	luthcond
256	clenn	lulength
257	cmass	lumass
258	cmflow	lumassfw
259	cnmax	lunounit
260	cnmin	lunounit
261	cntlmn	lutemp
262	cntlmx	lutemp
263	coef1	lunounit
264	coef2	lunounit
265	cof3sq	lunounit
266	comfrl	lumassfw
267	comfrv	lumassfw
268	conc	lunounit
269	concin	lunounit
270	conctb	lunounit
271	conctbab	lunounit
272	cond	luthcond
273	conoff	lunounit
274	conscl	lunounit
275	cosangl	lunounit
276	cosp	lunounit
277	coss	lunounit
278	cost	lunounit
279	cp	luspheat

280	cpow	lupower
281	cpowr	lunounit
282	cpowrabs	lunounit
283	cputot	lutime
284	cpvint	lutime
285	crliqfr	lunounit
286	crpress	lupressa
287	ctork	luctork
288	ctrans	lunounit
289	dbreac	lunounit
290	dcflow	lumassfw
291	dclqvol	lunounit
292	dds	lulength
293	delt	lutime
294	deltap	lupressd
295	deltl	lutempd
296	deltv	lutempd
297	dh	lulength
298	dia	lulength
299	diah	lulength
300	dmass	lumass
301	dmpint	lutime
302	dpcvn	lupressd
303	dpmax	lupressd
304	dpovn	lupressd
305	dprmax	lunounit
306	dr	lulength
307	dt	luangle
308	dtend	lutime
309	dtlmax	lutempd
310	dtmax	lutime
311	dtmin	lutime
312	dtrmax	lutempd
313	dtsm	lutime
314	dtsmax	lutempd
315	dtsofs	lunounit

316	dtsoft	lutime
317	dtsp	lutime
318	dtstrt	lutime
319	dtvmax	lutempd
320	dtxht	lutempd
321	dx	lulength
322	dxin	lulength
323	dy	lulength
324	dz	lulength
325	dznht	lulength
326	edh	lunounit
327	edint	lutime
328	effdsn	lunounit
329	effld	lunounit
330	effmi	luminert
331	effmi1	luminert
332	effstg	lunounit
333	efgen	lunounit
334	elev	lulength
335	emcif1	lunounit
336	emcif2	lurtemp
337	emcif3	lurtmsq
338	emcof1	lunounit
339	emcof2	lurtemp
340	emcof3	lurtmsq
341	enin1	luenergy
342	enin2	luenergy
343	eninp	luenergy
344	enth	luspener
345	epso	lunounit
346	epss	lunounit
347	epsw	lulength
348	errsm	lunounit
349	extsou	lupower
350	fa	luarea
351	favlve	lunounit

352	faxr	luarea
353	fayt	luarea
354	faz	luarea
355	fdfhl	lunounit
356	felv	lulength
357	ff	lunounit
358	fisphi	lunounit
359	flow	lumassfw
360	flowarea	luarea
361	flowin	lumassfw
362	flwin	lumassfw
363	flwoff	lumassfw
364	flwou	lumassfw
365	fmaxov	lunounit
366	fminov	lunounit
367	fp235	lunounit
368	fp238	lunounit
369	fp239	lunounit
370	fpuo2	lunounit
371	frcvn	lunounit
372	frfaxr	lunounit
373	frfayt	lunounit
374	frfaz	lunounit
375	fric	lunounit
376	fricr	lunounit
377	frovn	lunounit
378	frvol	lunounit
379	fsi	lunounit
380	fsmass	lumass
381	fso	lunounit
382	ftd	lunounit
383	ftx	luserdef
384	fty	lunounit
385	fucrac	lunounit
386	funh	lunounit
387	fxmass	lumassfw

388	fxsm	lulength
389	fysm	lulength
390	fzsm	lulength
391	gam	luvapgen
392	gamma	lunounit
393	gc	luacc
394	gfint	lutime
395	gmix	lunounit
396	gmles	lugapgas
397	grav	lunounit
398	gravxr	luacc
399	gravyt	luacc
400	gravz	luacc
401	gvf	lunounit
402	gxrc	lunounit
403	gytc	lunounit
404	gzc	lunounit
405	hbs	lulength
406	hd	lulength
407	hd-ht	lulength
408	hd3	lulength
409	hdm	lunounit
410	hdri	lulength
411	hdro	lulength
412	hdxr	lulength
413	hdyt	lulength
414	hdz	lulength
415	head	lupumphd
416	height	lulength
417	hgam	luheatfx
418	hgap	luhtc
419	hgapo	luhtc
420	hil	luhtc
421	hilg	luhtc
422	hiv	luhtc
423	hivg	luhtc

424	hl	luhtc
425	hli	luhtc
426	hlo	luhtc
427	holg	luhtc
428	houtl	luhtc
429	houtl1	luhtc
430	houtl2	luhtc
431	houtv	luhtc
432	houtv1	luhtc
433	houtv2	luhtc
434	hovg	luhtc
435	hs	lunounit
436	hsabs	lunounit
437	hsk	lulength
438	hsp1	lunounit
439	hsp2	lunounit
440	hsp3	lunounit
441	hsp4	lunounit
442	hstn	lutemp
443	htcwl	luhtc
444	htcwv	luhtc
445	htlsci	lupower
446	htlsco	lupower
447	htlsgi	lupower
448	htlsgo	lupower
449	htmli	lunounit
450	htmlo	lunounit
451	htmvi	lunounit
452	htmvo	lunounit
453	htp1	lunounit
454	htp2	lunounit
455	htp3	lunounit
456	htp4	lunounit
457	hv	luhtc
458	hvi	luhtc
459	hvlve	lulength

460	hvo	luhtc
461	imflow	lumass
462	inrta	luminert
463	lamda	lurtime
464	lamdh	lurtime
465	mflow	lumassfw
466	mfrl	lumassfw
467	mfrlr	lumassfw
468	mfrlt	lumassfw
469	mfrlz	lumassfw
470	mfrv	lumassfw
471	mfrvr	lumassfw
472	mfrvt	lumassfw
473	mfrvz	lumassfw
474	ml	lumassfw
475	mv	lumassfw
476	nf1sm	luacc
477	nf2sm	luacc
478	nf3sm	luacc
479	nfclsm	luacc
480	nfcvsm	luacc
481	nfl4sm	luacc
482	nflsm	luacc
483	nfv4sm	luacc
484	nfvsm	luacc
485	omega	luspeed
486	omegan	luspeed
487	omegd	luspeed
488	omegop	luspeed
489	omgoff	luspeed
490	omgscl	lunounit
491	omsasm	luangle
492	omsm	luangle
493	omtest	luspeed
494	p	lupressa
495	pa	lupressa

496	pain	lupressa
497	pan	lupressa
498	paoff	lupressa
499	paq	lupressa
500	pascl	lunounit
501	patb	lupressa
502	patbabs	lupressa
503	pdc	lupressa
504	pdrat	lunounit
505	pflow	lumassfw
506	pgapt	lupressa
507	pgreac	lunounit
508	phist	lupower
509	pin	lupressa
510	pinteg	luenergy
511	pldr	lulength
512	plen	lulength
513	plp	lupressa
514	plvol	luvolume
515	pmass	lumass
516	pmprf	lunounit
517	pmprfabs	lunounit
518	pmptb	luspeed
519	pmptbabs	luspeed
520	pmvl	lumassfw
521	pmvv	lumassfw
522	pn	lupressa
523	poff	lupressa
524	poffs	lupressa
525	popoff	lupower
526	popscl	lunounit
527	powd	lupower
528	power	lupower
529	powerc	lupower
530	powexp	lunounit
531	powin	lupower

532	powli	lupower
533	powlo	lupower
534	powoff	lupower
535	powop	lupower
536	powou	lupower
537	powr1	lupower
538	powr2	lupower
539	powrf	lunounit
540	powrf1	lunounit
541	powrf2	lunounit
542	powscl	lunounit
543	powstg	lupower
544	powtb	lupower
545	powtb1	lupower
546	powtb2	lupower
547	powtbabs	lupower
548	powvi	lupower
549	powvo	lupower
550	pp	lupressa
551	ppa	lupressa
552	ppower	lupower
553	pq	lupressa
554	pres1	lupressa
555	pres2	lupressa
556	pscl	lunounit
557	pset	lupressa
558	pslen	lulength
559	ptb	lupressa
560	ptbabs	lupressa
561	ptl	lutemp
562	ptv	lutemp
563	pup	lupressa
564	pwin1	lupower
565	pwin2	lupower
566	pwoff1	lupower
567	pwoff2	lupower

568	pwscl1	lunounit
569	pwscl2	lunounit
570	pwtb1abs	lupower
571	pwtb2abs	lupower
572	q235	luenfiss
573	q238	luenfiss
574	q239	luenfiss
575	qavg	luenfiss
576	qchf	luheatfx
577	qheat	lupower
578	qhstot	lupower
579	qhstr	lupower
580	qin	lupower
581	qout	luvolume
582	qp3in	lupower
583	qp3off	lupower
584	qp3rf	lunounit
585	qp3rf1	lunounit
586	qp3rf2	lunounit
587	qp3rfabs	lunounit
588	qp3scl	lunounit
589	qp3tb	lupower
590	qp3tb1	lupower
591	qp3tb2	lupower
592	qp3tbabs	lupower
593	qpin1	lupower
594	qpin2	lupower
595	qpoff1	lupower
596	qpoff2	lupower
597	qppg	luvolhts
598	qppl	luheatfx
599	qppp	lunounit
600	qpppq	lunounit
601	qppps	luvolhts
602	qppv	luheatfx
603	qpscl1	lunounit

604	qpscl2	lunounit
605	qptb1abs	lupower
606	qptb2abs	lupower
607	qualty	lunounit
608	r	lulength
609	r239pf	lunounit
610	radg	lulength
611	radig	lulength
612	radin	lulength
613	radin1	lulength
614	radin2	lulength
615	radrd	lulength
616	radt	lulength
617	rans	lunounit
618	rbmx	luprsrat
619	rcal	lunounit
620	rcbm	luserdef
621	rctc	luserdef
622	rctf	luserdef
623	rdiam	lulength
624	rdpwr	lunounit
625	rdpwrabs	lunounit
626	rdx	lunounit
627	react	lunounit
628	reactn	lunounit
629	regnm	lunounit
630	rflow	luvolflw
631	rfmxm	lumfwrat
632	rfmxv	luacc
633	rftb	lunounit
634	rftn	lutemp
635	rhead	lupumphd
636	rhol	luden
637	rhom	luden
638	rhop	luden
639	rhov	luden

640	rmass	lumass
641	rmatsm	lunounit
642	rmckn	lunounit
643	rmvm	lumassfw
644	roan	luden
645	roln	luden
646	romega	luspeed
647	romgmx	luradacc
648	rovn	luden
649	rpkf	lunounit
650	rpopmx	lupowrat
651	rpower	lupower
652	rpowmx	lupowrat
653	rpowri	lupower
654	rpowrn	lupower
655	rpwmx1	lupowrat
656	rpwmx2	lupowrat
657	rpwoffp	lupower
658	rpwoffr	lunounit
659	rpwrf	lunounit
660	rpwrt	lulength
661	rpwscl	lunounit
662	rpwtbabp	lupower
663	rpwtbabr	lunounit
664	rpwtbp	lupower
665	rpwtbr	lunounit
666	rqp3mx	lupowrat
667	rqpmx1	lupowrat
668	rqpmx2	lupowrat
669	rrho	luden
670	rrpwmxp	lupowrat
671	rrpwmxr	lurtime
672	rrs	lulength
673	rs	lunounit
674	rsabs	lunounit
675	rsm	lulength

676	rtork	lutorque
677	rtwfp	lunounit
678	rvmf	lumassfw
679	rvmx	lurtime
680	rvov	lurtime
681	rws	lulength
682	rzht	lulength
683	rzpwmx	lurtime
684	s	luden
685	sa	luarea
686	saf	lunounit
687	scn1	luserdef
688	scn2	luserdef
689	scn3	luserdef
690	scn4	luserdef
691	scn5	luserdef
692	sedint	lutime
693	setp	luserdef
694	setpnt	luserdef
695	shelv	lulength
696	shtd	lunounit
697	smom	lupumphd
698	sn	luden
699	solid	luden
700	stnui	lunounit
701	stnuo	lunounit
702	strtmp	lutemp
703	stype	lunounit
704	suprht	lutemp
705	sv	luserdef
706	t	luangle
707	t0sm	luangle
708	tai	luarea
709	tan	luarea
710	tcefn	luenergy
711	tcen	luenergy

