KfK 5010
Mai 1992

FLUTAN
Input Specifications

H. Borgwaldt, W. Baumann, G. Willerding
Institut für Neutronenphysik und Reaktortechnik
Projekt Nukleare Sicherheitsforschung

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE

Institut für Neutronenphysik und Reaktortechnik

Projekt Nukleare Sicherheitsforschung

KfK 5010

FLUTAN
INPUT SPECIFICATIONS

H. Borgwaldt, W. Baumann, G. Willerding

Kernforschungszentrum Karlsruhe GmbH , Karlsruhe

Als Manuskript gedruckt

Kernforschungszentrum Karlsruhe GmbH
Postfach 3640, 7500 Karlsruhe 1

ISSN 0303-4003

FLUTAN - INPUT SPECIFICATIONS

Abstract

FLUTAN is a highly vectorized computer code for 3-D fluiddynamic and thermal-hydraulic analyses in cartesian and cylinder coordinates. It is related to the family of COMMIX codes originally developed at Argonne National Laboratory, USA.

To a large extent, FLUTAN relies on basic concepts and structures imported from COMMIX-1B and COMMIX-2 which were made available to KfK in the frame of cooperation contracts in the fast reactor safety field.

While on the one hand not all features of the original COMMIX versions have been implemented in FLUTAN, the code on the other hand includes some essential innovative options like CRESOR solution algorithm, general 3-dimensional rebalancing scheme for solving the pressure equation, and LECUSSO-QUICK-FRAM techniques suitable for reducing "numerical diffusion" in both the enthalphy and momentum equations.

This report provides users with detailed input instructions, presents formulations of the various model options, and explains by means of comprehensive sample input, how to use the code.

FLUTAN - EINGABEBESCHREIBUNG

Zusammenfassung

FLUTAN ist ein hoch-vektorisierter Computercode für 3-dimensionale fluiddynamische und thermohydraulische Analysen sowohl in kartesischen als auch Zylinderkoordinaten. Er gehört zur Familie der ursprünglich am ANL (USA) entwickelten COMMIX-Programme.

Die in FLUTAN vorhandenen grundlegenden Konzepte und Strukturen entsprechen weitgehend den Vorläufern COMMIX-1B und COMMIX-2, die im Rahmen einer Zusammenarbeit auf dem Gebiet der Schnellbrüter-Sicherheit für KfK zugänglich waren.

Während der Code einerseits nicht alle physikalischen Modelle und Optionen aus früheren COMMIX-Versionen besitzt, wurde er andererseits mit wesentlichen Neuerungen ausgestattet. Diese sind z.B. der CRESOR-Algorithmus, die allgemeine 3-dimensionale Grobmaschen-Bilanzierung zur Lösung der Druckgleichung sowie die LECUSSO-QUICK-FRAM-Methode zur Verminderung "numerischer Diffusion" beim Impuls- und Enthalpietransport.

Der vorliegende Bericht vermittelt eine detaillierte Anleitung zur Benutzung des Codes, zeigt die mathematischen Formulierungen einer Reihe von Modellen und erläutert die Eingabe anhand eines umfassenden Beispiels.

FLUTAN

A Computer-Code for 3-Dimensional Fluid- and Thermo-Dynamic Analysis in Cartesian or Cylinder Coordinates.

FLUTAN is a highly vectorized program related to the family of COMMIX codes, which were originally developed at the Argonne National Laboratory (ANL), USA. Optimization and vectorization was done in the Institut für Neutronenphysik und Reaktortechnik (INR) of the Kernforschungszentrum Karlsruhe (KfK), initially on a CYBER-205 vector computer, later on Fujitsu/Siemens VP50 and VP400-EX. Program language is FORTRAN-77.

To a large extent, FLUTAN uses basic concepts and structures imported from the codes COMMIX-1B (cf. Ref./1/, /2/) and COMMIX-2 (cf. Ref./3/), which KfK was able to obtain from ANL in the frame of a US-German cooperation on fast reactor safety. Users completely unfamiliar with the general design of the COMMIX codes are advised first to consult the reports just mentioned and, especially, a recent report on COMMIX-1C (cf. Ref./4/, /5/).

Not all features of the original code versions have been implemented in FLUTAN. On the other hand, our version includes some essential innovations, e.g. the CRESOR algorithm (cf. Ref./6/) and general 3-dimensional rebalancing for solving the pressure equation, as well as the QUICK-LECUSSO-FRAM techniques (cf. Ref./7/, /8/, /9/) to tackle the numerical diffusion problems, both for the enthalpy and momentum equations.

Contents

Figures

General Comments

The units used in FLUTAN are meter, kilogram, second, and degrees Celsius. These and other derived units are indicated after the description of variables requiring them.

Default values are indicated either by an asterisk or a value in parentheses after the variable description.

Arrays are indicated by the use of a subscript following the variable name. The ranges of the subscripts are indicated in the following table.

Index	Range	Current Limit
I	IMAX	IJKMAX
J	JMAX	IJKMAX
K	KMAX	IJKMAX
N	NSURF	IJKMAX
L	NL1	LBOUND
M0	NM1	LCELL
NH	NHEATC	10
NM	NMATER	10
NF	NFORCE	100
NC	NCORR	20

The range limits IJKMAX, LBOUND, LCELL must be set properly at compile time by a PARAMETER statement.

Some Terminology

The computational area is partitioned into a number of computational cells, each bounded by consecutive X, Y, and Z direction grid planes. Surfaces (portions of a plane or cylinder) may be defined both on the exterior, bounding the computational area, and in the interior. The intersection of a surface and consecutive grid planes outlines a surface element. Surfaces which coincide with a grid plane are called regular surfaces, otherwise, they are called irregular surfaces. A regular cell is one with all faces coinciding with grid planes. Irregular cells have one irregular surface element.

General Input Structure

Input for FLUTAN can be described in one of two ways:

1. Cartesian Geometry: IGEOM = 0
2. Cylinder Geometry: IGEOM = - 1

Both geometry options allow the user to describe the geometry in terms of the cells formed by the X, Y, and Z or by the R, Θ, and Z grid planes. A typical input sequence is as follows:

Problem Description Records	(Optional)
NAMELIST /GEOM/	
Boundary Surface Identification Records	
NAMELIST /DATA/	
NAMELIST /TURB/	
Force Structure Specification Records	(Optional)
Thermal Structure Prototype Records	(Optional)
Thermal Structure Location Records	(Optional)
Boundary Value Initialization Records	
Internal Cell Initialization Records	
NAMELIST /REBAL/	(Optional)
Rebalancing Region Records	(Optional)

Problem Description Records

Any number of records with user comments can precede NAMELISTS or be interspersed between non-NAMELIST input as long as columns 1-4 are left blank. Comment text may also follow the data in formatted input records and may especially start in column 5 of a record with END plus trailing blank in columns 1-4.

Reserved Key Words

Input to FLUTAN is a mixture of NAMELISTs, formatted records and comments. When processing formatted input, columns 1-4 of each line of this type of input is compared with a group of key words. When a match is found the line is reread in the appropriate format. (If blanks are found the record is treated as a comment record.) The actual list of key words follows (_ indicates a leading blank).

AL	HL	P	RL	UREB	YFOR
ALX	HLB	PB	RLB	VELBN	ZFOR
ALY	IN	QBN	TL	VL	_&F
ALZ	IREG	QSOU	TLB	WL	_&M
END	OUT	REG	UL	XFOR	_&T

IMPORTANT

The following variables (Cf. General Comments, NAMELIST /GEOM/) are used to allocate space. It is important that they are specified correctly:
They can be approximated by a value larger than actually needed but limited by the "Current Limit." The required values are printed after being computed. Any of these variables that remain unchanged for a subsequent restart run should not be respecified since they are read from the restart file.

IMAX	JMAX	KMAX
NM1	NL1	IFREB

The amount of data specified explicitly by IFREB (User-specified Rebalancing) and implicitly through the Thermal Structure Prototype Records have to share an array, whose length NAVAIL is set at compile time by a PARAMETER statement in the header routine.

IGEOM

0 Regular box geometry option. (*)

-1 Cylindrical geometry option using box geometry input.

Notes:

1. A surface must be dedicated to the origin R = 0.0 if present.
   Set KFLOW(N) = -3 and KTEMP(N) = 400 for that surface.
2. For full 2π radian geometries J = 1 is automatically linked to J = JMAX, thus no surfaces need be defined at Y = 0.0 and Y = 2π.

IMAX The maximum number of cells in the X(R)-direction. (1)

JMAX The maximum number of cells in the Y(Θ)-direction. (1)

KMAX The maximum number of cells in the Z-direction. (1)

DX(I) The calculational cell sizes along the X-axis, **m**.

DY(J) The calculational cell sizes along the Y-axis, **m** or **rad**.

DZ(K) The calculational cell sizes along the Z-axis, **m**.

NL1 Total number of surface elements. (0)

NM1 Total number of computational cells. (0)

Note:
Both NL1 and NM1 can be approximated by values larger than actually required. However, if this is done they must not be included in NAMELIST /GEOM/ when restarting (ISTATE > 0).
The required values are printed when computed. To specify NM1 and/or NL1, makes sense only at the start of a steady-state run (ISTATE = 0).

NSURF The number of unique surfaces enclosing the calculational area. Unique surfaces are determined by a unique combination of the following three characteristics:

1. Velocity Boundary Condition
2. Temperature Boundary Condition
3. The unit normal vector to the surface.

The unit normal vectors referred to by the following three variables are those pointing into the calculational area.

XNORML(N) The X-component of the unit normal vector to surface N.

YNORML(N) The Y-component of the unit normal vector to surface N.

ZNORML(N) The Z-component of the unit normal vector to surface N.

ITURKE In this version of FLUTAN four turbulence models are included. For all of the details of input requirements for these options see the Turbulence Models in NAMELIST /TURB/.

0 Constant turbulent viscosity and conductivity model (*).

10 Zero-equation turbulence model.

11 One-equation turbulence model.

12 Two-equation turbulence model.

NFORCE Number of force structures. (0)

The input defining the force structures, i.e., the Force Structure section of NAMELIST /DATA/ and the FORCE STRUCTURE SPECIFICATION RECORDs, must be included at the start of a steady-state run (ISTATE = 0) when NFORCE > 0.