712	tchf	lutemp
713	tcilmf	lumass
714	tcivmf	lumass
715	tcolmf	lumass
716	tcore	lutemp
717	tcovmf	lumass
718	tcreac	lunounit
719	tdc	lutemp
720	tdm	lunounit
721	tdragxr	lunounit
722	tdragyt	lunounit
723	tdragz	lunounit
724	tend	lutime
725	tener	luenergy
726	tfmass	lumass
727	tfr0	lutorque
728	tfr1	lutorque
729	tfr2	lutorque
730	tfr3	lutorque
731	tfrb	luspeed
732	tfreac	lunounit
733	tfrl0	lutorque
734	tfrl1	lutorque
735	tfrl2	lutorque
736	tfrl3	lutorque
737	th	lulength
738	th1	lulength
739	th2	lulength
740	thg	lulength
741	tilg	lutemp
742	timdl	lutime
743	timdu	lutime
744	timet	lutime
745	tin	lutemp
746	tivg	lutemp
747	tk	luthcond

748	tl	lutemp
749	tldi	lutemp
750	tldo	lutemp
751	tlen	lulength
752	tli	lutemp
753	tlin	lutemp
754	tln	lutemp
755	tlo	lutemp
756	tloff	lutemp
757	tlp	lutemp
758	tlpliq	luvolume
759	tlq	lutemp
760	tlscl	lunounit
761	tltb	lutemp
762	tltbabs	lutemp
763	tneut	lutime
764	tnstep	lunouhit
765	tolg	lutemp
766	torque	lutorque
767	toutl	lutemp
768	toutl1	lutemp
769	toutl2	lutemp
770	toutv	lutemp
771	toutv1	lutemp
772	toutv2	lutemp
773	tovg	lutemp
774	tp	lulength
775	tpow	lupower
776	tpowr	lutime
777	tramax	lutemp
778	trbrf	lunounit
779	trbsig	lunounit
780	trbtb	lupower
781	trbtbabs	lupower
782	trh	lulength
783	trhmax	lutemp

784	trpsig	luserdef
785	trrl	lulength
786	ts	luserdef
787	tsat	lutemp
788	tscore	lutemp
789	tsdc	lutemp
790	tsdlt	lutime
791	tsdut	lutime
792	tslp	lutemp
793	tsp1	lunounit
794	tsp2	lunounit
795	tsp3	lunounit
796	tsp4	lunounit
797	tssn	lutemp
798	tsup	lutemp
799	ttheta	luangle
800	ttp1	lunounit
801	ttp2	lunounit
802	ttp3	lunounit
803	ttp4	lunounit
804	tup	lutemp
805	tv	lutemp
806	tvi	lutemp
807	tvin	lutemp
808	tvn	lutemp
809	tvo	lutemp
810	tvoff	lutemp
811	tvol	luvolume
812	tvq	lutemp
813	tvscl	lunounit
814	tvtb	lutemp
815	tvtbabs	lutemp
816	tw	lutemp
817	twaen	luenergy
818	twan	luenergy
819	tween	luenergy

820	twen	luenergy
821	twgi	lutemp
822	twgn	lutemp
823	twq	lutemp
824	twtold	lunounit
825	tx0vsm	luangle
826	ty0vsm	luangle
827	tz0vsm	luangle
828	uvsm	lunounit
829	vbmass	lumass
830	vcore	lumass
831	vdclq	lumass
832	vflow	luvolflw
833	vflowp	luvolflw
834	vl	luvel
835	vlin	luvel
836	vln	luvel
837	vlntxr	luvel
838	vlntyt	luvel
839	vlntz	luvel
840	vlnxr	luvel
841	vlnyt	luvel
842	vlnz	luvel
843	vloff	luvel
844	vloss	luvolume
845	vlpliq	lunounit
846	vlplm	lumass
847	vlq	luvel
848	vltn	luvel
849	vmass	lumass
850	vmfr	lumassfw
851	vmscl	lunounit
852	vmtbabsm	lumassfw
853	vmtbabsv	luvel
854	vmtbm	lumassfw
855	vmtbv	luvel

856	vol	luvolume
857	volin	luvolume
858	vrf	lunounit
859	vsflow	lumassfw
860	vtb1	lunounit
861	vtb2	lunounit
862	vupliq	lunounit
863	vuplm	lumass
864	vv	luvel
865	vvin	luvel
866	vvn	luvel
867	vvntxr	luvel
868	vvntyt	luvel
869	vvntz	luvel
870	vvnxr	luvel
871	vvnyt	luvel
872	vvnz	luvel
873	vvoff	luvel
874	vvq	luvel
875	vvscl	lunounit
876	vvtab	lunounit
877	vvtb	luvel
878	vvtbabs	luvel
879	vvtn	luvel
880	vwfmlx	lunounit
881	vwfmly	lunounit
882	vwfmlz	lunounit
883	vwfmvx	lunounit
884	vwfmvy	lunounit
885	vwfmvz	lunounit
886	waig	luarea
887	waog	luarea
888	wap	luarea
889	was	luarea
890	wdsasm	luangle
891	wdsm	luangle

892	wflxr	lunounit
893	wflyt	lunounit
894	wflz	lunounit
895	wfmfl	lunounit
896	wfmfv	lunounit
897	wfvxr	lunounit
898	wfvyt	lunounit
899	wfvz	lunounit
900	width	lulength
901	wsasm	luangle
902	wsm	luangle
903	x	lulength
904	x0sm	lulength
905	x0vsm	lulength
906	xco	lunounit
907	xcu	lunounit
908	xpos	lunounit
909	xsm	lulength
910	xvset	lunounit
911	y	lulength
912	y0sm	lulength
913	y0vsm	lulength
914	ysm	lulength
915	z	lulength
916	z0sm	lulength
917	z0vsm	lulength
918	zht	lulength
919	zhtr	lulength
920	zlpbot	lulength
921	zlptop	lulength
922	zpwin	luserdef
923	zpwoff	luserdef
924	zpwrf	lunounit
925	zpwtb	lunounit
926	zpwtbabs	lunounit
927	zpwzt	lulength

```
928 zs       lunounit
929 zsabs    lunounit
930 zsgrid   lulength
931 zsm      lulength
932 zupbot   lulength
933 zuptop   lulength
934 zzzzzzzz lunitnam
935 zlastone lunounit
```

F.6. LISTING OF FILE LABELV.H

```
         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890

      CHARACTER*2 lud,lutp,lus
      CHARACTER*3 lut,luz
      CHARACTER*4 luar,lue,lum,luvo
      CHARACTER*5 lup,lupd,luv,luvf
      CHARACTER*6 luis
      CHARACTER*7 luen,lumf,lupt,lupw
      CHARACTER*8 labels,labun,luid,lur,lusp,lusz,lutm
      CHARACTER*9 luha
      CHARACTER*10 lucp
      CHARACTER*11 luph
      CHARACTER*12 luhx,lutc,runcb
      CHARACTER*13 luh,luncb,lupcb
      CHARACTER*14 labsv
      CHARACTER*19 ludh
      CHARACTER*26 alpbet
      COMMON /labelv1/labun(150)
      COMMON /labelv2/luncb(2,150)
      COMMON /labelv3/lupcb(2,150)
      COMMON /labelv4/runcb(2,150)
      COMMON /labelv5/labsv(2,105)
      COMMON /labelv/factor(150),offset(150)
      COMMON /labelv/ih(26),itls(777),itsv(105),ils,ilu,ilun,iold
```

```
      COMMON /labelv/labels(777)
      COMMON /labelv/alpbet,luar,lucp,lud,ludh,lue,luen,luh,luha,luhx,   &
     & luid,luis,lum,lumf,lup,lupd,luph,lupt,lupw,lur,lus,lusp,lusz,     &
     & lut,lutc,lutm,lutp,luv,luvf,luvo,luz
```

F.7. LISTING OF FILE BLKDAT2.F

```
         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890

      BLOCK DATA blkdat2
      IMPLICIT REAL*8 (a-h,o-z)
!
!     subroutine block data blkdat2 initializes the
!     si/english units parameters in include labelv.h
!
      INCLUDE 'labelv.h'
!
      DATA (factor(i),i=  1,151)/                                        &
     &  1.00000000D+00, 1.00000000D+00, 1.80000000D+00, 1.80000000D+00,  &
     &  3.28083990D+00, 1.07639104D+01, 3.53146667D+01, 3.28083990D+00,  &
     &  3.28083990D+00, 3.34552563D-01, 1.58503222D+04, 1.60184634D+01,  &
     &  2.20462262D+00, 7.93664144D+03, 2.20462262D+00, 7.37338117D+02,  &
     &  2.24740658D+02, 6.24279606D-02, 3.46822003D-02, 1.90280424D-02,  &
     &  1.45037738D-04, 1.45037738D-04, 1.45037738D-04, 2.37303604D+01,  &
     &  7.37562149D-01, 7.72373277D-02, 8.08827404D-03, 3.41214163D+00,  &
     &  3.41214163D+00, 1.04002077D+00, 3.16998331D-01, 9.66210912D-02,  &
     &  5.77789317D-01, 1.76110184D-01, 1.89563424D+00, 9.47817120D-04,  &
     &  4.29922614D-04, 2.38845897D-04, 1.00000000D+00, 5.55555556D-01,  &
     &  4.53592370D-01, 6.89475729D+03, 9.54929659D+00, 9.54929659D+00,  &
     &  5.72957795D+01, 1.00000000D+00, 1.00000000D+00, 1.00000000D+00,  &
     &  3.08641975D-01, 1.00000000D+00, 4.30425636D+02, 100*1.0000D+00/
      DATA (offset(i),i=  1,151)/                                        &
     &  0.00000000D+00, 0.00000000D+00,-4.59670000D+02, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
```

```
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 0.00000000D+00,  &
     &  0.00000000D+00, 0.00000000D+00, 0.00000000D+00, 100*0.0000D+00/
      DATA ih,ils,ilu,ilun,iold/26*0,781, 52, 52,1/
      DATA (itls(i),i=  1,300)/                                            &
     &  6,24, 1, 1, 1, 1, 1, 1, 1, 1,35,35,35,13,13,52,52,45,30, 6,        &
     & 18,18,25, 5, 6, 6, 5,18,18,24, 5, 1,18,19,13,13,26,46,14,14,        &
     &  5, 5, 5,52,52,52, 2, 1,52,52, 2, 1,52,52, 1, 1,20,28,28,24,        &
     & 14,36,28,13, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,        &
     &  1, 1, 1, 1, 1, 1,35,35,35,20,20,20,20,20,14,14, 1,33, 5,13,        &
     & 14, 1, 1, 3, 3, 1, 1, 1,14,14, 1, 1, 1, 1,33, 1, 1, 1, 1, 1,        &
     &  1,38,28, 1, 1, 2, 2, 1,21,27, 1, 1,14, 1, 5, 2,22, 4, 4, 5,        &
     &  5, 5,13, 2,51,51,22,22,22, 1, 5,45, 2, 4, 2, 2, 4, 2, 4, 1,        &
     &  2, 2, 2, 4, 4, 5, 5, 5, 5, 5, 1, 2, 1, 1,24,24, 1, 1, 5, 1,        &
     & 40,49, 1,40,49,36,36,36,37, 1, 1, 5, 1,28, 6, 1, 6, 6, 6, 1,        &
     &  5, 1, 1,14, 6,14,14,14,14, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,        &
     &  1, 1, 1, 1,13, 1, 1,52, 1, 1, 1,14, 5, 5, 5,17, 1, 9, 2, 1,        &
     & 48, 1, 9, 9, 9, 1, 1, 1, 1, 5, 5, 5, 5, 1, 5, 5, 5, 5, 5,10,        &
     &  5,31,34,34,34,34,34,34,34,34,34,34,34,34,34,34,34,34,34, 1,        &
     &  1, 5, 1, 1, 1, 1, 3,34,34,28,28,28,28, 1, 1, 1, 1,33, 1, 1/
      DATA (itls(i),i=301,600)/                                            &
     &  1, 1,34,34, 5,34,42,13,24,39,39,14,14,14,14,14,14,14,14,14,        &
     & 14,14, 9, 9, 9, 9, 9, 9, 9, 9, 9,43,43,43,43,43, 1,45,45,43,        &
     & 21,21,21,21,21,21, 1,21,21,21, 1,14,21, 1,28,21,36, 5, 5,21,        &
     &  7,13, 1, 1,43,43,14,14,21,21,21,28, 1,28,28,28, 1,28,28,28,        &
     & 28,28,28,28,28, 1, 1, 1, 1,28,28,28,28,28,28,28,21,21,28,21,        &
     & 21,21, 1,21, 5,21,21, 3, 3,21,28,28,28,28, 1, 1,28,28,47,47,        &
     & 47,47,31,28,28,28,28, 7,28,28, 1, 1, 1, 1, 1,28,28,28,28,28,        &
     & 28,28,28,32,31, 1, 1,32,31, 1, 1,28,28, 1, 5, 1, 5, 5, 5, 5,        &
```

```
     &  5, 5, 5, 1,23, 1,52,52,52, 5, 1, 1, 1, 1, 1, 1,11,15, 9, 1,     &
     &  3,10,18,18,18,18,13, 1, 1,14,18,18,43,44,18, 1,29,28,29,28,     &
     & 28,29,29,28, 1, 1, 5, 1,28, 1,28, 1,29,29,29,18,29,39, 5, 1,     &
     &  1, 5,25, 1,14,39,39, 5, 5,39,18, 6, 1,52,52,52,52,52, 2,52,     &
     & 52, 5, 1,10,18,18, 1, 1, 3, 1, 3,52,45,45, 6, 6,36,36, 3,13,     &
     & 13,13, 3,13, 1, 3, 1, 1, 1, 1, 2,36,13,25,25,25,25,43, 1,25,     &
     & 25,25,25, 5, 5, 5, 5, 3, 2, 2, 2, 3, 3,33, 3, 3, 3, 5, 3, 3/
      DATA (itls(i),i=601,781)/                                         &
     &  3, 3, 3, 3, 7, 3, 1, 3, 3, 2, 1, 3,25, 3, 3, 3, 3, 3, 3, 3,     &
     &  5,28, 2, 3, 1, 1,28,28, 5, 3,52, 5,52, 3, 3, 3, 2, 2, 3, 1,     &
     &  1, 1, 1, 3, 3,45, 1, 1, 1, 1, 3, 3, 3, 3, 3, 3, 3, 7, 3, 1,     &
     &  3, 3, 3,36,36,36,36, 3, 3, 3, 1,45,45,45, 1,13,13,13,11,11,     &
     &  8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 7, 1,13, 8, 8,13,14, 1,14, 8,     &
     & 14, 8, 7, 7, 1,14, 1, 1, 1,13, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8,     &
     &  8, 1, 1, 8, 8, 8, 1, 1, 1, 1, 1, 1, 6, 6, 6, 6,45,45, 1, 1,     &
     &  1, 1, 1, 1, 1, 1, 5,45,45, 5, 5, 5, 1, 1, 1, 5, 1, 5, 5, 5,     &
     &  5, 5, 5, 5, 5, 5, 5, 5,52,52, 1, 1, 1, 5, 1, 1, 5, 5, 5, 5,     &
     & 50/
      DATA (itsv(i),i=  1,106)/                                         &
     &  2,28,21,21, 3, 5, 5, 3, 3,14,14,21,14,14,14, 5, 1, 1,28, 2,     &
     &  5,21, 3, 3, 3, 3, 3, 1,14,14,14,14,14,14, 8, 8, 8, 8, 8, 8,     &
     &  1,43, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 1, 1,52, 1,28,28, 3,     &
     &  3,10,25,10, 5, 5, 5, 5,14,14,14, 8, 8, 8,18,18,18,18,13,21,     &
     & 37,37,37, 3, 3,38,38,37,28,28,28,34,34,34,34,35,35,20,20,20,     &
     & 18,17,28,28, 3,51/
      DATA (labels(i),i=  1, 75)/                                       &
     & 'ads     ','aeffmi  ','alp     ','alpha   ','alphap  ',          &
     & 'alpin   ','alpn    ','alpoff  ','alptb   ','alreac  ',          &
     & 'alv     ','alve    ','alven   ','am      ','amh2    ',          &
     & 'amncss  ','amxcss  ','angl    ','apowr   ','area    ',          &
     & 'arln    ','arvn    ','atork   ','atw     ','avent   ',          &
     & 'avlve   ','aw      ','bcr0    ','bcr1    ','beffmi  ',          &
     & 'belv    ','beta    ','bpp0    ','bpp1    ','bsa     ',          &
     & 'bsmass  ','btork   ','burn    ','bxa     ','bxmass  ',          &
     & 'bxsm    ','bysm    ','bzsm    ','cb      ','cbcon1  ',          &
     & 'cbcon2  ','cbdt    ','cbeta   ','cbftab  ','cbgain  ',          &
```

```
     & 'cbtau   ','cbwt    ','cbxmax  ','cbxmin  ','ccflc   ',          &
     & 'ccflm   ','ccif    ','cdgn    ','cdhn    ','ceffmi  ',          &
     & 'cemfr   ','cener   ','cepwn   ','cfmass  ','cfrlxr  ',          &
     & 'cfrlyt  ','cfrlz   ','cfrvxr  ','cfrvyt  ','cfrvz   ',          &
     & 'cfz     ','cfz3    ','cfzlxr  ','cfzlyt  ','cfzlz   '/
       DATA (labels(i),i= 76,150)/                                      &
     & 'cfzvxr  ','cfzvyt  ','cfzvz   ','chm12   ','chm13   ',          &
     & 'chm14   ','chm15   ','chm22   ','chm23   ','chm24   ',          &
     & 'chm25   ','chti    ','chtia   ','chtin   ','cif     ',          &
     & 'cifn    ','cifxr   ','cifyt   ','cifz    ','cimfrl  ',          &
     & 'cimfrv  ','cks     ','ckw     ','clenn   ','cmass   ',          &
     & 'cmflow  ','cnmax   ','cnmin   ','cntlmn  ','cntlmx  ',          &
     & 'coef1   ','coef2   ','cof3sq  ','comfrl  ','comfrv  ',          &
     & 'conc    ','concin  ','conctb  ','conctbab','cond    ',          &
     & 'conoff  ','conscl  ','cosangl ','cosp    ','coss    ',          &
     & 'cost    ','cp      ','cpow    ','cpowr   ','cpowrabs',          &
     & 'cputot  ','cpvint  ','crliqfr ','crpress ','ctork   ',          &
     & 'ctrans  ','dbreac  ','dcflow  ','dclqvol ','dds     ',          &
     & 'delt    ','deltap  ','deltl   ','deltv   ','dh      ',          &
     & 'dia     ','diah    ','dmass   ','dmpint  ','dnew    ',          &
     & 'dnewn   ','dpcvn   ','dpmax   ','dpovn   ','dprmax  '/
       DATA (labels(i),i=151,225)/                                      &
     & 'dr      ','dt      ','dtend   ','dtlmax  ','dtmax   ',          &
     & 'dtmin   ','dtrmax  ','dtsm    ','dtsmax  ','dtsofs  ',          &
     & 'dtsoft  ','dtsp    ','dtstrt  ','dtvmax  ','dtxht   ',          &
     & 'dx      ','dxin    ','dy      ','dz      ','dznht   ',          &
     & 'edh     ','edint   ','effdsn  ','effld   ','effmi   ',          &
     & 'effmil  ','effstg  ','efgen   ','elev    ','emcif1  ',          &
     & 'emcif2  ','emcif3  ','emcof1  ','emcof2  ','emcof3  ',          &
     & 'enin1   ','enin2   ','eninp   ','enth    ','epso    ',          &
     & 'epss    ','epsw    ','errsm   ','extsou  ','fa      ',          &
     & 'favlve  ','faxr    ','fayt    ','faz     ','fdfhl   ',          &
     & 'felv    ','ff      ','fisphi  ','flow    ','flowarea',          &
     & 'flowin  ','flwin   ','flwoff  ','flwou   ','fmaxov  ',          &
     & 'fminov  ','fp235   ','fp238   ','fp239   ','fpuo2   ',          &
     & 'frcvn   ','frfaxr  ','frfayt  ','frfaz   ','fric    ',          &
```

```
     & 'fricr   ','frovn   ','frvol   ','fsi     ','fsmass  '/
      DATA (labels(i),i=226,300)/                                   &
     & 'fso     ','ftd     ','ftx     ','fty     ','fucrac  ',       &
     & 'funh    ','fxmass  ','fxsm    ','fysm    ','fzsm    ',       &
     & 'gam     ','gamma   ','gc      ','gfint   ','gmix    ',       &
     & 'gmles   ','grav    ','gravxr  ','gravyt  ','gravz   ',       &
     & 'gvf     ','gxrc    ','gytc    ','gzc     ','hbs     ',       &
     & 'hd      ','hd-ht   ','hd3     ','hdm     ','hdri    ',       &
     & 'hdro    ','hdxr    ','hdyt    ','hdz     ','head    ',       &
     & 'height  ','hgam    ','hgap    ','hgapo   ','hil     ',       &
     & 'hilg    ','hiv     ','hivg    ','hl      ','hli     ',       &
     & 'hlo     ','holg    ','houtl   ','houtl1  ','houtl2  ',       &
     & 'houtv   ','houtv1  ','houtv2  ','hovg    ','hs      ',       &
     & 'hsabs   ','hsk     ','hsp1    ','hsp2    ','hsp3    ',       &
     & 'hsp4    ','hstn    ','htcwl   ','htcwv   ','htlsci  ',       &
     & 'htlsco  ','htlsgi  ','htlsgo  ','htmli   ','htmlo   ',       &
     & 'htmvi   ','htmvo   ','htnew   ','htp1    ','htp2    '/
      DATA (labels(i),i=301,375)/                                   &
     & 'htp3    ','htp4    ','hv      ','hvi     ','hvlve   ',       &
     & 'hvo     ','hynew   ','imflow  ','inrta   ','lamda   ',       &
     & 'lamdh   ','mflow   ','mfrl    ','mfrlr   ','mfrlt   ',       &
     & 'mfrlz   ','mfrv    ','mfrvr   ','mfrvt   ','mfrvz   ',       &
     & 'ml      ','mv      ','nf1sm   ','nf2sm   ','nf3sm   ',       &
     & 'nfclsm  ','nfcvsm  ','nfl4sm  ','nflsm   ','nfv4sm  ',       &
     & 'nfvsm   ','omega   ','omegan  ','omegd   ','omegop  ',       &
     & 'omgoff  ','omgscl  ','omsasm  ','omsm    ','omtest  ',       &
     & 'p       ','pa      ','pain    ','pan     ','paoff   ',       &
     & 'paq     ','pascl   ','patb    ','patbabs ','pdc     ',       &
     & 'pdrat   ','pflow   ','pgapt   ','pgreac  ','phist   ',       &
     & 'pin     ','pinteg  ','pldr    ','plen    ','plp     ',       &
     & 'plvol   ','pmass   ','pmprf   ','pmprfabs','pmptb   ',       &
     & 'pmptbabs','pmvl    ','pmvv    ','pn      ','poff    ',       &
     & 'poffs   ','popoff  ','popscl  ','powd    ','power   '/
      DATA (labels(i),i=376,450)/                                   &
     & 'powerc  ','powexp  ','powin   ','powli   ','powlo   ',       &
     & 'powoff  ','powop   ','powou   ','powr1   ','powr2   ',       &
```

```
     & 'powrf   ','powrf1  ','powrf2  ','powscl  ','powstg  ',      &
     & 'powtb   ','powtb1  ','powtb2  ','powtbabs','powvi   ',      &
     & 'powvo   ','pp      ','ppa     ','ppower  ','pq      ',      &
     & 'pres1   ','pres2   ','pscl    ','pset    ','pslen   ',      &
     & 'ptb     ','ptbabs  ','ptl     ','ptv     ','pup     ',      &
     & 'pwin1   ','pwin2   ','pwoff1  ','pwoff2  ','pwscl1  ',      &
     & 'pwscl2  ','pwtb1abs','pwtb2abs','q235    ','q238    ',      &
     & 'q239    ','qavg    ','qchf    ','qheat   ','qhstot  ',      &
     & 'qhstr   ','qin     ','qout    ','qp3in   ','qp3off  ',      &
     & 'qp3rf   ','qp3rf1  ','qp3rf2  ','qp3rfabs','qp3scl  ',      &
     & 'qp3tb   ','qp3tb1  ','qp3tb2  ','qp3tbabs','qpin1   ',      &
     & 'qpin2   ','qpoff1  ','qpoff2  ','qppg    ','qppl    ',      &
     & 'qppp    ','qpppq   ','qppps   ','qppv    ','qpscl1  '/
      DATA (labels(i),i=451,525)/                                   &
     & 'qpscl2  ','qptb1abs','qptb2abs','qualty  ','r       ',      &
     & 'r239pf  ','radg    ','radig   ','radin   ','radin1  ',      &
     & 'radin2  ','radrd   ','radt    ','rans    ','rbmx    ',      &
     & 'rcal    ','rcbm    ','rctc    ','rctf    ','rdiam   ',      &
     & 'rdpwr   ','rdpwrabs','rdx     ','react   ','reactn  ',      &