These two sections can be completely redefined at the beginning of a transient run (ISTATE = 2) by setting NEWFOR = 1 in NAMELIST /DATA/.

ISTRUC *0* No thermal structures are used. (*)

Do not include THERMAL STRUCTURE PROTOTYPE RECORDs or THERMAL STRUCTURE LOCATION RECORDs in the input.

1 The input defining the thermal structures, i.e.

THERMAL STRUCTURE PROTOTYPE RECORDs and THERMAL STRUCTURE LOCATION RECORDs must be included at the start of a steady-state run (ISTATE = 0) when ISTRUC = 1.

These two sections can be completely redefined at the beginning of a transient (ISTATE = 2) by setting NEWTS = 1 in NAMELIST /DATA/.

IPRES0 I-index of the pressure reference point.

JPRES0 J-index of the pressure reference point.

KPRES0 K-index of the pressure reference point.

Restart Option (Use of Unit 9 & 10)

There are two ways to force the code to write a restart file. The first is to run the job to maximum CPU-time.

This is done by specifying large values for NTMAX and TIMAX. The amount of time remaining for the job is checked at the end of each iteration using the system routine TREMAN (See the appendix section entitled MACHINE DEPENDENT ROUTINES.) If the amount of time remaining is greater than TREST, an input parameter in NAMELIST /DATA/, another iteration is performed. If not, depending on IFRES, a restart file is written.

The second way to obtain a restart file is to set NTMAX or TIMAX to a time step or time which will be reached before the CPU job time expires. A restart file will be written at this time step or time. Thereafter execution terminates.

When restarting from a previous run, make sure that ISTATE is set to the appropriate value. Also, it is advisable to delete all input for variables that one does not intend to change. In some cases variables will be reset back to their initial values if the input specification remains in the input stream. In short, the minimum input necessary is the correct input for restart cases.

IFRES *0* New case with no restart data written. (*)

1 New case with restart data written to Unit 10.

2 Restart of previous run, restart data read from Unit 9 with no new restart file written.

3 Restart of previous run, restart data read from Unit 9 with new restart file written to Unit 10.

Solution Method Control

ISYMCH

1 No solution of momentum equation; else same as 2.

2 SOR-Method for all equations.

3 CRESOR-Method for pressure equation; else same as 2. (*)
CRESOR is a combination of a 2-step adaptive red/black SOR technique with the method of Conjugate Residuals. Global convergence is accelerated by coarse mesh rebalancing, i.e. direct determination of pressure increment corrections, so as to yield mass balance within the coarse mesh.(Cf.Ref./6/)

4 ADI-Method for pressure equation; else same as 2.
(Alternating Direction Implicit; implementation suspended).

5 Direct Solution Method for all equations.
The direct method imposes strong limitation on NM1 (actually 5000).

Rebalancing Option

IFREB

0 No user-specified-region rebalancing. (*)

> 0 Rebalancing is performed over user defined rebalancing regions. The value of IFREB is used to allocate dynamic storage and must be at least as large as the total number of cells in the rebalancing regions plus the total number of cells used to specify rebalancing surfaces. A value of 2 × NM1 should be adequate space for most cases. The exact value needed will be printed in the Rebalancing Summary.

The input defining the rebalancing regions i.e., the Rebalancing Option section of NAMELIST /REBAL/ and the REBALANCING REGION RECORDs must be included at the start of a steady state run (ISTATE = 0) when IFREB > 0.

These two sections can be completely redefined at the beginning of a transient run (ISTATE = 2) by setting NEWREB = 1 in NAMELIST /DATA/.

Note: In addition to rebalancing over user specified regions, plane-by-plane rebalancing is available and controlled by the variables IXREB, IYREB, and IZREB in the Rebalancing Option section of NAMELIST /DATA/. In this case , however, IFREB = 0 is required.

Linkage of User Provided Heat Exchanger Model Package

A flag is intended to control inclusion of user provided routines for special heat exchanger model:

IHXMOD

0 No linkage of special heat exchanger model package (*)

1 Linkage of special heat exchanger model package (cf. KTEMP)

BOUNDARY SURFACE IDENTIFICATION RECORDS

This set of records must follow NAMELIST /GEOM/ and be present at the start of steady-state runs (ISTATE = 0).

The purpose of these records is to specify sets of cells forming boundary surfaces which completely enclose the calculational region and to define any other boundary surfaces inside the calculational region. These interior boundary surfaces must completely surround a surface, a cell, or a group of cells. To completely surround a surface one must specify two boundary surfaces with normals in opposite directions. A single-sided boundary surface is not allowed in the interior of the calculational region. Also be sure that all surfaces specified bound calculational cells. Each boundary surface is defined by specifying one or more BOUNDARY SURFACE IDENTIFICATION RECORDs, each of which contains the following variables in FORMAT (A4,E10.3,7I4):

KEY AREA IB IE JB JE KB KE N

KEY

REG The surface is a regular surface.

Regular surfaces lie on grid planes.

IREG The surface is an irregular surface.

Irregular surfaces do not lie on grid planes.

END A record with 'END' in columns 1-4 must terminate the BOUNDARY SURFACE IDENTIFICATION RECORDs.

AREA

<0.0 or blank The area of each surface element is set to its actual geometrical value, e.g. for cartesian geometry either DX × DY, DY × DZ, or DX × DZ, whichever is appropriate.

≥0.0 The area of each surface element is assigned a value of AREA, m^2.

IB,JB,KB
IE,JE,KE

These six variables are the beginning and ending I-, J-, and K-indices that define a rectangular solid composed of one or more cells. The rectangular solid that defines or partially defines a surface is the one adjacent to and on the side pointed to by the surface normal. (Keep in mind that the surface normals XNORML, YNORML, and ZNORML always point into the calculational region.) The intersection of each cell and the surface defines a surface element.

N

The surface number. All surfaces with the same combination of the following three characteristics can be assigned the same surface number:

1. Velocity boundary condition,
2. Temperature boundary condition,
3. Unit normal vector to the surface.

Notes:

1. It is possible for two surface elements to lie in the same surface and have either the same or different surface numbers as well as for two surface elements to lie in different surfaces and have the same or different surface numbers.
2. The order of the BOUNDARY SURFACE IDENTIFICATION RECORDs must be as follows:
    - All IREG records (irregular surfaces) must precede all REG records (regular surfaces).
    - The surface numbers, N, of all IREG - and REG records must start from 1 and form a series with increment 1.
3. When using cylindrical geometry (IGEOM = -1), a surface must be specified at the origin when calculational cells are bounded by the origin. When an annular region is being modeled, a surface should not be defined at the origin but rather at the boundary of the first (counting from the center) calculational cell. Set KFLOW(N) = -3 and KTEMP(N) = 400 for surfaces defined at the origin.
4. When using cylindrical geometry (IGEOM = -1), with 2π radian geometries, J = 1 and J = JMAX are automatically linked, thus, no surfaces need be defined at Y = 0.0 and Y = 2π.

ALPHA *0.0* Semi-implicit time advancement for the equations for both, turbulent kinetic energy and turbulent kinetic energy dissipation.

1.0 Fully-implicit time advancement for the equations for both, turbulent kinetic energy and turbulent kinetic energy dissipation. (*)

FCTLO See FCTHI

FCTHI To allow the user to spot check property values a small table is printed with nine temperature values ranging from FCTLO to FCTHI at a pressure PRES0. When the sodium package is present, the default values of FCTLO and FCTHI are 300.0 and 700.0 °C. When the water package is present, the default values are 20.0 and 100.0 °C.

IFENER *0* No enthalpy calculation.

1 Enthalpy calculation is performed. (*)
Thermal diffusion D across the interface F of two neighbouring cells is computed as
$D_{i+1/2} = 2F/(\Delta x_i/d_i + \Delta x_{i+1}/d_{i+1}) \times (h_i - h_{i+1})$
h specific enthalpy, Δx cell width normal to F, d thermal diffusivity ($d = \lambda/c_p$; λ thermal conductivity, c_p specific heat).

2 Enthalpy calculation is performed.
Accounting for the cell volume porosity γ , the thermal diffusion D is computed as
$D_{i+1/2} = 2F/(\gamma_i \Delta x_i/d_i + \gamma_{i+1} \Delta x_{i+1}/d_{i+1}) \times (h_i - h_{i+1})$

IFITEN *0* Enthalpy calculation factor = 1.0×DTFUEL. (*)

1 Enthalpy calculation factor = 0.5×DTFUEL.

IDRODT Control parameter for the treatment of the compressibility term $\left(\frac{\partial \rho}{\partial t}\right)_0$ in the pressure equation ($\delta p = p - p^n$)

$$a_0 \delta p_0^{r+1} - \sum_{\beta=1}^{\beta=6} a_\beta \delta p_\beta^{r+1} = -V_{f0}\left(\frac{\partial \rho}{\partial t}\right)_0 + G^r$$

where r is an iteration index, n indicates the last time step number and G^r collects all other terms of the right hand side:

≤ 0 $\quad \simeq \frac{\rho_0^r - \rho_0^n}{\Delta t}$

1 - 9 $\quad \simeq \frac{\rho_0^{r+1} - \rho_0^n}{\Delta t} = \frac{\rho_0^{r+1} - \rho_0^r}{\Delta t} + \frac{\rho_0^r - \rho_0^n}{\Delta t}$

$$\simeq \frac{1}{\Delta t}\left(\frac{\partial \rho}{\partial p}\right)_0^r (p_0^{r+1} - p_0^r) + \frac{\rho_0^r - \rho_0^n}{\Delta t}$$

$$= \frac{1}{\Delta t}\left(\frac{\partial \rho}{\partial p}\right)_0^r (\delta p_0^{r+1} - \delta p_0^r) + \frac{\rho_0^r - \rho_0^n}{\Delta t}$$

Introducing the compressibility into the pressure equation, the term with the unknown δp_0^{r+1} ought to be brought to the left hand side, thus improving diagonal dominance.
The same applies to the next option.

$$\geq 10 \quad = \left(\frac{\partial \rho}{\partial p}\right)_h \frac{\partial p}{\partial t} + \left(\frac{\partial \rho}{\partial h}\right)_p \frac{\partial h}{\partial t}$$

$$\simeq \left(\frac{\partial \rho}{\partial p}\right)^r_h \frac{p_0^{r+1} - p_0^n}{\Delta t} + \left(\frac{\partial \rho}{\partial h}\right)^n_p \frac{h_0^{r+1} - h_0^n}{\Delta t}$$

$$= \frac{1}{\Delta t}\left(\frac{\partial \rho}{\partial p}\right)^r_h \delta p_0^{r+1} + \left(\frac{\partial \rho}{\partial h}\right)^n_p \frac{h_0^{r+1} - h_0^n}{\Delta t}$$

Note: For IDRODT = 2, 4, 6, .. the slope of the vapor pressure curve is computed from the Clausius-Clapeyron-Equation. Otherwise the analytical derivative of the semi-empirical vapor pressure is used.

The two variables below give the user some control over the frequency that the momentum and enthalpy calculations are performed. The need for this control might arise in cases when one of the two fields (velocity or enthalpy) varies slowly compared to the other. The intent is to be able to perform one of the calculations (momentum or enthalpy) every time step while performing the other only occasionally resulting in a savings of CPU time. Before the user activates these variables it is highly recommended that s/he understands the full implications of this approximation. The following combinations are allowed:

ISETMO = 1 and ISETEN = N where N is any nonzero integer.

ISETMO = N and ISETEN = 1 where N is any nonzero integer.

ISETMO = M and ISETEN = N where one of the following conditions is satisfied:

- M < 0 and N divides M or
- N < 0 and M divides N or

ISETEN *N < 0* When N is less than zero the enthalpy calculation is turned off every -Nth timestep.

N > 0 When N is greater than zero the enthalpy calculation is turned on only every -Nth timestep. (1)

ISETMO *N < 0* When N is less than zero the momentum calculation is turned off every -Nth timestep.

N > 0 When N is greater than zero the momentum calculation is turned on only every -Nth timestep. (1)

Note: At the commencement of a computation ISETMO must not be set (different from 1).

LCRES *N ≥ 0* If the CRESOR solution method is used for the pressure equation, N is the orthogonalization level for the residual increments. (2)
N is limited to 7 by the code.

NUMDIF *N ≥ 0* Cf. paragraph on numerical diffusion. (0)

FACFRE *< .5* Cf. paragraph on numerical diffusion. (0.1)

FACFRM *< .5* Cf. paragraph on numerical diffusion. (0.1)

ISTATE

0 Start of steady-state run. Geometry, boundary conditions, and initial conditions are specified from the input stream. Other parameters take default values or zero. (*)

1 Continuation of a steady-state run. Initial conditions are read from the restart file of a previous run in which steady-state has not yet been achieved. Some parameters may be changed in the input stream.

2 Beginning of a transient run. Initial conditions are read from the restart file of a previous run. It is desirable although not necessary that this previous run has achieved steady-state. Some parameters may be changed in the input stream.

3 Continuation of a transient run. Initial conditions are read from the restart file of a previous beginning-of-transient run or continuation-of-transient run. Limited changes may be made in the input stream.

The following three parameters are used when ISTATE = 2. In other cases these variables are ignored.

NEWTS

0 No new thermal structure input is read.

1 New thermal structure information is read if ISTRUC = 1 and ISTATE = 2

NEWREB

0 No new rebalancing information is read.

1 New rebalancing information is read if IFREB > 0 and ISTATE = 2.

NEWFOR

0 No new force structure information is read.

1 New force structure information is read if NFORCE > 0 and ISTATE = 2.

MODEL Two-phase model flag (cf. Ref./3/).

1 Slip-Model. A simplified Two-Phase Model with

- either a constant slip ratio with the limiting case of the Homogeneous Equilibrium Model (slip ratio = 1.0)
- or a constant relative slip velocity normalized by the mixture velocity

in each of the coordinate directions. (*)

2 Separated Phases Model (suspended).

I2PMUL Controlling the meaning of the slip parameter triple VSLIPX/Y/Z

1102 VSLIPX is constant slip ratio, i.e. U_g/U_l (*)

1103 VSLIPX is constant normalized relative slip velocity, i.e. $(U_g - U_l)/U_m$ U_m is the mass weighted velocity of the phase mixture. (with similar expressions for the Y & Z-direction)

VSLIPX Slip parameter for X-direction, (1.0)

VSLIPY Slip parameter for Y-direction, (1.0)

VSLIPZ Slip parameter for Z-direction, (1.0)

Note: The default values 1.0 (I2PMUL = 1102) refer to the Homogen. Equilibrium Model. (In the single (liquid or vapor) phase regime VSLIPX/Y/Z are not used).

Time and Time Step Related Parameters

TSTART Initial time, s. (0.0)
This value should be reset to zero at the beginning of a transient run, ISTATE = 2.

IDTIME *0* The time step size is taken from the user specified variable DT.

1 The time step size is computed internally as the product of the the largest allowable time increment given the conditions (Courant time step size) and a user specified variable, RDTIME. (*)

DT(1) Time step size for time steps 1 through LASTDT, **s.** (0.1)

This value is used only if IDTIME = 0.

DT(2) Time step size for time steps after LASTDT, **s.** (0.1)

This value is used only if IDTIME = 0.

LASTDT This variable in combination with DT allows the user to change the time step size during a run. The time step size for all time steps through LASTDT is taken from DT(1). After step number LASTDT, the time step size is taken from DT(2). (99999)
This value is used only if IDTIME = 0.

RDTIME The time step size is computed internally as the product of the largest allowable time increment following Courant and this variable, RDTIME. (0.8)
This value is used only if IDTIME = 1.

NTMAX The maximum time step number for this run. Normal termination occurs after completion of this time step. (99999)

TIMAX The maximum time of this run. Normal termination occurs after this time has been reached, **s.** (3.6E + 7)

TIMAX refers to the simulation or problem time and not the computer CPU time needed to run the problem.

TREST The amount of time remaining for the job is checked at the end of each iteration. If the amount of time remaining is greater than TREST another iteration is performed. If not, the job terminates regularly, a restart file may be written (cf. IFRES).
When running long jobs or jobs requiring several seconds per iteration, one might wish to choose a larger more conservative value of TREST, **s.** (5.0)

This implementation depends on the routine TREMAN which returns the time left until the total job time as specified on the JOB card has elapsed.

Two parameters are reserved for future code extensions:

DTENER Cf. DTWALL

DTFUEL Cf. DTWALL

Iteration Control Parameters

The general definitions and default values of control parameters are given in this section. For a diagram showing the loop to which each variable relates, see the CONTROL PARAMETERS AT A GLANCE section in the appendix.

IT(1) Number of inner iterations for time steps 1 through LASTIT. (1)

IT(2) Number of inner iterations for time steps after LASTIT. (10)
During a transient (ISTATE=2 or ISTATE=3) the number of iterations within a time step should be increased to insure that the solution is converged within each time step. This can be verified by assuring that the number of iterations, printed under the heading "IT" in the time step summary, is less than the input value of IT. Before increasing IT beyond 100, however, the user is encouraged to examine the input and results for possible improvements.

LASTIT This variable in combination with IT allows the user to change the number of iterations per time step during a run. The number of iterations for all time steps through LASTIT is taken from IT(1). After step number LASTIT, the number of iterations is taken from IT(2). (99999)

ITENMX Reserved for future code extensions.

ITMAXP Number of iterations in the pressure iteration loop. (100)

ITMAXE Number of iterations in the enthalpy iteration loop. (100)

ITMOMX Number of iterations in the "MOMI" iteration loop. (1)

ITMASX Number of iterations in the "PEQN" iteration loop. (1)

OMEGA Relaxation factor for pressure solution. (1.5)
When CRESOR, being adaptive, is used, not in effect.

OMEGAV Under-relaxation factor for the momentum equation coefficients. (0.8)

OMEGAE Under-relaxation factor for the enthalpy equation coefficients. (0.8)

RELAXE Relaxation factor for enthalpy solution. (0.95)

OMEGAM Factor for pressure increment correction term (for IDRODT $\neq$ 0):
$-\omega_m \times \Delta t \int \delta L \, d\tau \,/\, \int (\partial\rho/\partial p)\, d\tau$, δL = mass residue ($\mathbf{kgs^{-1}m^{-3}}$). (1.0)

OMEGAR Controls use of (homogeneous) initial or actual values of density.
For OMEGAR $\leq$0.0 the initial values are used. (1.0)

EPS1 Steady state convergence criterion parameter. (1.0E-4)

EPS2 Steady state convergence criterion parameter. (1.0E-6)

EPS3 Steady state convergence criterion parameter. (5.0E-5)

EPS4 Reserved for future code extensions.

EPS5 Enthalpy convergence criterion parameter. (1.0E-5)

Note: For EPS1, EPS2, EPS3, EPS5 cf. page 50.

Boundary Condition Types

All external surfaces must have a velocity boundary condition type and a temperature/heat flux boundary condition type. Internal surfaces may also be assigned boundary condition types.

KFLOW(N) Type of velocity boundary condition. (The default for all NSURF surfaces is 1)

- *-5* Continuative mass flow outlet.
- *-4* Uniform velocity outlet.
- *-3* Free slip boundary.
- *-2* Continuative velocity outlet.
- *-1* Continuative momentum outlet.
- *1* Constant velocity boundary with normal velocity set from VELOC(N) or explicitly specified by the BOUNDARY VALUE INITIALIZATION RECORDs. The tangential component is in effect zero. The presence of a solid wall (no slip boundary) must be indicated by using this type of boundary condition (KFLOW(N) = 1) with the constant velocity set to 0.0. (*)
- *100 + NF* Uniform transient velocity boundary with normal velocity set from the product of the NFth transient function and VELOC(N).

KTEMP(N) Type of temperature/heat flux boundary condition.