     & 'regnm   ','rflow   ','rfmxm   ','rfmxv   ','rftb    ',      &
     & 'rftn    ','rhead   ','rhol    ','rhom    ','rhop    ',      &
     & 'rhov    ','rmass   ','rmatsm  ','rmckn   ','rmvm    ',      &
     & 'roan    ','roln    ','romega  ','romgmx  ','rovn    ',      &
     & 'rpkf    ','rpopmx  ','rpower  ','rpowmx  ','rpowri  ',      &
     & 'rpowrn  ','rpwmx1  ','rpwmx2  ','rpwoffp ','rpwoffr ',      &
     & 'rpwrf   ','rpwrt   ','rpwscl  ','rpwtbabp','rpwtbabr',      &
     & 'rpwtbp  ','rpwtbr  ','rqp3mx  ','rqpmx1  ','rqpmx2  ',      &
     & 'rrho    ','rrpwmxp ','rrpwmxr ','rrs     ','rs      ',      &
     & 'rsabs   ','rsm     ','rtork   ','rtwfp   ','rvmf    '/
      DATA (labels(i),i=526,600)/                                   &
     & 'rvmx    ','rvov    ','rws     ','rzht    ','rzpwmx  ',      &
     & 's       ','sa      ','saf     ','scn1    ','scn2    ',      &
     & 'scn3    ','scn4    ','scn5    ','sedint  ','setp    ',      &
     & 'setpnt  ','shelv   ','shtd    ','smom    ','sn      ',      &
     & 'solid   ','stnui   ','stnuo   ','strtmp  ','stype   ',      &
     & 'suprht  ','sv      ','t       ','t0sm    ','tai     ',      &
```

```
     & 'tan     ','tcefn   ','tcen    ','tchf    ','tcilmf  ',            &
     & 'tcivmf  ','tcolmf  ','tcore   ','tcovmf  ','tcreac  ',            &
     & 'tdc     ','tdm     ','tdragxr ','tdragyt ','tdragz  ',            &
     & 'tend    ','tener   ','tfmass  ','tfr0    ','tfr1    ',            &
     & 'tfr2    ','tfr3    ','tfrb    ','tfreac  ','tfrl0   ',            &
     & 'tfrl1   ','tfrl2   ','tfrl3   ','th      ','th1     ',            &
     & 'th2     ','thg     ','tilg    ','timdl   ','timdu   ',            &
     & 'timet   ','tin     ','tivg    ','tk      ','tl      ',            &
     & 'tldi    ','tldo    ','tlen    ','tli     ','tlin    '/
      DATA (labels(i),i=601,675)/                                         &
     & 'tln     ','tlo     ','tloff   ','tlp     ','tlpliq  ',            &
     & 'tlq     ','tlscl   ','tltb    ','tltbabs ','tneut   ',            &
     & 'tnstep  ','tolg    ','torque  ','toutl   ','toutl1  ',            &
     & 'toutl2  ','toutv   ','toutv1  ','toutv2  ','tovg    ',            &
     & 'tp      ','tpow    ','tpowr   ','tramax  ','trbrf   ',            &
     & 'trbsig  ','trbtb   ','trbtbabs','trh     ','trhmax  ',            &
     & 'trpsig  ','trrl    ','ts      ','tsat    ','tscore  ',            &
     & 'tsdc    ','tsdlt   ','tsdut   ','tslp    ','tsp1    ',            &
     & 'tsp2    ','tsp3    ','tsp4    ','tssn    ','tsup    ',            &
     & 'ttheta  ','ttp1    ','ttp2    ','ttp3    ','ttp4    ',            &
     & 'tup     ','tv      ','tvi     ','tvin    ','tvn     ',            &
     & 'tvo     ','tvoff   ','tvol    ','tvq     ','tvscl   ',            &
     & 'tvtb    ','tvtbabs ','tw      ','twaen   ','twan    ',            &
     & 'tween   ','twen    ','twgi    ','twgn    ','twq     ',            &
     & 'twtold  ','tx0vsm  ','ty0vsm  ','tz0vsm  ','uvsm    '/
      DATA (labels(i),i=676,750)/                                         &
     & 'vbmass  ','vcore   ','vdclq   ','vflow   ','vflowp  ',            &
     & 'vl      ','vlin    ','vln     ','vlntxr  ','vlntyt  ',            &
     & 'vlntz   ','vlnxr   ','vlnyt   ','vlnz    ','vloff   ',            &
     & 'vloss   ','vlpliq  ','vlplm   ','vlq     ','vltn    ',            &
     & 'vmass   ','vmfr    ','vmscl   ','vmtbabsm','vmtbabsv',            &
     & 'vmtbm   ','vmtbv   ','vol     ','volin   ','vrf     ',            &
     & 'vsflow  ','vtb1    ','vtb2    ','vupliq  ','vuplm   ',            &
     & 'vv      ','vvin    ','vvn     ','vvntxr  ','vvntyt  ',            &
     & 'vvntz   ','vvnxr   ','vvnyt   ','vvnz    ','vvoff   ',            &
     & 'vvq     ','vvscl   ','vvtab   ','vvtb    ','vvtbabs ',            &
```

```
     & 'vvtn    ','vwfmlx  ','vwfmly  ','vwfmlz  ','vwfmvx  ',        &
     & 'vwfmvy  ','vwfmvz  ','waig    ','waog    ','wap     ',        &
     & 'was     ','wdsasm  ','wdsm    ','wflxr   ','wflyt   ',        &
     & 'wflz    ','wfmfl   ','wfmfv   ','wfvxr   ','wfvyt   ',        &
     & 'wfvz    ','width   ','wsasm   ','wsm     ','x       '/
      DATA (labels(i),i=751,781)/                                     &
     & 'x0sm    ','x0vsm   ','xco     ','xcu     ','xpos    ',        &
     & 'xsm     ','xvset   ','y       ','y0sm    ','y0vsm   ',        &
     & 'ysm     ','z       ','z0sm    ','z0vsm   ','zht     ',        &
     & 'zhtr    ','zlpbot  ','zlptop  ','zpwin   ','zpwoff  ',        &
     & 'zpwrf   ','zpwtb   ','zpwtbabs','zpwzt   ','zs      ',        &
     & 'zsabs   ','zsgrid  ','zsm     ','zupbot  ','zuptop  ',        &
     & 'zzzzzzzz'/
      DATA ((labsv(i,j),i=1,2),j=  1, 15)/                            &
     & 'time (s)       ','time (s)       ',                           &
     & 'core power (w)','power (btu/hr)',                             &
     & 'pri press (pa)','p press (psia)',                             &
     & 'pzr press (pa)','pz pres (psia)',                             &
     & 'prizr temp (k)','prizr temp (f)',                             &
     & 'pz liq lev (m)','pz lq lev (ft)',                             &
     & 'tk liq lev (m)','tk lq lev (ft)',                             &
     & 'hot-l temp (k)','hot-l temp (f)',                             &
     & 'cld-l temp (k)','cld-l temp (f)',                             &
     & 'p mflow (kg/s)','mflow (lbm/hr)',                             &
     & 'ecc mfw (kg/s)','eccmf (lbm/hr)',                             &
     & 'sec press (pa)','sc pres (psia)',                             &
     & 'st mflw (kg/s)','st mf (lbm/hr)',                             &
     & 'fw mflw (kg/s)','fw mf (lbm/hr)',                             &
     & 'afw mfw (kg/s)','af mf (lbm/hr)'/
      DATA ((labsv(i,j),i=1,2),j= 16, 30)/                            &
     & 'sc liq lev (m)','sc lq lev (ft)',                             &
     & 'user defined  ','user defined  ',                             &
     & 'user defined  ','user defined  ',                             &
     & 'core power (w)','power (btu/hr)',                             &
     & 'pwr period (s)','pwr period (s)',                             &
     & 'liq level (m) ','liq level (ft)',                             &
```

```
     & 'pressure (pa) ','press (psia)  ',                    &
     & 'gas temp (k)  ','gas temp (f)  ',                    &
     & 'liq temp (k)  ','liq temp (f)  ',                    &
     & 'in sf temp (k)','in sf temp (f)',                    &
     & 'surf temp (k) ','surf temp (f) ',                    &
     & 'htstr temp (k)','htstr temp (f)',                    &
     & 'void fraction ','void fraction ',                    &
     & 'yt v mf (kg/s)','y vmf (lbm/hr)',                    &
     & 'z vp mf (kg/s)','z vmf (lbm/hr)'/
      DATA ((labsv(i,j),i=1,2),j= 31, 45)/                   &
     & 'xr v mf (kg/s)','x vmf (lbm/hr)',                    &
     & 'yt l mf (kg/s)','y lmf (lbm/hr)',                    &
     & 'z lq mf (kg/s)','z lmf (lbm/hr)',                    &
     & 'xr l mf (kg/s)','x lmf (lbm/hr)',                    &
     & 'yt g vel (m/s)','y g vel (ft/s)',                    &
     & 'z gs vel (m/s)','z g vel (ft/s)',                    &
     & 'xr g vel (m/s)','x g vel (ft/s)',                    &
     & 'yt l vel (m/s)','y l vel (ft/s)',                    &
     & 'z lq vel (m/s)','z l vel (ft/s)',                    &
     & 'xr l vel (m/s)','x l vel (ft/s)',                    &
     & 'dis solute/liq','dis solute/liq',                    &
     & 'pm spd (rad/s)','pm speed (rpm)',                    &
     & 'valve farea fr','valve farea fr',                    &
     & 'valve stem pos','valve stem pos',                    &
     & 'mult cnst keff','mult cnst keff'/
      DATA ((labsv(i,j),i=1,2),j= 46, 60)/                   &
     & 'prog reac     ','prog reac     ',                    &
     & 'tot fdbk reac ','tot fdbk reac ',                    &
     & 'fuel temp reac','fuel temp reac',                    &
     & 'cool temp reac','cool temp reac',                    &
     & 'void frac reac','void frac reac',                    &
     & 'solute reac   ','solute reac   ',                    &
     & 'av fl temp (k)','av fl temp (f)',                    &
     & 'av cl temp (k)','av cl temp (f)',                    &
     & 'avg void fr   ','avg void fr   ',                    &
     & 'avg sol (ppm) ','avg sol (ppm) ',                    &
```

```
     & 'trp signal (*)','trp signal (*)',                        &
     & 'trp set status','trp set status',                        &
     & 'prompt pwr (w)','pt pw (btu/hr)',                        &
     & 'decayh pwr (w)','dh pw (btu/hr)',                        &
     & 'a mx sf tp (k)','a mx sf tp (f)'/
      DATA ((labsv(i,j),i=1,2),j= 61, 75)/                       &
     & 's mx sf tp (k)','s mx sf tp (f)',                        &
     & 'pmp hd (m2/s2)','ph(lbf*ft/lbm)',                        &
     & 'torque (pa*m3)','torq (lbf*ft) ',                        &
     & 'p msou (m2/s2)','ms(lbf*ft/lbm)',                        &
     & 'vlve h dia (m)','vlv h dia (ft)',                        &
     & 'yt hyd dia (m)','y hyd dia (ft)',                        &
     & 'z hyd diam (m)','z hyd dia (ft)',                        &
     & 'xr hyd dia (m)','x hyd dia (ft)',                        &
     & 'yt m mf (kg/s)','y mmf (lbm/hr)',                        &
     & 'z m mfw (kg/s)','z mmf (lbm/hr)',                        &
     & 'xr m mf (kg/s)','x mmf (lbm/hr)',                        &
     & 'yt m vel (m/s)','y m vel (ft/s)',                        &
     & 'z mx vel (m/s)','z m vel (ft/s)',                        &
     & 'xr m vel (m/s)','x m vel (ft/s)',                        &
     & 'vp den (kg/m3)','v dn (lbm/ft3)'/
      DATA ((labsv(i,j),i=1,2),j= 76, 90)/                       &
     & 'lq den (kg/m3)','l dn (lbm/ft3)',                        &
     & 'mi den (kg/m3)','m dn (lbm/ft3)',                        &
     & 'ng den (kg/m3)','ng d (lbm/ft3)',                        &
     & 'ngas mass (kg)','ng mass (lbm) ',                        &
     & 'ng press (pa) ','ng pres (psia)',                        &
     & 'ng ie (w*s/kg)','ng e (btu/lbm)',                        &
     & 'vp ie (w*s/kg)','v ie (btu/lbm)',                        &
     & 'lq ie (w*s/kg)','l ie (btu/lbm)',                        &
     & 'sat temp s (k)','sat temp s (f)',                        &
     & 'sat temp t (k)','sat temp t (f)',                        &
     & 'vcv (w*s/kg/k)','vc (btu/lbm/f)',                        &
     & 'lcv (w*s/kg/k)','lc (btu/lbm/f)',                        &
     & 'ht vp (w*s/kg)','htvp (btu/lbm)',                        &
     & 'shloss vap (w)','shlsv (btu/hr)',                        &
```

```
     & 'shloss liq (w)','shlsl (btu/hr)'/
      DATA ((labsv(i,j),i=1,2),j= 91,105)/                      &
     & 'inf ht flw (w)','ihtfw (btu/hr)',                       &
     & 'v htc (w/m2/k)','(btu/ft2/f/hr)',                       &
     & 'l htc (w/m2/k)','(btu/ft2/f/hr)',                       &
     & 'v htc (w/m2/k)','(btu/ft2/f/hr)',                       &
     & 'l htc (w/m2/k)','(btu/ft2/f/hr)',                       &
     & 'ia*vhtc (w/k) ','avh (btu/f/hr)',                       &
     & 'ia*lhtc (w/k) ','alh (btu/f/hr)',                       &
     & 'yt idc (kg/m4)','yidc (lbm/ft4)',                       &
     & 'z idc (kg/m4) ','zidc (lbm/ft4)',                       &
     & 'xr idc (kg/m4)','xidc (lbm/ft4)',                       &
     & 'ps den (kg/m3)','ps d (lbm/ft3)',                       &
     & 'vgen (kg/m3/s)','vg(lbm/ft3/hr)',                       &
     & 'is ht loss (w)','is hl (btu/hr)',                       &
     & 'os ht loss (w)','os hl (btu/hr)',                       &
     & 'c mix temp (k)','c mix temp (f)'/
      DATA ((labsv(i,j),i=1,2),j=106,106)/                      &
     & 'd/p (kg/m3/pa)','d/p (lb/ft3/p)'/
      DATA (labun(i),i=  1,151)/                                &
     & 'lunounit','lutime  ','lutemp  ','lutempd ','lulength',  &
     & 'luarea  ','luvolume','luvel   ','luacc   ','lupumphd',  &
     & 'luvolflw','luspvol ','lumass  ','lumassfw','lumfwrat',  &
     & 'lumassfx','luvapgen','luden   ','luddendt','luidrag ',  &
     & 'lupressa','lupressd','luprsrat','luminert','lutorque',  &
     & 'lubtork ','luctork ','lupower ','lupowrat','lulinhts',  &
     & 'luheatfx','luvolhts','luthcond','luhtc   ','luihttf ',  &
     & 'luenergy','luspener','luspheat','lurtime ','lurtemp ',  &