(The default for all NSURF surfaces is 1)

- *1* Specified constant temperature boundary with temperature set from TEMP(N) or the BOUNDARY VALUE INITIALIZATION RECORDs. (*) The surface heat flux is nominally computed considering the fluid conduction but not the presence of a wall. If one wishes to account for both the fluid convection and a wall conduction, the following four variables from the Wall Model section below must be specified: IHTWAL(N), HYDWAL(N), WALLDX(N), and MATWAL(N).
- *100 + NF* Uniform transient temperature boundary with temperature set from the product of the NFth transient function and TEMP(N). The surface heat flux is computed with the options as specified above for KTEMP(N) = 1.
- *200* Specified constant heat flux boundary with normal heat flux set from TEMP(N) or the BOUNDARY VALUE INITIALIZATION RECORDs.
- *300 + NF* Uniform transient heat flux boundary with normal heat flux set from the product of the NFth transient function and TEMP(N).
- *400* Adiabatic or zero diffusive heat flux boundary.
- *500 + NF* Duct wall temperature boundary. This boundary condition type accounts for fluid convection, thermal capacity of the wall, and the heat transfer to the surrounding atmosphere or medium. The variables in the Wall Model section below must be specified. The transient function defined by NF is a multiplier of the volumetric heat source in the wall. If a constant volumetric heat source is desired, simply specify a value of 500 for KTEMP(N).
- *600* *If IHXMOD = 1*, KTEMP = 600 refers to the special heat exchanger model package.

Note: For KFLOW(N) = -3, FLUTAN automatically assumes KTEMP(N) = 400.

KPRES(N) Type of pressure boundary condition. Pressure boundary conditions are applied to the cells adjacent and interior to the boundary surface specified. (The default for all NSURF surfaces is 0)

0 No pressure boundary condition is applied. (*)

1 Uniform constant pressure boundary with pressure set from PRES(N).

100+NF Uniform transient pressure boundary with pressure set from the product of the NFth transient function and PRES(N).

Boundary and Cell Initialization

The following three variables allow easy specification of uniform velocity, temperature/heat flux, and pressure values at boundaries at the beginning of a run (ISTATE=0). To change surface values of velocity, temperature, or pressure on subsequent restarts (ISTATE=1, 2, or 3) the BOUNDARY VALUE INITIALIZATION RECORDs must be used. Nonuniform distributions can also be specified by using the BOUNDARY VALUE INITIALIZATION RECORDs.

VELOC(N) Initial velocity at surface N in the direction indicated by XNORML(N), YNORML(N), and ZNORML(N), **ms**$^{-1}$. (0.0)

TEMP(N) According to the boundary condition type, cf. KTEMP(N):

Initial temperature for surface N, **°C**. (0.0)

Initial heat flux, **Wm**$^{-2}$. (0.0)

PRES(N) Initial pressure for surface N, **Pa**. (0.0)

THETA(N) Initial dimensionless boundary volume fraction. (1.0)

TEMP0 Initial temperature of all internal cells, **°C**. (0.0)

PRES0 Initial pressure at the pressure reference point located at (IPRES0,JPRES0,KPRES0), **Pa**. (1.01325E+5)

The initial static head pressure at any point is computed with respect to the pressure reference point.

GRAVX X-component of gravity vector, **ms**$^{-2}$. (0.0)

GRAVY Y-component of gravity vector, **ms**$^{-2}$. (0.0)

GRAVZ Z-component of gravity vector, **ms**$^{-2}$. (0.0)

Wall Model

The variables in this section are used when specifying temperature boundary condition type 1, 100 + NF, or 500 + NF.

WALLDX(N) Wall thickness, **m.** (1.0)

MATWAL(N) Material type for surface N. The value of this variable is used as the index NM in the Material Properties (Solids) section below. (1)

IHTWAL(N) Heat-transfer correlation number for the calculation of heat transfer between coolant and wall. The value of this variable is used as the index NH in the Fluid-Structure Heat Transfer section below. (0)

Note.

If the default value is taken, then the coolant to wall heat-transfer coefficient, if used, is evaluated simply as the fluid conductivity divided by the fluid conduction length.

HYDWAL(N) Hydraulic diameter (characteristic length) associated with surface N, **m.** (0.0)

The transient volumetric heat source is given by the product the following three variables and the transient function NF.

WALLQS(N) Average wall volumetric heat source, $\mathbf{Wm^{-3}}$. (0.0)

TSINK(N) Temperature of surrounding atmosphere or medium, **°C.** (0.0)

HSINK(N) Heat-transfer coefficient from wall to surrounding atmosphere or medium, $\mathbf{Wm^{-2}K^{-1}}$. (0.0)

DTWALL Time step size used for with temperature boundary condition type 500 + NF. This time step size is used only until steady-state is reached, **s.** (1.0E + 40)

Fluid-Structure Heat Transfer

Heat transfer correlations are defined by specifying coefficients to compute the Nusselt number. These coefficients and thus the heat transfer correlations are indexed by the values of IHTWAL in the Wall Model and IHT in the THERMAL STRUCTURE PROTOTYPE RECORDs. The Nusselt number (NU) is computed from the following equation:

$$NU = HEATC1(NH) + HEATC2(NH) \times RE^{HEATC3(NH)} \times PR^{HEATC4(NH)}$$

where

RE is the Reynolds number, and

PR is the Prandtl number.

NHEATC Number of heat transfer correlations. (1)

This value must be at least as large as the largest value of IHT and IHTWAL.

HEATC1(NH) Nusselt number coefficient.

Since the Nusselt number, NU, must always be positive, HEATC1(NH) should be positive to accomodate a zero-flow situation. (3.66)

HEATC2(NH) Nusselt number coefficient. (0.023)

HEATC3(NH) Nusselt number coefficient. (0.8)

HEATC4(NH) Nusselt number coefficient. (0.4)

Note: The default values are set for water, must be respecified for sodium or user-provided fluid packages.

The Nusselt number is used to specify the heat transfer cofficient (h) in the following equation:

$$h = (k/D) \times NU$$

where

k is conductivity and

D is the reference length.

h is in turn used to compute the Fluid-Structure heat transfer (q) as follows:

$$q = A \times h \times (T_s - T_f)$$

where

A is the area,

T_s is the temperature of the structure, and

T_f is the temperature of the fluid.

QK( ) Thermal Structure heat source multiplier. (1.0)
As the Thermal Structure is axially aligned, the index of the intervals along that axis is corresponding to the index of QK( ).

Material Properties of Solid Wall Structures

The following equations are used to define the thermal conductivity, specific heat, and density of materials other than the coolant.

CONDUCTIVITY	= C0K (NM) + C1K (NM) × TC +C2K (NM) × TC^2	$Wm^{-1}K^{-1}$
SPECIFIC HEAT	= C0CP(NM) + C1CP(NM) × TC +C2CP(NM) × TC^2	$Jkg^{-1}K^{-1}$
DENSITY	= C0RO(NM) + C1RO(NM) × TC +C2RO(NM) × TC^2	kgm^{-3}

where TC is the temperature in **°C** and NM is the number of the material region.

The coefficients listed below are indexed by values of MATWAL from the Wall Model section of NAMELIST /DATA/ and MI from from the THERMAL STRUCTURE PROTOTYPE RECORDs.

NMATER Number of materials. (0) This value must be at least as large as the largest value of MATWAL and MI.

C0K(NM) Conductivity coefficient. (0.0)

C1K(NM) Conductivity coefficient. (0.0)

C2K(NM) Conductivity coefficient. (0.0)

C0CP(NM) Specific heat coefficient. (0.0)

C1CP(NM) Specific heat coefficient. (0.0)

C2CP(NM) Specific heat coefficient. (0.0)

C0RO(NM) Density coefficient. (0.0)

C1RO(NM) Density coefficient. (0.0)

C2RO(NM) Density coefficient. (0.0)

Transient Functions

All transient driving functions are input into the following three variables. They must be input at the beginning of the transient (ISTATE = 2) even if they have been input previously. Each function is defined by a user specified set of points. Cubic spline fit coefficients are then generated in SUBROUTINE FITIT.

50 equally spaced values are printed to allow the user to check the adequacy of the input distribution. 10-15 values with points concentrated at rapidly changing Y-values should be adequate.

Currently the total number of points allowed for the specification of transient functions is 100.

TVAL(NP) The independent variable, usually time, for the transient functions.

FVAL(NP) The dependent variable for the transient functions. The first value of the second function immediately follows the last value of the first function. The same pattern must be followed for all subsequent functions. Make sure that the entire range of the function used lies within the range input as the fitting routine does not extrapolate. Discontinuities are indicated by specifying the same X-coordinate twice with the same or different Y-coordinate values.

NEND(NF) The number of points in the NFth transient function.

NTOTS In order to simplify thermal structure input in certain cases, the heat source transient function numbers can be overridden in NAMELIST /DATA/. These values are input into the variable NTOTS in the order in which the thermal structure prototypes were defined. Any values specified in NTOTS will override all other input and previous values. If no values of NTOTS are defined, no changes to the heat source transient function numbers are made.

NOFQT Number of the transient function which is used as a multiplier of total heat source when no thermal structures are present. (0)

Plotting Option (Use of Unit 20)

The following list -open to future extension- is showing the physical quantities and the corresponding FORTRAN arrays which selectively may be written to Unit 20 for off-line plotting purposes.

1. Velocity	UL, VL, WL, VELBN
2. Temperature	TL, TLB
3. Pressure	P, PB
4. Enthalpy	HL, HLB
5. Density	RL, RLB
6. Turbulent Kinetic Energy	TURK, TURKB
7. Turbulent Kinetic Energy Dissipation	TKED, TKEDB
8. Turbulent Viscosity	TURVIS
9. Turbulent Conductivity	TURCON

Two input arrays have to be used to specify the plotting information:

IFPLOT Actually 9 values, 1 for each of the 9 quantities listed above, are used to specify whether the corresponding arrays are to be written to Unit 20 for plotting purposes, or not:

1 The corresponding arrays are written.
(* for IFPLOT(1) and (2), i.e. temperature and velocity arrays are written by default)

0 The corresponding arrays are not written.
(* for IFPLOT(3) to (9))

NTPLOT Up to 25 values to specify when plotting information is to be written to Unit 20. The following are acceptable values of NTPLOT:

0 No more plotting information is written. (*)

> 0 Time step number for which plotting information is written to Unit 20. After the Nth positive time step in NTPLOT has been processed, the N + 1th value of NTPLOT is used to determine subsequent writes.

< 0 A value -N indicates that information is written to Unit 20 every Nth time step. No subsequent values of NTPLOT are considered.

Example.

NTPLOT = -5 indicates that every 5th step is to be processed.

NTPLOT = 5, 10, -20 indicates that steps 5, 10, 20, 40, 60, etc., are to be processed.

NTPLOT = 10, 20, 0 indicates that only steps 10 and 20 are to be processed.

Printing Option

Calls to SUBROUTINE OUTPUT are controlled by the two variables NTPRNT and TPRNT. They can be used individually or together. The information printed at each call to SUBROUTINE OUTPUT is determined by the variables ISTPR and NTHPR described below.

NTPRNT Up to 50 time step numbers at which SUBROUTINE OUTPUT is to be called. The following are acceptable values for NTPRNT:

0 No more calls to SUBROUTINE OUTPUT.
When restarting, previous specification of NTPRNT values may be overridden by specifying the desired new values followed by a zero in NTPRNT.

> 0 Time step number for which SUBROUTINE OUTPUT is to be called. After the Nth positive time step in NTPRNT has been processed the N + 1th value of NTPRNT is used to determine subsequent calls to OUTPUT.

< 0 A value -N indicates that SUBROUTINE OUTPUT is called every Nth time step. No subsequent values of NTPRNT are considered.

-9999 SUBROUTINE OUTPUT is called just before the run is terminated. (*)

Example.

NTPRNT = 0 indicates that after initialization, SUBROUTINE OUTPUT is never called.

NTPRNT = 5, 10, -9999 indicates that SUBROUTINE OUTPUT is called at steps 5, 10, and just before termination.

TPRNT Up to 50 problem time values (**s**) at which SUBROUTINE OUTPUT is to be called. The following are acceptable values of TPRNT:

0.0 No more calls to SUBROUTINE OUTPUT. (*)
When restarting, previous specification of TPRNT values may be overridden by specifying the desired new values followed by a zero in TPRNT.

> 0.0 Times at or after which SUBROUTINE OUTPUT is to be called, **s**. After the Nth positive time in TPRNT has been processed, the N + 1th value of TPRNT is used to determine subsequent calls to OUTPUT.

< 0.0 A value of -T indicates that SUBROUTINE OUTPUT is to be called at T-second intervals. If the Nth value is negative, then the N + 1th value stores the next time value at which OUTPUT is to be called. This is nominally set to zero but can be specified by the user. No subsequent values of TPRNT are considered.

Example.

TPRNT = 1.0, 5.0, -10.0 indicates that OUTPUT is to be called at or after times 1.0, 5.0, 10.0, 20.0, . . etc..

TPRNT = -5.0, 10.0 indicates that OUTPUT is to be called at times 10.0, 15.0, 20.0,. . . etc.

ISTPR Up to 50 coded values which specify the arrays to be printed in the first call to SUBROUTINE OUTPUT. (0)

NTHPR Up to 50 coded values which specify the arrays to be printed in all calls after the first call to OUTPUT.

For 'internal arrays', each value of ISTPR and NTHPR is a signed integer of the form 'SVVPLL' (5 or more digits; cf.LL) which is coded according to the following rules:

```
S     +..Only the plane specified by 'VVPLL' is printed. (*)
        A plus sign is assumed and need not be specified.
      -..All planes between the values of 'LL' on the current
        and following values of ISTPR or NTHPR are printed.

VV    01..UL      U-component of velocity.
      02..VL      V-component of velocity.
      03..WL      W-component of velocity.
      04..HL      Enthalpy.
      05..TL      Temperature.
      06..AL      Volume porosity.
      07..RL      Density.
      08..P       Static Pressure.
      09..DL      Residual mass.
      10..ALX     X-direction surface permeability.
      11..ALY     Y-direction surface permeability.
      12..ALZ     Z-direction surface permeability.
      13..        not used
      14..TURK    Turbulent kinetic energy.
      15..QSOUR   Volumetric heat source.
      16..PSTAT0  Initial Static pressure.
      17..        P-PSTAT0.
      18..THL     Dimensionless Volume Fraction.
      19..        not used
      20..TURCON  Turbulent conductivity.
      21..TURVIS  Turbulent viscosity.
      22..TKED    Dissipation of turbulent kinetic energy.

P      1..An I-plane is printed.
       2..A  J-plane is printed.
       3..A  K-plane is printed.

LL     Specific plane to be printed.
       If S is +, only one plane is indicated.
       If S is -, the couple of 'LL' values from the current and
       from the next item of ISTPR  or NTHPR is indicating the
       range of planes to be printed.
       The S-VV-P portion of the "next" item is not significant.
       Note:
       The number indicated by LL has to be specified with the
       number of digits (leading zeros, eventually) necessary
       to represent the the maximum of IMAX,JMAX,KMAX. If only
       one digit were necessary still two have to be specified.
```

For thermal structure information, each value of ISTPR and NTHPR is a signed five digit integer of the form 'S8NNNN' which is coded according to the following rules:

```
S     +..Only structure number 'NNNN' is printed. (*)
         A plus sign is assumed and need not be specified.
      -..All structure between the values of 'NNNN' in the
         current and following values of ISTPR and NTHPR are
         printed.

NNNN  Specific structure to be printed.  If S is '+', only one
      structure is indicated.  If S is '-', the 'NNNN' values
      in the current and next values of ISTPR or NTHPR indicate
      the range of structures to be printed.
```

For 'surface arrays', each value of ISTPR and NTHPR is a signed integer of the form 'S9VVLL' (5 or more digits; cf.LL) which is coded according to the following rules:

```
S     +..Only the surface number 'LL' is printed. (*)
         A plus sign is assumed and need not be specified.
      -..All surfaces between the values of 'LL' in the
         current and following values of ISTPR or NTHPR are
         printed.

VV    01..VELBN   Normal surface velocity.
      02..QBN     Normal surface heat flux.
      03..MB      Adjacent internal cell number.
      04..HLB     Surface enthalpy.
      05..TLB     Surface temperature.
      06..AREA    Surface element area.
      07..RLB     Surface density.
      08..PB      Surface pressure.
      09..        Adjacent internal cell indices.  Each value is
                  of the form 'IIJJKK' where II is the I index,
                  JJ is the J index, and KK is the K index.
      10..        Overall heat transfer coefficient from coolant
                  to wall as used in the transient duct wall
                  model (KTEMP(LL)=500).
      11..THLB    Dimensionless Boundary Volume Fraction.

LL    Specific surface to be printed.
      If S is +, only one surface is printed.
      If S is -, the couple of 'LL' values from the current and
      from the next item of ISTPR or NTHPR is indicating the
      range of surfaces to be printed.
      The S-9-VV portion of the "next" item is not significant.
      Note:
      The number indicated by LL has to be specified with the
      number of digits (leading zeros, eventually) necessary
      to represent the the maximum of IMAX,JMAX,KMAX. If only
      one digit were necessary still two have to be specified.
```

Example.

ISTPR = 6105, -10301, -10305 (or 10305 or 99905 or 05 or -05 e.g.)

NTHPR = 1105, -2301, -2305, 90101, -90501, -90505,

indicates that the first call to OUTPUT will print the I = 5 plane of volume porosity and K-planes 1 through 5 of the X-direction surface permeability. On all subsequent calls, to OUTPUT, the I = 5 plane of the U component of velocity, K-planes 1 through 5 of the V component of velocity, the boundary velocity for surface 1, and surface temperature for surfaces 1 through 5.

LMPRNT

0 No detailed information of iterative processes is printed.

1 For each iteration detailed information is printed.(*)

2 Cell number and surface number arrays are printed (useful only for debugging purposes).

Force Structures

The Force Structures parameters are required only when NFORCE of NAMELIST /GEOM/ is greater than zero.

The Force Structure is a mechanism whereby a force can be applied to the fluid across a cell face between two computational cells.
To this end the user has to establish some generic force correlation(s), as follows:

For the laminar regime:

$$FCORR = ACORRL(NC) \times RE^{BCORRL(NC)} + CCORRL(NC)$$

For the turbulent regime:

$$FCORR = ACORRT(NC) \times RE^{BCORRL(NC)} + CCORRT(NC)$$

Regime transition is accomplished automatically by calculating both values and taking the larger one.

$$RE = RL \times \sqrt{(UL^2 + VL^2 + WL^2)} \times REYLEN(NC)/VIS, \text{ and}$$

RL is the local density,

UL, VL, and WL are local velocities, and

VIS is the local viscosity.

Then the drag or resistance forces (**Pa m**$^{-1}$) will have the following components:

$$DPDX = - FORCEF(NF) \times RL \times ABS(UL) \times UL \times FCORR/CLENTH(NF)$$

$$DPDY = - FORCEF(NF) \times RL \times ABS(VL) \times VL \times FCORR/CLENTH(NF)$$

$$DPDZ = - FORCEF(NF) \times RL \times ABS(WL) \times WL \times FCORR/CLENTH(NF)$$

Note: The FORCE STRUCTURE SPECIFICATION RECORDs (cf. page 37) are used to specify the locations of the Force Structures.

FORCEF(NF)		Force coefficient for force structure NF.
REYLEN(NF)		Length used to compute the Reynolds number for force structure N, **m**.
CLENTH(NF)	*> 0.0*	The value input is used as the characteristic length in the above equation, **m**.
	< 0.0	A characteristic length computed from either DX, DY, or DZ, whichever is appropriate, is used for CLENTH(NF) in the above equation.
ICORR(NF)		The correlation type of force structure NF. The values of ICORR must be less than 50 and are used as indices of the user specified correlation variables below.
NCORR		The number of correlation types available for force structures. This value must equal or exceed the maximum value specified in ICORR but be less than 50.
ACORRL(NC) **BCORRL(NC)** **CCORRL(NC)**		Correlation coefficients when the Reynolds number above, RE, is in the laminar regime.

ACORRT(NC)
BCORRT(NC)
CCORRT(NC)

Correlation coefficients when the Reynolds number above, RE, is in the turbulent regime.