     & 'lurmass ','lurpress','luspeed ','luradacc','luangle ',  &
     & 'luburnup','luenfiss','lugapgas','lurtmsq ','lunitnam',  &
     & 'luddendp',100*'luserdef'/
      DATA ((luncb(i,j),i=1,2),j=  1, 15)/                      &
     & ' -           ',' -           ',                         &
     & ' s           ',' s           ',                         &
     & ' k           ',' f           ',                         &
     & ' k           ',' f           ',                         &
```

```
     & ' m           ',' ft          ',                                &
     & ' m2          ',' ft2         ',                                &
     & ' m3          ',' ft3         ',                                &
     & ' m/s         ',' ft/s        ',                                &
     & ' m/s2        ',' ft/s2       ',                                &
     & ' m2/s2       ',' lbf*ft/lbm  ',                                &
     & ' m3/s        ',' gpm         ',                                &
     & ' m3/kg       ',' ft3/lbm     ',                                &
     & ' kg          ',' lbm         ',                                &
     & ' kg/s        ',' lbm/hr      ',                                &
     & ' kg/s2       ',' lbm/s2      '/
      DATA ((luncb(i,j),i=1,2),j= 16, 30)/                             &
     & ' kg/m2/s     ',' lbm/ft2/hr  ',                                &
     & ' kg/m3/s     ',' lbm/ft3/hr  ',                                &
     & ' kg/m3       ',' lbm/ft3     ',                                &
     & ' kg/m3/k     ',' lbm/ft3/f   ',                                &
     & ' kg/m4       ',' lbm/ft4     ',                                &
     & ' pa          ',' psia        ',                                &
     & ' pa          ',' psid        ',                                &
     & ' pa/s        ',' psi/s       ',                                &
     & ' kg*m2       ',' lbm*ft2     ',                                &
     & ' pa*m3       ',' lbf*ft      ',                                &
     & ' pa*m3*s/rad ',' lbf*ft/rpm  ',                                &
     & ' pa*m3*s2/r2 ',' lbf*ft/rpm2 ',                                &
     & ' w           ',' btu/hr      ',                                &
     & ' w/s         ',' btu/hr/s    ',                                &
     & ' w/m         ',' btu/ft/hr   '/
      DATA ((luncb(i,j),i=1,2),j= 31, 45)/                             &
     & ' w/m2        ',' btu/ft2/hr  ',                                &
     & ' w/m3        ',' btu/ft3/hr  ',                                &
     & ' w/m/k       ',' btu/ft/f/hr ',                                &
     & ' w/m2/k      ',' btu/ft2/f/h ',                                &
     & ' w/k         ',' btu/f/hr    ',                                &
     & ' w*s         ',' btu         ',                                &
     & ' w*s/kg      ',' btu/lbm     ',                                &
     & ' w*s/kg/k    ',' btu/lbm/f   ',                                &
```

```
     & ' 1/s          ',' 1/s          ',                                   &
     & ' 1/k          ',' 1/f          ',                                   &
     & ' 1/kg         ',' 1/lbm        ',                                   &
     & ' 1/pa         ',' 1/psi        ',                                   &
     & ' rad/s        ',' rpm          ',                                   &
     & ' rad/s2       ',' rpm/s        ',                                   &
     & ' rad          ',' deg          '/
      DATA ((luncb(i,j),i=1,2),j= 46,151)/                                  &
     & ' mwd/mtu      ',' mwd/mtu      ',                                   &
     & ' mev/fiss     ',' mev/fiss     ',                                   &
     & ' g-moles      ',' g-moles      ',                                   &
     & ' 1/k2         ',' 1/f2         ',                                   &
     & ' *            ',' *            ',                                   &
     & ' kg/m3/pa     ',' lbm/ft3/psi  ',                                   &
     &              200*' *            '/
      DATA ((lupcb(i,j),i=1,2),j=  1, 15)/                                  &
     & '(-)           ','(-)           ',                                   &
     & '(s)           ','(s)           ',                                   &
     & '(k)           ','(f)           ',                                   &
     & '(k)           ','(f)           ',                                   &
     & '(m)           ','(ft)          ',                                   &
     & '(m2)          ','(ft2)         ',                                   &
     & '(m3)          ','(ft3)         ',                                   &
     & '(m/s)         ','(ft/s)        ',                                   &
     & '(m/s2)        ','(ft/s2)       ',                                   &
     & '(m2/s2)       ','(lbf*ft/lbm)  ',                                   &
     & '(m3/s)        ','(gpm)         ',                                   &
     & '(m3/kg)       ','(ft3/lbm)     ',                                   &
     & '(kg)          ','(lbm)         ',                                   &
     & '(kg/s)        ','(lbm/hr)      ',                                   &
     & '(kg/s2)       ','(lbm/s2)      '/
      DATA ((lupcb(i,j),i=1,2),j= 16, 30)/                                  &
     & '(kg/m2/s)     ','(lbm/ft2/hr)  ',                                   &
     & '(kg/m3/s)     ','(lbm/ft3/hr)  ',                                   &
     & '(kg/m3)       ','(lbm/ft3)     ',                                   &
     & '(kg/m3/k)     ','(lbm/ft3/f)   ',                                   &
```

```
     & '(kg/m4)       ','(lbm/ft4)     ',                                    &
     & '(pa)          ','(psia)        ',                                    &
     & '(pa)          ','(psid)        ',                                    &
     & '(pa/s)        ','(psi/s)       ',                                    &
     & '(kg*m2)       ','(lbm*ft2)     ',                                    &
     & '(pa*m3)       ','(lbf*ft)      ',                                    &
     & '(pa*m3*s/rad)','(lbf*ft/rpm) ',                                      &
     & '(pa*m3*s2/r2)','(lbf*ft/rpm2)',                                      &
     & '(w)           ','(btu/hr)      ',                                    &
     & '(w/s)         ','(btu/hr/s)    ',                                    &
     & '(w/m)         ','(btu/ft/hr)   '/
      DATA ((lupcb(i,j),i=1,2),j= 31, 45)/                                   &
     & '(w/m2)        ','(btu/ft2/hr) ',                                     &
     & '(w/m3)        ','(btu/ft3/hr) ',                                     &
     & '(w/m/k)       ','(btu/ft/f/hr)',                                     &
     & '(w/m2/k)      ','(btu/ft2/f/h)',                                     &
     & '(w/k)         ','(btu/f/hr)    ',                                    &
     & '(w*s)         ','(btu)         ',                                    &
     & '(w*s/kg)      ','(btu/lbm)     ',                                    &
     & '(w*s/kg/k)    ','(btu/lbm/f)   ',                                    &
     & '(1/s)         ','(1/s)         ',                                    &
     & '(1/k)         ','(1/f)         ',                                    &
     & '(1/kg)        ','(1/lbm)       ',                                    &
     & '(1/pa)        ','(1/psi)       ',                                    &
     & '(rad/s)       ','(rpm)         ',                                    &
     & '(rad/s2)      ','(rpm/s)       ',                                    &
     & '(rad)         ','(deg)         '/
      DATA ((lupcb(i,j),i=1,2),j= 46,151)/                                   &
     & '(mwd/mtu)     ','(mwd/mtu)     ',                                    &
     & '(mev/fiss)    ','(mev/fiss)    ',                                    &
     & '(g-moles)     ','(g-moles)     ',                                    &
     & '(1/k2)        ','(1/f2)        ',                                    &
     & '(*)           ','(*)           ',                                    &
     & '(kg/m3/pa)    ','(lbm/ft3/psi)',                                     &
     &              200*'(*)           '/
      DATA ((runcb(i,j),i=1,2),j=  1, 15)/                                   &
```

```
     & '             -' ,'             -' ,                              &
     & '             s' ,'             s' ,                              &
     & '             k' ,'             f' ,                              &
     & '             k' ,'             f' ,                              &
     & '             m' ,'            ft' ,                              &
     & '            m2' ,'           ft2' ,                              &
     & '            m3' ,'           ft3' ,                              &
     & '           m/s' ,'          ft/s' ,                              &
     & '          m/s2' ,'         ft/s2' ,                              &
     & '         m2/s2' ,'   lbf*ft/lbm' ,                              &
     & '          m3/s' ,'           gpm' ,                              &
     & '         m3/kg' ,'       ft3/lbm' ,                              &
     & '            kg' ,'           lbm' ,                              &
     & '          kg/s' ,'        lbm/hr' ,                              &
     & '         kg/s2' ,'        lbm/s2' /
      DATA ((runcb(i,j),i=1,2),j= 16, 30)/                              &
     & '       kg/m2/s' ,'   lbm/ft2/hr' ,                              &
     & '       kg/m3/s' ,'   lbm/ft3/hr' ,                              &
     & '         kg/m3' ,'       lbm/ft3' ,                              &
     & '       kg/m3/k' ,'    lbm/ft3/f' ,                              &
     & '         kg/m4' ,'       lbm/ft4' ,                              &
     & '            pa' ,'          psia' ,                              &
     & '            pa' ,'          psid' ,                              &
     & '          pa/s' ,'         psi/s' ,                              &
     & '         kg*m2' ,'       lbm*ft2' ,                              &
     & '         pa*m3' ,'        lbf*ft' ,                              &
     & ' pa*m3*s/rad' ,'   lbf*ft/rpm' ,                              &
     & ' pa*m3*s2/r2' ,' lbf*ft/rpm2' ,                              &
     & '             w' ,'        btu/hr' ,                              &
     & '           w/s' ,'      btu/hr/s' ,                              &
     & '           w/m' ,'     btu/ft/hr' /
      DATA ((runcb(i,j),i=1,2),j= 31, 45)/                              &
     & '          w/m2' ,'   btu/ft2/hr' ,                              &
     & '          w/m3' ,'   btu/ft3/hr' ,                              &
     & '         w/m/k' ,' btu/ft/f/hr' ,                              &
     & '        w/m2/k' ,' btu/ft2/f/h' ,                              &
```

```
     & '          w/k' ,'    btu/f/hr' ,                                  &
     & '          w*s' ,'         btu' ,                                  &
     & '       w*s/kg' ,'     btu/lbm' ,                                  &
     & '     w*s/kg/k' ,'   btu/lbm/f' ,                                  &
     & '          1/s' ,'         1/s' ,                                  &
     & '          1/k' ,'         1/f' ,                                  &
     & '         1/kg' ,'       1/lbm' ,                                  &
     & '         1/pa' ,'       1/psi' ,                                  &
     & '        rad/s' ,'         rpm' ,                                  &
     & '       rad/s2' ,'       rpm/s' ,                                  &
     & '          rad' ,'         deg' /
      DATA ((runcb(i,j),i=1,2),j= 46,151)/                               &
     & '      mwd/mtu' ,'     mwd/mtu' ,                                  &
     & '     mev/fiss' ,'    mev/fiss' ,                                  &
     & '      g-moles' ,'     g-moles' ,                                  &
     & '         1/k2' ,'        1/f2' ,                                  &
     & '            *' ,'           *' ,                                  &
     & '    kg/m3/pa' ,' lbm/ft3/psi' ,                                   &
     &             200*'             *' /
      DATA alpbet/'abcdefghijklmnopqrstuvwxyz'/
      DATA luar,lucp,lud ,ludh,lue ,luen,luh ,luha/8*' '/
      DATA luhx,luid,luis,lum ,lumf,lup ,lupd,luph/8*' '/
      DATA lupt,lupw,lur ,lus ,lusp,lusz,lut /7*' '/
      DATA lutc,lutm,lutp,luv ,luvf,luvo,luz /7*' '/
      END
```