Reducing Numerical Diffusion

When the direction of the flow is highly oblique to the grid lines, numerical diffusion may be significant. Several options to reduce this numerical diffusion are currently under assessment. The default for computing the convective flux terms of the enthalpy and momentum equation is the pure-upwind differencing scheme (NUMDIF = 0).
If, however, the user feels that reducing numerical diffusion is necessary for a specific problem then he may override the default value of NUMDIF (0) by specifying NUMDIF as a number with 5 digits, which have the following meaning:

```
Digit # 1 specifies treatment of convective terms in the enthalpy
          equation.

   0   first-order upwind technique,
   1   (not used),
   2   QUICK technique,
   3   QUICK assisted by the FRAM technique,
   4   LECUSSO technique,
   5   LECUSSO assisted by the FRAM technique.

Digit # 2 (reserved for future use).

Digit # 3 specifies treatment of convective terms in the momentum
          equations.

   0   first-order upwind technique,
   1   (not used),
   2   QUICK technique,
   3   QUICK assisted by the FRAM technique,
   4   LECUSSO technique,
   5   LECUSSO assisted by the FRAM technique.

Digit # 4 specifies handling of boundary conditions for the convective
          terms in the momentum equations.

   0   first-order upwind technique,
   1   second-order interpolation,
   2   LECUSSO technique (implementation pending).

Digit # 5 specifies the way to estimate the transport velocity at the
          center plane of a momentum control volume.

   0   linear interpolation between front and back surface,
   1   hard-limited second-order interpolation between front and back
       surface,
   2   LECUSSO interpolation (implementation pending).
```

Notes:

1. Cf. Ref./7/, /8/, /9/ about the LECUSSO technique.
2. When FRAM technique is used in the enthalpy (momentum) equation the value FACFRE (FACFRM) is used for discrimination. The default value 0.1 for both cases may be overridden in NAMELIST /DATA/.

Rebalancing Option

Large scale pressure distributions such as those which exist in an initial static state or which occur during overall velocity transients are most effectively addressed with the coarse mesh rebalancing scheme. This rebalancing is effective in reducing the number of iterations required to achieve convergence of the pressure equation.

Rebalancing has been implemented in two different modes which can be only applied separately. Plane-by-plane rebalancing in the X-, Y-, or Z-direction can be applied simply by specifying the appropriate values for IXREB, IYREB, and IZREB. Only one plane-by-plane rebalancing option can be specified.

IXREB	*0*	No X-direction plane-by-plane rebalancing. (*)
	1	Plane-by-plane rebalancing in the X-direction is performed.
IYREB	*0*	No Y-direction plane-by-plane rebalancing. (*)
	1	Plane-by-plane rebalancing in the Y-direction is performed.
IZREB	*0*	No Z-direction plane-by-plane rebalancing. (*)
	1	Plane-by-plane rebalancing in the Z-direction is performed.

In each rebalancing region the pressure is adjusted uniformly in such a way as to force the net mass conservation.

The frequency at which rebalancing occurs is specified by the following variable.

IREBIT	*N>0*	Rebalancing is performed before every N'th iteration.(10)

If user-defined-region rebalancing is desired, IFREB in NAMELIST /GEOM/ must be assigned an appropriate positive value and NAMELIST /REBAL/ must be supplied plus -optional- REBALANCING REGION RECORDs.

One approach to choosing rebalancing regions is to exclude all cells adjacent to exits and then group the remaining cells into as many rebalancing regions as possible. Another guideline is to put rebalancing surfaces between regions of grossly different pressures.

TURBULENCE MODELING

For Turbulence Modelling in FLUTAN cf. Ref./1/, /2/, /4/, /5/, /10/ !

In all of the following turbulence models an effective viscosity is used in the diffusion term of the momentum equation. This effective viscosity is the sum of the turbulent viscosity and the molecular viscosity. Similarly an effective thermal conductivity is used in the diffusion term of the enthalpy equation which is likewise the sum of the turbulent thermal conductivity and the molecular thermal conductivity.

Constant Turbulent Diffusivity Model (ITURKE=0)

The turbulent viscosity and/or turbulent conductivity are assumed to have some constant value (≥0.0) everywhere.

For this option the following input must be specified:

TURBC	Turbulent conductivity, **$Wm^{-1}K^{-1}$**. (0.0)
TURBV	Turbulent viscosity, **Pa s.** (0.0)

Zero-Equation Turbulence Model (ITURKE=10)

This option does not solve any governing equations involving turbulent quantities.

The turbulent viscosity (TURVIS) is computed from the following equation:

$$\mu_t = \rho \times l_m^2 \times \sqrt{\sum (\partial U_i/\partial x_j \, (\partial U_i/\partial x_j + \partial U_j/\partial x_i))}$$

U_i	average velocity in the direction of the i'th coordinate
l_m	Prandtl mixing length scale ($\chi \times y_w$),
χ	von Karman constant (AKAPPA)
y_w	distance to the nearest wall determined by the code (a cutoff value of 0.175 ×HYDIN is used)

According to the definition of the turbulent Prandtl Number (PRNDLH) for thermal energy transfer

$$Pr_t = \nu_t / \Gamma_t \quad (\Gamma_t = \lambda_t / \rho c_p;\ \text{turbulent thermal diffusivity})$$

the turbulent conductivity is computed from the following equation:

$$\lambda_t = c_p \times \mu_t / Pr_t$$

Accounting for buoyancy effects, both ν_t and Γ_t are amplified by a factor introducing the Richardson number Ri determined by the code

$$\nu_t = \nu_t^0 (1 + \beta_\nu \times Ri)^{\alpha_\nu}$$
$$\Gamma_t = \Gamma_t^0 (1 + \beta_\gamma \times Ri)^{\alpha_\gamma}$$

For this option the following input must be specified:

Note: The user is advised to override the default value for HYDIN.

AKAPPA	von Karman constant. (0.4)
HYDIN	Hydraulic diameter, **m**. (X(IMAX))
OMEGAT	Relaxation factor for turbulent viscosity. (0.7)
ALFAN	Coefficient for turbulent kinematic viscosity, viz. α_ν . (-0.5)
BETAN	Coefficient for turbulent kinematic viscosity, viz. β_ν . (10.0)
ALFAG	Coefficient for turbulent thermal diffusivity, viz. α_γ . (-1.5)
BETAG	Coefficient for turbulent thermal diffusivity, viz. β_γ . (3.33)
PRNDLH	Turbulent Prandtl number for thermal energy transfer, viz. Pr_t for $Ri = 0$: ν_t^0 / Γ_t^0 . (0.9)

One-Equation Turbulence Model (ITURKE=11)

The equation for turbulent kinetic energy (TURK) is solved:

$$\frac{\partial(\rho k)}{\partial t} + \sum_i \frac{\partial(\rho k U_i)}{\partial x_i} = \sum_i \frac{\partial(\mu^k \partial k/\partial x_i)}{\partial x_i} + S_k$$

k	turbulent kinetic energy $\mathbf{m^2 s^{-2}}$
U_i	average velocity in the direction of the i'th coordinate
$\mu^k = \mu_l + \mu_t / \sigma_g$	effective (laminar plus turbulent) diffusivity
σ_g	Prandtl number for turbulent kinetic energy (PRNDLK)
$S_k = P_k + S_g - \rho\varepsilon$	source of turbulent kinetic energy made up of the - production due to the main stream or alternately due to wall effects - production or dissipation due to buoyancy - dissipation through the fluid viscosity (sink)

The turbulent viscosity (TURVIS) is computed using the following equation:

$$\mu_t = \frac{c_\mu \rho k^2}{\varepsilon}$$

c_μ	coefficient for computation of turbulent viscosity (CDTURB)
ρ	local density,
k	local turbulent kinetic energy
ε	dissipation rate of turbulent kinetic energy computed from the following equation:

$$\varepsilon = \frac{c_\mu^{3/4} k^{3/2}}{\chi y_w}$$

χ	von Karman constant (AKAPPA)
y_w	distance to the nearest wall (a cutoff value of 0.175 x HYDIN is used in the code).

Wall function corrections are applied to cells adjacent to solid walls for both the turbulent kinetic energy equation and the momentum equations.
The turbulent shear stress in the turbulent zone next to the viscous sublayer is computed from the following equation:

$$\tau^t = \frac{\chi \rho c_\mu^{1/4} U_p k^{1/2}}{\ln\left(\frac{E \rho c_\mu^{1/4} y_w k^{1/2}}{\mu_t}\right)}$$

k	turbulent kinetic energy $\mathbf{m^2 s^{-2}}$
ρ	density
U_p	velocity component parallel to the wall
y_w	wall distance
E	wall roughness (EE)
χ	von Karman constant (AKAPPA)
μ_t	turbulent viscosity
c_μ	coefficient for computation of turbulent viscosity (CDTURB)

For this option the following input must be specified:

Note: The user is advised to override the default values HYDIN and TDIN.

AKAPPA von Karman constant. (0.4)

CDTURB Coefficient for computation of turbulent viscosity. (0.09)

EE Wall roughness coefficient. (9.0)

EPS6 Convergence criterion parameter for the equation for turbulent kinetic energy. (1.0E-5)

HYDIN Hydraulic diameter, **m**. (X(IMAX))

ITMAXK Maximum number of iterations for turbulent kinetic energy equations. (50)

ITURMX Reserved for future code extensions. (1)

OMEGAK Relaxation factor for the turbulent kinetic energy equation coefficients. (0.7)

OMEGAT Relaxation factor for turbulent viscosity. (0.7)

PRNDLH Turbulent Prandtl number for thermal energy transfer. (0.9)

PRNDLK Turbulent Prandtl number for turbulent kinetic energy. (1.0)

RELAXK Relaxation factor for turbulent kinetic energy solution. (0.8)

TDIN Coefficient to compute inlet turbulent kinetic energy dissipation rate, m^{-1}. (2000.)

$TKEDIL = TDIN \times TURK^{1.5}$

TURK is the inlet turbulent kinetic energy and
TDIN can be either determined empirically or by using the following equation:

$$TDIN = CDTURB/(0.04 \times HYDIN)$$

TKIN Coefficient to compute inlet turbulent kinetic energy. (0.001)

$TKINIL = TKIN \times VELBN^2$

VELBN is the inlet velocity

Two-Equation Turbulence Model (ITURKE=12)

This is the most rigorous turbulence model. Both the equation for turbulent kinetic energy (TURK, cf. One-Equation Model) and the equation for the dissipation rate of turbulent kinetic energy (TKED) are solved:

$$\frac{\partial(\rho\varepsilon)}{\partial t} + \sum_i \frac{\partial(\rho\varepsilon U_i)}{\partial x_i} = \sum_i \frac{\partial(\mu^\varepsilon \partial\varepsilon/\partial x_i)}{\partial x_i} + S_\varepsilon$$

ε	turbulent kinetic energy dissipation rate $\mathbf{m^2 s^{-3}}$
U_i	average velocity in the direction of the i'th coordinate
$\mu^\varepsilon = \mu_l + \mu_t / \sigma_\varepsilon$	effective (laminar plus turbulent) diffusivity
σ_ε	Prandtl number for turbulent kinetic energy dissipation (PRNDLD)

$$S_\varepsilon = c_{1\varepsilon}(\varepsilon/k)(P_k + S_g)(1 + c_{3\varepsilon}R_f) - c_{2\varepsilon}\rho\varepsilon^2/k$$

source of turbulent kinetic energy dissipation
- with P_k und S_g being the terms from the tur. kin. energy equation
- R_f the Richardson number, viz. $-S_g/P_k$.

The turbulent viscosity (TURVIS) is computed using the following equation:

$$\mu_t = \frac{c_\mu \rho k^2}{\varepsilon}$$

c_μ	coefficient for computation of turbulent viscosity (CDTURB)
ρ	local density,
k	local turbulent kinetic energy
ε	dissipation rate of turbulent kinetic energy computed from the differential equation above.

Wall function corrections are applied to cells adjacent to solid walls for both the turbulent kinetic energy equation and the momentum equations.
The turbulent shear stress in the turbulent zone next to the viscous sublayer is computed from the following equation:

$$\tau^t = \frac{\chi \rho c_\mu^{1/4} U_p k^{1/2}}{\ln\left(\frac{E \rho c_\mu^{1/4} y_w k^{1/2}}{\mu_t}\right)}$$

k	turbulent kinetic energy $\mathbf{m^2 s^{-2}}$
ρ	density
U_p	velocity component parallel to the wall
y_w	wall distance
E	wall roughness (EE)
χ	von Karman constant (AKAPPA)
μ_t	turbulent viscosity
c_μ	coefficient for computation of turbulent viscosity (CDTURB)

For this option the following input must be specified:

Note: The user is advised to override the default values HYDIN and TDIN.

AKAPPA von Karman constant. (0.4)

CDTURB Coefficient for computation of shear stress near the wall. (0.09)

CT1 Empirical constant used in the equation to compute turbulent kinetic energy, viz. $c_{1\varepsilon}$. (1.47)

CT2 Empirical constant used in the equation to compute the dissipation rate of turbulent kinetic energy, viz. $c_{2\varepsilon}$. (1.92)

CT3 Empirical constant used in the equation to compute turbulent kinetic energy, viz. $c_{3\varepsilon}$. (0.8)

EE Wall roughness coefficient. (9.0)

EPS6 Convergence criterion parameter for the equations of both, the turbulent kinetic energy and the dissipation rate of turbulent kinetic energy. (1.0E-5)

HYDIN Hydraulic diameter, **m**. (X(IMAX))

ITMAXK Maximum number of iterations for turbulent kinetic energy. equations. (50)

ITURMX Reserved for future code extensions. (1)

OMEGAD Relaxation factor for equation to compute the dissipation rate of turbulent kinetic energy. (0.7)

OMEGAK Relaxation factor for equation to compute the turbulent kinetic energy. (0.7)

OMEGAT Relaxation factor for turbulent viscosity. (0.7)

PRNDLD Turbulent Prandtl number for the dissipation rate of turbulent kinetic energy. (1.3)

PRNDLH Turbulent Prandtl number for thermal energy transfer. (0.9)

PRNDLK Turbulent Prandtl number for turbulent kinetic energy. (1.0)

RELAXK Relaxation factor for turbulent kinetic energy solution. (0.8)

TDIN Coefficient to compute inlet turbulent kinetic energy dissipation rate, $\mathbf{m}^{-1}$. (2000.)

$$TKEDIL = TDIN \times TURK^{1.5}$$

TURK is the inlet turbulent kinetic energy and
TDIN can be either determined empirically or by using the following equation:

$$TDIN = CDTURB/(0.04 \times HYDIN)$$

TKIN Coefficient to compute inlet turbulent kinetic energy. (0.001)

$$TKINIL = TKIN \times VELBN^2$$

VELBN is the inlet velocity

This set of records - to be completed with an END record - must be included only when NFORCE > 0 in NAMELIST /GEOM/ (cf. ISTATE, NEWFOR in NAMELIST /DATA/).

These records are used to locate the force structures described in the Force Structure section of NAMELIST /DATA/. These forces can be applied at cell faces between two computational cells. The locations therefore correspond to portions of grid planes. Each record in this section contains the following variables in the FORMAT (A4,7I4).

KEY N IB IE JB JE KB KE

KEY

XFOR X-direction force.

YFOR Y-direction force.

ZFOR Z-direction force.

END This record terminates the FORCE STRUCTURE SPECIFICATION RECORDs. It is only necessary when NFORCE > 0.

N Force structure number.

IB,JB,KB
IE,JE,KE These six variables are the beginning and ending I-, J-, and K-indices that define a rectangular solid composed of one or more cells defining cell faces at which the force with force structure number N is to be applied:
The cell face defined by cell (I,J,K) for an X-direction force is that one between cells (I,J,K) and (I + 1,J,K). For a Y-direction force, it is the one between cells (I,J,K) and (I,J + 1,K), and for a Z-direction force, it is the one between cells (I,J,K) and (I,J,K + 1).

This set of records - to be completed with an END record - must be included only when ISTRUC = 1 in NAMELIST /GEOM/ (cf. ISTATE, NEWTS in NAMELIST /DATA/).

A thermal structure is a collection of thermal structure elements each of which has the same characteristics as specified by a thermal structure prototype. Thermal structure prototypes are defined using TYPE, FLUID, and MATERIAL namelists with the names /T/, /F/, and /M/ respectively. The order in which these namelists are input indicates the construction of the thermal structures and must conform to the following rules:

1. A TYPE namelist must commence the definition of each thermal structure prototype.
2. If fluid interacts with surface one, a FLUID namelist must be present after the TYPE namelist (before the first MATERIAL namelist). If, in addition, fluid interacts with surface two, a FLUID namelist must also be present after the last MATERIAL namelist.
3. A gap exists after each material except the last. The gap parameters are specified in the MATERIAL namelist.
4. The initial default for all namelist variables is zero. Subsequent defaults are the values in effect after reading the previous namelist. If, for example, the geometrical type is the same for all thermal structure prototypes, IXYZ need be specified only on the first TYPE namelist.
5. The definition of thermal structure prototype N + 1 must follow the definition of thermal structure prototype N.
6. An END record must follow the last prototype definition.
   Blank records or comment records may be interspersed as desired.
7. See the warning in the THERMAL STRUCTURE LOCATION RECORDs section.

The precise definition of each record is as follows:

TYPE NAMELIST /T/

N Thermal structure prototype number. This number does not need to correspond to its index or ordinal number.

IXYZ Geometrical type or characteristic.

- *1* Rods (cylinders) with axis aligned in the I-direction.
- *2* Rods (cylinders) with axis aligned in the J-direction.
- *3* Rods (cylinders) with axis aligned in the K-direction.
- *11* Slab with the normal aligned in the I-direction.
- *12* Slab with the normal aligned in the J-direction.
- *13* Slab with the normal aligned in the K-direction.
- *101* Sphere aligned in the I-direction.
- *102* Sphere aligned in the J-direction.
- *103* Sphere aligned in the K-direction.

The alignment specification is included in the spherical option to allow the normalized axial power distribution multiplier, QK, to be operative.

NT The number of the transient function to be used as a multiplier for the heat source.

RODFR **Rods or cylindrical thermal structures:**

> 0 Number or fraction of actual rods interacting with each associated coolant cell.

< 0 The absolute value is the number or fraction of rods per unit area interacting with each associated coolant cell, m^{-2}.
The rods are perpendicular to the cell area.

Slab thermal structures:

> 0 Slab area in each associated coolant cell, m^2.

< 0 The absolute value is the slab area divided by the cell area. In the case of two-sided thermal structures this value is equivalent to a solid permeability for the structure.

Spherical thermal structures:

> 0 Number or fraction of spheres interacting with each associated coolant cell.

< 0 The absolute value is the number or fraction of spheres per unit volume interacting with each associated coolant cell, m^{-3}.

OUTR Thermal structure outer radius, **m**.
This is not used for slab type thermal structures.

FLUID NAMELIST /FI

IHT Heat transfer correlation index. This value is used as the index, NH, of the variables HEATC1, HEATC2, HEATC3 and HEATC4 described in the Fluid-Structure Heat Transfer section of NAMELIST /DATA/.

HYD Hydraulic diameter or reference length. This value is used as D, the reference length, as described in the Fluid-Structure Heat Transfer Section of NAMELIST /DATA/.

MATERIAL NAMELIST /MI

MI Material type index. This value is used as the index NM described in the Material Properties (Solids) Section of NAMELIST /DATA/.

NP Number of partitions in the material. A thermal structure temperature will be computed for each material partition.

DR Partition size, **m**.

Q Volumetric heat source for the material region, Wm^{-3}

The following gap properties must be correctly specified or defaulted only when another material follows. If a fluid follows, the gap properties are ignored.

SGAP Gap size, **m**.

HGAP Gap heat transfer coefficient, $Wm^{-2}K^{-1}$

This set of records - to be completed with an END record - must be included only when ISTRUC = 1 in NAMELIST /GEOM/ (cf. ISTATE, NEWTS in NAMELIST /DATA/).

Once the thermal structure prototypes have been defined the location of the thermal structure elements are specified by the THERMAL STRUCTURE LOCATION RECORDs. These records contain the following variables in FORMAT (A4,7I4)

LOC NUM IB IE JB JE KB KE

LOC *OUT* The cells specified interact with the outside or surface 1.

IN The cells specified interact with the inside or surface 2.

END A record containing 'END' in columns 1-4 is needed to terminate the THERMAL STRUCTURE LOCATION RECORDs.

NUM Thermal structure prototype number.