APPENDIX G
TRAC-M Control-Logic BIT Definitions

G.1. INTRODUCTION

For each TRAC-M hydraulic component, the container A array stores a REAL*8 old-time (beginning-of-timestep) BIT array and new-time (end-of-timestep) BITN array with elements for each mesh-cell center or face. BIT and BITN are single dimensioned for 1D hydraulic components and triple dimensioned for the 3D VESSEL component. Bit information is stored in array elements BIT(J) and BITN(J) for the center or negative face of cell J of a 1D hydraulic component and stored in array elements BIT(I,J,K) and BITN(I,J,K) for the center or positive faces of cell I,J,K of a 3D VESSEL component.

Each bit is an off (0) or on (1) indicator for a thermal-hydraulic state condition at the mesh-cell center or face. Currently, 30 bit indicators are defined: bit identification (id) numbers 1 to 17 are for cell-center conditions, and bit id numbers 20 to 32 are for cell-face conditions. The bit id number is the right-to-left bit number in the REAL*8 BIT or BITN word storage format. Section G.2 provides a detailed description of those 30 bits and defines their parameter-constant name (PCN) id numbers.

Five C-language functions and subroutines are used to obtain (access) or define the bit-indicator 0 or 1 values. Integer function BTESTC(BIT(J), PCN) obtains the bit-indicator 0 or 1 value of BIT(J) for bit id number integer PCN. An example of PCN is integer `satLineCrossVap=1` for bit id number 1 (see Sec. G.2). REAL*8 function IBCLRC(BIT(J), PCN) or IBSETC(BIT(J), PCN) defines the bit-indicator 0 or 1 value, respectively, to bit id number integer PCN of BIT(J). Subroutine ON1123C(BIT(J), NCF) defines all bits to be 0 except bit id numbers 2, 11, 12, 13, 30, and 32 for cell centers and cell faces J to J+NCF–1 of array BIT. Subroutine OF1123C(BIT(J), NCF) defines bit id numbers 11, 12, and 13 to be 0 for cell centers and cell faces J to J+NCF –1 of array BIT. BITN(J), BIT(I,J,K), and BITN(I,J,K) can be used in place of BIT(J) in the above usage.

G.2. LISTING OF FILE BITFLAGS.H

The following is a modified listing of header file bitflags.h. The comments differ slightly from the header file bitflags.h in TRAC-M Version 1.10+ because of editing changes made here for correction and further clarity. This documents the purpose and usage of each of the defined 30 bits of arrays BIT and BITN.

```
! *** header file bitflags.h ***
!
! The array fbit, which holds unchanging geometric information
! for the 3D hydro, is not discussed in this file.
!
! Note that TRAC now uses a bit-numbering convention from "right
! to left", starting with bit 1. The F90 intrinsic bit manipula-
! tion routines (ibset, btest, etc.) also go from "right to left",
! but start with bit number 0.
!
! Note that the bitn arrays are cleared with 0.0 (floating point
! zero). The Cray and the supported IEEE platforms all represent
! 0.0 as all-zeros.
!
! This version of header file bitflags.h re-maps the original bit
! identification (id) numbers (i.e., as used in the pre-branch
! code -- TRAC-P Version 5.4.25) into the range 1 - 32; it also
! groups all the id numbers according to whether the bit is
! defined for a mesh-cell center or face.
!
! The bit current id numbers, original id numbers, and mesh-cell
! center or face defined form are as follows:
!
! **************  ***************  ****************
! Current number  Original number  Cell center/face
! **************  ***************  ****************
!
!       1                42                c
!       2                43                c
!       3                 3                c
!       4                 4                c
!       5                 5                c
!       6                 6                c
!       7                20                c
!       8                21                c
!       9                24                c
!      10                34                c
```

```
!        11                 11                c
!        12                 12                c
!        13                 13                c
!        14                 26                c
!        15                 27                c
!        16                 29                c
!        17                 30                c
!        18 ***not used***
!        19 ***not used***
!        20                 18                f
!        21                 19                f
!        22                 22                f
!        23                 23                f
!        24                 10                f
!        25                 25                f
!        26                 44                f
!        27                 45                f
!        28                 28                f
!        29                 46                f
!        30                 33                f
!        31                 31                f
!        32                 32                f
!
! Bits 33 and higher ***not used***
!
! The current bit id-number definitions of purpose, where set,
! and where used and their integer parameter-constant id-number
! declarations are as follows:
!
! *************
! *** Bit 1 ***
!
! Purpose: Used in interfacial heat transfer logic, to determine
!          if the gas temperature crossed the saturation line
!          since the previous timestep. Bit 1 is set on in the
!          new-time (bitn) array if gas temperature tvn is greater
!          than tssn (saturation temperature at steam partial
!          pressure). If comparison with the old-time (bit) array
```

```
!           shows the saturation line was crossed, the relaxation-
!           limiter logic (in a transient calculation) on changes
!           in chti (vapor interfacial heat transfer coefficient x
!           interfacial area) and chtia (noncondensable-gas inter-
!           facial htc x interfacial area) is bypassed.
!
!           Use is identical for 1D, 3D, and plenum.
!
! Set in:   htif (outer stage - 1D, 3D, and plenum) -- bitn also
!           cleared
!
! Used in: htif (outer stage - 1D, 3D, and plenum)
!
      INTEGER satLineCrossVap
      PARAMETER (satLineCrossVap=1)
!
! *************
! *** Bit 2 ***
!
! Purpose: Bit 2 is the liquid analog of bit 1; its logic for
!           clearing, setting, and testing in routine htif is the
!           same as for bit 1, comparing liquid temperature tln to
!           tssn. If the liquid temperature crossed the saturation
!           line since the last timestep, the transient relaxation-
!           limiter logic on changes in alve (liquid-side interfa-
!           cial htc x interfacial area) and alv (flashing inter-
!           facial htc x interfacial area) is bypassed.
!
!           Use is identical for 1D, 3D, and plenum (but see fol-
!           lowing note).
!
!           Note that subroutine inner calls subroutine on1123c.
!           Subroutine on1123c clears all bits except 2, 11, 12,
!           13, 30, and 32. The intention is to clear all new-
!           time bits in the bitn array except for water-packer
!           flags and bits set in the prep stage, for 1D components
!           (the plenum is excluded). Protecting bit 2 is no
!           longer needed and in any event this logic is not par-
```

```
!          allel with that for bit 1. This does not appear to
!          cause an actual error in the calculation, but it should
!          be further investigated and at least cleaned up.
!
! Set in:  htif (outer stage - 1D, 3D, and plenum) -- bitn also
!          cleared
!
! Used in: htif (outer stage - 1D, 3D, and plenum)
!
      INTEGER satLineCrossLiq
      PARAMETER (satLineCrossLiq=2)
!
! *************
! *** Bit 3 ***
!
! Purpose: Used in reiteration logic when the gas volume fraction
!          is out of bounds in basic (outer) step. If the gas
!          volume fraction exceeds tolerance of 10(-12) (i.e., if
!          .le. -1.0e-12 or .ge. (1.0+1.0e-12)), bit 3 is set on
!          and the logical reiteration flag is set to .true.. If
!          bit 3 has been set on a previous iteration, this test
!          on the gas volume fraction is bypassed.
!
!          Usage identical in 1D, 3D, and plenum hydro.
!
! Set in:  tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!          tfplbk (outer stage - plenum)
!
! Used in: tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!          tfplbk (outer stage - plenum)
!
      INTEGER oneVoidFrReit
      PARAMETER (oneVoidFrReit=3)
!
! *************
! *** Bit 4 ***
```

```
!
! Purpose: Two distinct uses. In the initialization stage, bit 4
!          is set to indicate that internally-used FRICs have been
!          calculated from user-input K factors (this logic is
!          part of the input-error checking for consistency at
!          component junctions). During the calculation, bit 4 is
!          set to indicate the mean mass equation will be solved
!          rather than the gas and liquid mass equations (flow is
!          single phase or nearly single phase).
!
!          The input-checking-use of bit 4 is for 1D components.
!          The hydro-use of bit 4 is similar in 1D, 3D, and
!          plenum.
!
!          The parameter meanEqnSet is only meant to be used for
!          the hydro calculation, for 1D, 3D, and plenum.
!
! Set in:  chbset (init stage)
!          tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
! Used in: chkbd (init stage)
!          tf1ds (outer stage - 1D), tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!          tfpln (outer stage - plenum), tfplbk (outer stage -
!               plenum)
!
      INTEGER meanEqnSet
      PARAMETER (meanEqnSet=4)
!
! *************
! *** Bit 5 ***
!
! Purpose: Used in calculation of interfacial heat and mass trans-
!          fer in basic (outer) step. Bit 5 is set on in a hydro
!          cell for condensation conditions (negative gamma and
!          gas volume fraction greater than zero; see following
```

```
!          note on plenum).
!
!          Use is very similar in 1D, 3D, and plenum. Plenum
!          logic for setting does not have test on the gas volume
!          fraction.
!
! Set in:  tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
! Used in: tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
      INTEGER condensing
      PARAMETER (condensing=5)
!
! *************
! *** Bit 6 ***
!
! Purpose: Evaporation/flashing analog of bit 5. Bit 6 is set on
!          if gamma is positive and the gas volume fraction is
!          less than one.
!
!          Use is very similar in 1D, 3D, and plenum. Plenum
!          logic for setting does have test on the gas volume
!          fraction.
!
! Set in:  tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
! Used in: tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
      INTEGER evapOrFlashing
      PARAMETER (evapOrFlashing=6)
```

```
!
! *************
! *** Bit 7 ***
!
! Purpose: When bit 7 is on, the old-time/new-time weighting
!          factor for donor-cell quantities used in the 1D and
!          plenum mass and energy equations is set to 1.0. This
!          forces the fluxes to 100% new-time weighting. The
!          explicit/implicit weighting factor is local variable
!          xvset, which is also local array dalp, which is array
!          rhs in the 1D and plenum data.
!
!          Bit 7 is used in similar fashion by 1D and plenum; it
!          is not used by 3D for any purpose, including the 3D
!          xvset logic. Bit 7 is cleared in subroutine htif for
!          all components, but this has no effect on 3D. htif is
!          only called on the first Newton iteration (oitno=1);
!          once bit 7 is set for a given series of iterations, it
!          remains set.
!
! Set in:  htif (outer stage - 1D, 3D, and plenum) -- bitn cleared
!          tf1ds (outer stage - 1D)
!          tfpln (outer stage - plenum)
!
! Used in: tf1ds (outer stage - 1D)
!          tfpln (outer stage - plenum)
!
!
      INTEGER freezeXvset
      PARAMETER (freezeXvset=7)
!
! *************
! *** Bit 8 ***
!
! Purpose: Used in the equation-set logic in basic (outer) stage.
!          Set in the back-substitution routines for use in
!          subsequent iterations for a given timestep for basic
!          energy equation. Bit 8 is set on for situation of
```

```
!          almost, but not quite, solid water in a cell (very
!          small bubbles). The old-time gas volume fraction must
!          be .le. 1.0e-8 and the new-time gas volume fraction
!          must be .lt. 1.0e-12. When on, bit 8 forces the gas
!          temperature to equal the saturation temperature corres-
!          ponding to the partial pressure of steam.
!
!          The gas volume fraction test for setting bit 8 has been
!          modified by update fixb21 (bit 8 was bit 21 in Version
!          5.4.25).
!
!          Use is same in 1D, 3D, and plenum.
!
! Set in:  tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!          tfplbk (outer stage - plenum)
!
!
! Used in: tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
      INTEGER tinyBubbles
      PARAMETER (tinyBubbles=8)
!
! *************
! *** Bit 9 ***
!
! Purpose: Set in basic (outer) step when special logic is used to
!          change the current guess for the new-time value of the
!          gas volume fraction before linearization. If bit 9 is
!          set (from a previous iteration), the special gas volume
!          fraction logic is bypassed (i.e., the bit is used to
!          allow only one use of this logic in a given series of
!          of Newton iterations).
!
!          1D, 3D, and plenum logic the same (1D and plenum use
!          old and new time bits 20 and 21 for velocity-reversal
```

```
!          information; 3D uses old/new time donor-cell factor
!          arrays (owlz, wlz, etc.) for same purpose.
!
! Set in:  tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
! Used in: tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
      INTEGER triedVoidFrReset
      PARAMETER (triedVoidFrReset=9)
!
! **************
! *** Bit 10 ***
!
! Purpose: Used in 3D hydro only (there is identical logic in the
!          1D that does not use a bit flag). Bit 10 is set on for
!          a mesh cell when the net noncondensable-gas flow into
!          the cell is .gt. 1.0e-20 kg for the current timestep.
!          Bit 10 is subsequently used in the same step in the
!          logic to set an initial guess for the noncondensable-
!          gas partial pressure. If bit 10 is not on, the initial
!          guess is bypassed (there are other tests that also can
!          bypass the noncondensable-gas logic). The initial
!          noncondensable-gas partial pressure guess is the total
!          pressure minus the saturation pressure corresponding to
!          the current liquid temperature.
!
!          The 1-D logic that corresponds to that for bit 10 is
!          in subroutine tf1ds3, at statement label 1337 in the
!          pre-branch code (Version 5.4.25); the noncondensable-
!          gas flow is in array dr.
!
! Set in:  tf3ds (outer stage - 3D)
!
! Used in: tf3ds3 (outer stage - 3D)
```

```
!
      INTEGER netAirFlow
      PARAMETER (netAirFlow=10)
!
! **************
! *** Bit 11 ***
!
! Purpose: Used with bits 12 and 13 in 1D water packing/stretch
!          logic. Used with bit 13 in 3D water pack/stretch
!          logic. Not used by plenum.
!
!          Water packing and stretching are checked for in each 1D
!          and 3D cell at the start of the back-substitution rou-
!          tines of the outer stage (tf1ds3 and tf3ds3 for 1D and
!          3D, respectively). If water packing is detected the
!          back substitution is skipped and backup to the start of
!          outer is forced. In the 1D bit 11 is set on for pack-
!          ing or stretching at a cell's left face (bit 12 is used
!          for the right face); in the case of a stretch bit 13 is
!          also set on. In the 3D bit 11 indicates packing and
!          bit 13 indicates stretch for the cell (the stretch
!          information is passed to the bd array by routine j3d).
!
!          Note that subroutine inner calls subroutine on1123c
!          to clear all 1D-component bits except 2, 11, 12, 13,
!          30, and 32 (see additional notes on bit 2). Subroutine
!          poster calls subroutine of1123c to clear 1D-component
!          bits 11, 12, and 13 if water packing flag ipakon .ne. 0
!          (bit and bitn arrays). j3d (vessel source junction
!          boundary array routine) also calls of1123c for bd(53).
!
!          Parameter packAtLeftFace is intended for 1D use.
!          Parameter pack3D is intended for 3D use.
!
! Set in:  j3d -- bd(53) only
!          tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!          poster (post stage - 1D) -- bitn and bit cleared if
```

```
!          ipakon .ne. 0
!
! Used in: tf1ds (outer stage - 1D)
!          tf1ds1 (outer stage - 1D)
!          tf1ds3 (outer stage - 1D)
!          tf3ds1 (outer stage - 3D) -- bit 13 not used
!          tf3ds3 (outer stage - 3D) -- bit 13 not used
!
      INTEGER packAtLeftFace
      INTEGER pack3D
      PARAMETER (packAtLeftFace=11)
      PARAMETER (pack3D=11)
!
! **************
! *** Bit 12 ***
!
! Purpose: Used with bits 11 and 13 in 1D water packing/stretch
!          logic. Not used by 3D or plenum. Indicates pack or
!          stretch detected at 1D cell's right face. See addi-
!          tional notes under bit 11.
!
!          See bit 11 on use of subroutines on1123c and of1123c.
!
! Set in:  j3d -- bd(53) only
!          tf1ds3 (outer stage - 1D)
!          poster (post stage - 1D) -- bitn and bit cleared if
!                                      ipakon .ne. 0
!
! Used in: tf1ds (outer stage - 1D)
!          tf1ds1 (outer stage - 1D)
!          tf1ds3 (outer stage - 1D)
!
      INTEGER packAtRightFace
      PARAMETER (packAtRightFace=12)
!
! **************
! *** Bit 13 ***
!
```

```
! Purpose: Used with bits 11 and 12 in 1D water packing/stretch
!          logic. Used with bit 11 in 3D water pack/stretch
!          logic. Not used by plenum.
!
!          See bit 11 on use of subroutines on1123c and of1123c.
!
!          Parameter stretch is intended for 1D use.
!          Parameter stretch3D is intended for 3D use (this is
!          passed to the bd array by routine j3d).
!
! Set in:  j3d -- bd(53) only
!          tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!          poster (post stage - 1D) -- bitn and bit cleared if
!                                      ipakon  .ne. 0
!
! Used in: tf1ds1 (outer stage - 1D)
!          tf1ds3 (outer stage - 1D)
!
      INTEGER stretch
      INTEGER stretch3D
      PARAMETER (stretch=13)
      PARAMETER (stretch3D=13)
!
! **************
! *** Bit 14 ***
!
! Purpose: Used in timestep-size control logic, in conjunction
!          with bit 15. Bits 14 and 15, used with the gas volume
!          fraction arrays alpn, alp, and alpo, save the gas-
!          volume-fraction change behavior looking back over three
!          timesteps. Bits 14 and 15 control calculation of var-
!          iables oau and oal (in common block chgalp), which are
!          used in subroutine newdlt to determine the timestep
!          size at the start of the next timestep. oau is the
!          largest increase in the gas volume fraction in the
!          system immediately after a decrease, which in turn had
!          followed an increase (all for a given hydro cell). oal
```

```
!           measures the analogous situation for a decrease in the
!           gas volume fraction. Bit 14 is set on in the bitn
!           array for a hydro cell when the gas volume fraction
!           has increased in that cell with respect to the previous
!           timestep.
!
!           Use of bit 14 is identical in 1D, 3D, and plenum.
!
!           Note that blkdat now sets variables xoau and xoal
!           (common block chgalp) to 1.0, which effectively turns
!           off the oscillating-gas-volume-fraction (oau or oal)
!           timestep-size control. Gas-volume-fraction-change
!           timestep-size control now only uses variables dau and
!           dal, which only look back to the previous timestep.
!           The dau/dal logic only needs arrays alpn and alp, and
!           not bits 14 and 15.
!
! Set in:   poster (post stage - 1D)
!           bkstb3 (post stage - 3D)
!           plen3 (post stage - plenum)
!
! Used in:  poster (post stage - 1D)
!           bkstb3 (post stage - 3D)
!           plen3 (post stage - plenum)
!
      INTEGER newVoidFrUp
      PARAMETER (newVoidFrUp=14)
!
! **************
! *** Bit 15 ***
!
! Purpose:  Used in conjunction with bit 14 for oscillating-gas-
!           volume-fraction timestep-size control. Bit 15 is set
!           on (bitn array) for a hydro cell when old-time bit 14
!           (bit array) is on (i.e., when the gas volume fraction
!           increased during previous timestep). Bit 15 is saved
!           in the old-time bit array, for use in the oau/oal
!           logic.
```

```
!
!          Use of bit 15 is identical in 1D, 3D, and plenum.
!
!          Same note applies concerning variables xoau and xoal
!          as for bit 14.
!
! Set in:  poster (post stage - 1D)
!          bkstb3 (post stage - 3D)
!          plen3 (post stage - plenum)
!
! Used in: poster (post stage - 1D)
!          bkstb3 (post stage - 3D)
!          plen3 (post stage - plenum)
!
      INTEGER oldVoidFrUp
      PARAMETER (oldVoidFrUp=15)
!
! **************
! *** Bit 16 ***
!
! Purpose: Set on for a cell when the net mass flow into the cell
!          is negative. When bit 16 is on, the water pack/stretch
!          logic in the back-substitution routines is bypassed.
!
!          Use is same in 1D, 3D, and plenum.
!
! Set in:  tf1ds (outer stage - 1D) -- always cleared before logic
!                                      for setting
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
! Used in: tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!          tfplbk (outer stage - plenum)
!
      INTEGER netMassOut
      PARAMETER (netMassOut=16)
!
```

```
! **************
! *** Bit 17 ***
!
! Purpose: Used in equation-set logic. The back-substitution
!          routines have logic to force the gas volume fraction to
!          1.0 or 0.0 if bit 4 (for one of the single-phase mass
!          equation sets) is on. If bit 17 is also on, forcing
!          the gas volume fraction to 0.0 is bypassed. Instead,
!          an equation is used to set steam pressure to the sat-
!          uration pressure corresponding to the liquid tempera-
!          ture.
!
!          Use is same in 1D, 3D, and plenum.
!
! Set in:  tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!          tfpln (outer stage - plenum)
!
! Used in: tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!          tfplbk (outer stage - plenum)
!
      INTEGER specEqnSteamP
      PARAMETER (specEqnSteamP=17)
!
! **************
! *** Bit 18 *** not used
!
! **************
! *** Bit 19 *** not used
!
! **************
! *** Bit 20 ***
!
! Purpose: For 1D and plenum hydro only (including break and fill
!          components). Cell-face flag to indicate gas-velocity
!          direction; used in logic for gas donor-cell weighting
!          factors and gas-velocity reversal. The gas-velocity
```

```
!         reversal information is used with corresponding liquid
!         information in the reiteration logic (see bits 22 and
!         23), and alone in the interfacial shear logic (see bit
!         26). Gas-velocity reversal information also is used
!         in the logic for the special gas-volume-fraction guess
!         (see bit 9).
!
!         Bit 20 is set on when the gas velocity is negative.
!
! Set in:  tf1ds1 (outer stage - 1D)
!          tf1ds3 (outer stage - 1D)
!
! Used in: break1 (prep stage) -- bd(38) only
!          fill1 (prep stage) -- bd(38) only
!          flux (prep stage - 1D)
!          auxpln (outer stage - plenum) - bd(53) only
!          tf1ds (outer stage - 1D)
!          tf1ds1 (outer stage - 1D) -- bd(53) only
!          tf1ds3 (outer stage - 1D)
!          tfpln (outer stage - plenum) -- bd(38) and bd(53) only
!          poster (post stage - 1D) -- bit 21 not used
!          stbme (post stage - 1D)
!          tee3 (post stage - 1D)
!
      INTEGER negVapVel
      PARAMETER (negVapVel=20)
!
! **************
! *** Bit 21 ***
!
! Purpose: Liquid analog of bit 20. For 1D and plenum hydro only
!          (including break and fill components). Cell-face flag
!          to indicate liquid-velocity direction; used in logic
!          for liquid donor-cell weighting factors and liquid-
!          velocity reversal. The liquid-velocity reversal infor-
!          mation is used with corresponding gas information in
!          the reiteration logic (see bits 22 and 23); there is
!          no corresponding use in the gas-direction interfacial
```

```
!          shear logic (see bit 26). Liquid-velocity reversal
!          information also is used in the logic for the special
!          gas-volume-fraction guess (see bit 9).
!
!          Bit 21 is set on when the liquid velocity is negative.
!
! Set in:  tf1ds1 (outer stage - 1D)
!          tf1ds3 (outer stage - 1D)
!
! Used in: break1 (prep stage) -- bd(38) only
!          fill1 (prep stage) -- bd(38) only
!          flux (prep stage - 1D)
!          auxpln (outer stage - plenum) - bd(53) only
!          tf1ds (outer stage - 1D)
!          tf1ds1 (outer stage - 1D) -- bd(53) only
!          tf1ds3 (outer stage - 1D)
!          tfpln (outer stage - plenum) -- bd(38) and bd(53) only
!          stbme (post stage - 1D)
!          tee3 (post stage - 1D)
!
      INTEGER negLiqVel
      PARAMETER (negLiqVel=21)
!
! **************
! *** Bit 22 ***
!
! Purpose: Used in logic that determines if a reiteration is
!          forced by a flow reversal. Bit 22 is set on if the gas
!          mass-flow threshold for a flow reversal reiteration is
!          exceeded. This threshold is set by variable frev (com-
!          mon block xvol).
!
!          Used in similar fashion for 1D and 3D; not used by
!          plenum. In 3D, bit 22 is for radial (or x) face (bits
!          24 and 25 are used for same purpose for axial and theta
!          (or y) faces). For 1D, new-time bit 20 is first used
!          to check for a gas flow reversal; then bit 22 is used
!          to see if the gas mass flow sensitivity level has been
```

```
!          exceeded.
!
!          Parameter significantVapFlow is intended for 1D use.
!          Parameter significantVapFlowxr is intended for 3D use.
!
! Set in:  tf1ds (outer stage - 1D)
!          tf3ds (outer stage - 3D)
!
! Used in: tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!
      INTEGER significantVapFlow
      INTEGER significantVapFlowxr
      PARAMETER (significantVapFlow=22)
      PARAMETER (significantVapFlowxr=22)
!
! **************
! *** Bit 23 ***
!
! Purpose: Liquid analog of bit 22 (similar 3D use for bits 28 and
!          31). Used in logic that determines if a reiteration is
!          forced by a flow reversal. Bit 23 is set on if the
!          liquid mass-flow threshold for a flow reversal reiter-
!          ation is exceeded. This threshold is set by variable
!          frev (common block xvol).
!
!          Used in similar fashion for 1D and 3D; not used by
!          plenum. In 3D, bit 23 is for radial (or x) face (bits
!          28 and 31 are used for same purpose for theta (or y)
!          and axial faces). For 1D, new-time bit 21 is first
!          used to check for a liquid flow reversal; then bit 23
!          is used to see if the liquid mass-flow sensitivity
!          level has been exceeded.
!
!          Parameter significantLiqFlow is intended for 1D use.
!          Parameter significantLiqFlowxr is intended for 3D use.
!
! Set in:  tf1ds (outer stage - 1D)
```

```
!          tf3ds (outer stage - 3D)
!
! Used in: tf1ds3 (outer stage - 1D)
!          tf3ds3 (outer stage - 3D)
!
      INTEGER significantLiqFlow
      INTEGER significantLiqFlowxr
      PARAMETER (significantLiqFlow=23)
      PARAMETER (significantLiqFlowxr=23)
!
! **************
! *** Bit 24 ***
!
! Purpose: 3D hydro only; same use as bit 22, but for axial face.
!          Uses variable frev for gas-flow threshold.
!
! Set in:  tf3ds (outer stage - 3D)
!
! Used in: tf3ds3 (outer stage - 3D)
!
      INTEGER significantVapFlowz
      PARAMETER (significantVapFlowz=24)
!
! **************
! *** Bit 25 ***
!
! Purpose: 3D hydro only; same use as bit 22, but for theta (or y)
!          face. Uses variable frev for gas-flow threshold.
!
! Set in:  tf3ds (outer stage - 3D)
!
! Used in: tf3ds3 (outer stage - 3D)
!
      INTEGER significantVapFlowyt
      PARAMETER (significantVapFlowyt=25)
!
! **************
! *** Bit 26 ***
```

```
!
! Purpose: Set in post stage to indicate the gas velocity has
!          changed direction during the timestep being completed.
!          Used in prep stage of subsequent timestep in calcula-
!          tion of interfacial shear coefficients. If bit is on,
!          relaxation-limiter logic for interfacial shear coeffi-
!          cient (used in transient calculation) is turned off.
!
!          Used in similar fashion for 1D and 3D; not used by
!          plenum. In 3D, bit 26 is for the theta (or y) face
!          (bits 27 and 29 are used for the same purpose for the
!          axial and radial (or x) faces). 1D hydro sets new-time
!          bit 26 according to the status of old-time bit 20 and
!          the new-time gas velocity; 3D hydro sets bit 26 accord-
!          ing to the status of old-time and new-time donor-cell
!          factors for gas at the theta (or y) face (arrays owvyt
!          and wvyt).
!
!          Parameter changeVapVel is intended for 1D use.
!          Parameter changeVapVelyt is intended for 3D use.
!
! Set in:  poster (post stage - 1D)
!          ff3d (post stage - 3D)
!
! Used in: femom (prep stage - 1D)
!          cif3 (prep stage - 3D)
!
      INTEGER changeVapVel
      INTEGER changeVapVelyt
      PARAMETER (changeVapVel=26)
      PARAMETER (changeVapVelyt=26)
!
! **************
! *** Bit 27 ***
!
! Purpose: 3D hydro only; same use as bit 26, but for axial face.
!          Set according to status of arrays owvz and wvz.
!
```

```
! Set in:  ff3d (post stage - 3D)
!
! Used in: cif3 (prep stage - 3D)
!
      INTEGER changeVapVelz
      PARAMETER (changeVapVelz=27)
!
! **************
! *** Bit 28 ***
!
! Purpose: 3D hydro only; same use as bit 23, but for theta (or y)
!          face. Uses variable frev for liquid-flow threshold.
!
! Set in:  tf3ds (outer stage - 3D)
!
! Used in: tf3ds3 (outer stage - 3D)
!
      INTEGER significantLiqFlowyt
      PARAMETER (significantLiqFlowyt=28)
!
! **************
! *** Bit 29 ***
!
! Purpose: 3D hydro only; same use as bit 26, but for radial
!          (or x) face. Set according to status of arrays owvxr
!          and wvxr.
!
! Set in:  ff3d (post stage - 3D)
!
! Used in: cif3 (prep stage - 3D)
!
      INTEGER changeVapVelxr
      PARAMETER (changeVapVelxr=29)
!
! **************
! *** Bit 30 ***
!
! Purpose: Flag for the choked-flow model. Bit 30 is set on for
```

```
!          a cell edge if subroutine choke determines choked flow
!          exists at the cell edge (the evaluation of the choked-
!          flow model at a cell edge is invoked by user input).
!          1D only.
!
!          Subroutine choke is called by subroutine femom.
!          Subroutines femom and tf1ds1 use bit 30 to apply the
!          choked-flow velocity. The choked-flow velocity is not
!          applied by subroutine tf1ds1 if flow is into a water-
!          packed cell.
!
!          Bit 30 is one of the bits "protected" by subroutine
!          on1123c (see notes on bit 11).
!
!          If bit 30 is on, subroutine ecomp prints -1.111e-11
!          for the liquid wall friction for 1D components. This
!          value is also written to the xtv graphics file.
!
! Set in:  femom (prep stage - 1D) -- calls choke
!
! Used in: ecomp (large edits for 1D)
!          graf (edits to xtv graphics file)
!          femom (prep stage - 1D)
!          tf1ds1 (outer stage - 1D)
!          tf1ds3 (outer stage - 1D)
!
      INTEGER chokedFlowOn
      PARAMETER (chokedFlowOn=30)
!
! **************
! *** Bit 31 ***
!
! Purpose: 3D hydro only; same use as bit 23, but for axial face.
!          Uses variable frev for liquid-flow threshold.
!
! Set in:  tf3ds (outer stage - 3D)
!
! Used in: tf3ds3 (outer stage - 3D)
```

```
!
      INTEGER significantLiqFlowz
      PARAMETER (significantLiqFlowz=31)
!
! **************
! *** Bit 32 ***
!
! Purpose: Used to control the choked-flow model when namelist
!          variable icflow is 2 (which invokes user-control of
!          the model at all 1D cell faces). Bit 32 is set during
!          the input stage when icflow .eq. 2 and component input-
!          array icflg is nonzero for a cell face (indicating the
!          choked-flow model is to be evaluated for the face).
!          1D only.
!
!          If icflow .eq. 2 and bit 32 is not set, the call to
!          subroutine choke in subroutine femom is bypassed (see
!          bit 30).
!
!          Bit 32 is one of the bits "protected" by subroutine
!          on1123c (see notes on bit 11).
!
!          Bit 32 also is used by subroutine chkbd as an input
!          check on the consistency of the choked-flow-option
!          input-array icflg at component junctions.
!
! Set in:  rcomp (input for 1D)
!          preper (prep stage - 1D) -- after bitn is cleared,
!                                      bit 32 is reset in bitn
!                                      if old-time bit 32 was on
!
! Used in: chkbd (boundary array consistency check)
!          preper (prep stage - 1D) -- only to reset new-time bit
!                                      32
!          tf1ds1 (outer stage - 1D)
!
      INTEGER userChokeControl
      PARAMETER (userChokeControl=32)
```

```
!
! *************************
! *** bits 33 and higher *** not used
!
!
! *** end header file bitflags.h ***
```