IB,JE,KB
IE,JE,KE These six variables are the beginning and ending I-, J-, and K-indices that define a rectangular (cylindrical) solid composed of one or more cells which are to interact with thermal structure NUM.

Notes:

1. A cell should not be specified twice by the indices unless the true intention is to have two occurrences of the thermal structure prototype NUM.
2. Many THERMAL STRUCTURE LOCATION RECORDs may be needed to define all the cells interacting with a given thermal structure prototype.
3. The order in which cells are specified is arbitrary except when the thermal structure prototype has fluid cells interacting with both surfaces. In this case cells are paired off in the order in which they are specified. The number of cells interacting with one surface must equal the number of cells interacting with the other surface.

WARNING!

While all thermal structure variables can be redefined upon restart, changes in the order in which the thermal structures are defined, changes in NP, or changes in the order of or the values on the THERMAL STRUCTURE LOCATION RECORDs will scramble the internally stored thermal structure temperatures.

BOUNDARY VALUE INITIALIZATION RECORDS

The purpose of this set of input records is to permit initialization of boundary values of any of the arrays listed below. Uniform temperature and velocity boundary conditions can be more easily specified using the variables TEMP and VELOC in NAMELIST /DATA/. Each record in this section contains the following variables in the FORMAT (A4,E10.3,7I4).

KEY RVAL IB IE JB JE KB KE N

KEY

HLB	Enthalpy, **Jkg**$^{-1}$
PB	Pressure, **Pa**
QBN	Heat flux, **Wm**$^{-2}$
RLB	Density, **kgm**$^{-3}$
TLB	Temperature, **°C**
VELB	Magnitude of the velocity normal to the surface in the direction indicated by XNORML(N), YNORML(N), and ZNORML(N), **ms**$^{-1}$
END	Terminating the BOUNDARY VALUE INITIALIZATION RECORDs. This record must always be included.

RVAL The value to be assigned to the variable named.

IB,JE,KB
IE,JE,KE These six variables are the beginning and ending I-, J-, and K-indices that define a rectangular solid composed of one or more cells. The rectangular solid that defines or partially defines a surface is the one which is totally interior and adjacent to, or partially interior to and intersecting that surface.
Cf. BOUNDARY SURFACE IDENTIFICATION CARDs.

N The surface number of the boundary being set.

Note: The END - record must be included even if actually there is no boundary value initialization.

The purpose of this set of input records is to permit initialization of internal cell values of any of the arrays listed below. Each record of this section contains the following variables in the FORMAT (A4,E10.3,6I4):

KEY RVAL IB IE JB JE KB KE

KEY

AL Volume porosity, the dimensionless ratio of fluid volume in a cell to total cell volume. (1.0)

ALX Surface permeability, the dimensionless ratio of free flow area to the total surface element area, between cell (I,J,K) and cell (I+1,J,K). (1.0)

ALY Surface permeability, the dimensionless ratio of free flow area to the total surface element area, between cell (I,J,K) and cell (I,J+1,K). (1.0)

ALZ Surface permeability, the dimensionless ratio of free flow area to the total surface element area, between cell (I,J,K) and cell (I,J,K+1). (1.0)

HL Enthalpy, **Jkg**$^{-1}$. (0.0)

P Pressure, **Pa** (0.0)

QSOU Volumetric heat source per computational cell volume DX(I) × DY(J) × DZ(K), **Wm**$^{-3}$. (0.0)

TL Temperature, **°C**. (0.0)

UL U-component of velocity at the surface element between cell (I,J,K) and cell (I+1,J,K), **ms**$^{-1}$. (0.0)

VL V-component of velocity at the surface element between cell (I,J,K) and cell (I,J+1,K), **ms**$^{-1}$. (0.0)

WL W-component of velocity at the surface element between cell (I,J,K) and cell (I,J,K+1), **ms**$^{-1}$. (0.0)

END Terminating the INTERNAL CELL INITIALIZATION RECORDs. This record must always be included.

RVAL The value to be assigned to the variable named.

IB,JB,KB
IE,JE,KE These six variables are the beginning and ending I-, J-, and K-indices that define a rectangular solid composed of one or more cells to become initialized (cf. Note 1.).

Notes:

1. The values of ALX, ALY, ALZ resp. UL, VL, WL apply to cell faces of the "rectangular solid" which are not elements of boundary surfaces. Surfaces lying on boundaries must be initialized using
   - BOUNDARY SURFACE IDENTIFICATION RECORDs as to the definition of non-standard surface areas
   - BOUNDARY VALUE INITIALIZATION RECORDs as to the definition of normal velocities.
2. The END - record must be included even if actually there is no internal cell initialization.

This set of records must be included only when IFREB > 0 (NAMELIST /GEOM/) (cf. ISTATE, NEWREB (NAMELIST /DATA/).

NAMELIST /REBALI

If IFREB > 0, the user must specify in (a separate, last) NAMELIST /REBAL/ the following variables:

NUSREB number of rebalancing regions specified by the user. (0) Specification is done through REBALANCING REGION RECORDs (see below). (0)

NXSEGM number of groups, into which the X-grid is segmented. (1) This involves only cells not allocated by REBALANCING REGION RECORDs.

NYSEGM same for the Y-grid.
NZSEGM same for the Z-grid.

LXSEGM(I) I = 1,...,(NYSEGM-1)
numbers of increments for segmentation of the X-grid. Must be positive. The missing value (for I = NXSEGM) is not input but evaluated from the sum (up to NXSEGM-1) and the required total IMAX.

LYSEGM(I) I = 1,...,(NYSEGM-1)
corresponding input array for the Y-direction.
LZSEGM(I) I = 1,...,(NYSEGM-1)
corresponding input array for the Z-direction.

REBALANCING REGION RECORDS (required if NUSREB > 0)

Each record contains specifications

KEY NUM IB IE JB JE KB KE in FORMAT (A4,7I4).

The meaning of items in these records is

KEY = UREB marks records with valid geometry specifications.
KEY = END marks end of specifications.
KEY = blank used to insert comments (starting in column 5).

NUM (sequential) index of user-specified rebalancing region (NUM = 1,...,NUSREB)

IB, JB, KB
IE, JE, KE specify a box of cells which is part of rebalancing region NUM. Standard conventions are followed here.

This input is checked for completeness (cf. NUSREB) and sequencing of region index NUM.

Note: The code checks that the regions specified for general, 3-dimensional rebalancing (in detail, through rebalancing region records and more globally, through NXSEGM, LXSEGM etc.) are, in fact, assemblies of cells, which are geometrically and physically connected.

MACHINE DEPENDENT ROUTINES

Only one machine dependent function is required in FLUTAN, **TREMAN.**

This function returns the CPU time left for the current run in units of seconds. This time starts at the TIME value specified on the JOB card and ends at zero when the job is terminated by the system. It is used for timing and to determine when to terminate and write a restart file.

The routine is available in ASSEMBLER for IBM computers and in FORTRAN for the KfK-VP400-EX (using system routine RTIME).

Another useful (but not required) IBM-ASSEMBLER routine CALSEQ is referenced only once in the error handling routine TRACBK. On IBM-computers and compatibles (e.g. the VP- and S-series of Fujitsu/Siemens) it is used to obtain the calling sequence of routines to the location of the detected error.

STORAGE ALLOCATION

Space for a few of the geometry dependent variables is allocated dynamically in the array S (INTS), which is defined in the header routine FLUTAN (Parameter NAVAIL). The address of such variable is computed at the beginning of each run. These addresses are then passed into called subroutines where the variables have individual names and are adequately dimensioned.

Generally, data are held in named COMMON blocks, dimensions of arrays given by the PARAMETER constants LCELL, LBOUND, IJKMAX. These parameters are held in the "Common Deck" PARAM, which is adequately handled by the source management code HISTORIAN (corresponding to CDC UPDATE and similar products).

An alternative to the use of HISTORIAN would be setting up the HISTORIAN decks as members of an IBM-Partitioned Data Set (PDS) and applying (instead of the HISTORIAN CALL-function) the FORTRAN INCLUDE-directive.

STEADY-STATE DEFINITION

Steady-state is reached when the following conditions are met:

1. DL < 1.0 where DL = (maximum cell residue)/DCONV,
   DCONV = EPS1 x (UVWMAX + EPS2), and UVWMAX is computed in SUBROUTINE GDCONV.
2. The change of the U-velocity component divided by the maximum velocity magnitude in the entire field is less than EPS3.
3. The change of the V-velocity component divided by the maximum velocity magnitude in the entire field is less than EPS3.
4. The change of the W-velocity component divided by the maximum velocity magnitude in the entire field is less than EPS3.
5. Maximum (DH/H) < EPS3 where H is the current enthalpy and DH is the change in enthalpy over two consecutive time steps.

ERROR HANDLING

Much effort has been invested to instrument all routines with self explaining error messages, giving name of routine, error type and information usable to locate the error in source text.
After any error is detected, the routine TRACBK is called with a non-zero argument to terminate the job.

SPECIAL TOOLS

Preparing and debugging input for FLUTAN may become a cumbersome procedure, especially for large systems. In general they require both a voluminous description with increased probability for errors, and an extensive working storage causing low job priority.

In order to improve upon these circumstances both

- a couple of generators for geometrical input
  - FUNGEO (for cartesian and cylindrical coordinates, cf. Ref./11/)
  - HEXPIN (for Rod Bundle Geometry, cf. Ref./12/)
- a general input checker

  COMCHK (cf. Ref./13/)

have been developed.
Different from the procedure followed in FLUTAN, the input checker COMCHK, while sampling incident errors, carries on checking as long as possible and reasonable.
Storage requirements are reduced to about 10 percent as compared to FLUTAN.
Besides the capability to identify multiple input errors, COMCHK offers additional means for input verification.

FLUTAN NAMELIST SUMMARY

Namelist-groups of FLUTAN code
(alphabetical order by columns)

(1) Namelist &GEOM :

DX	IFRES	IPRES0	JMAX	NFORCE	XNORML
DY	IGEOM	ISTRUC	JPRES0	NL1	YNORML
DZ	IHXMOD	ISYMCH	KMAX	NM1	ZNORML
IFREB	IMAX	ITURKE	KPRES0	NSURF	

(2) Namelist &DATA :

ACORRL	C2RO	GRAVX	IREBIT	KPRES	NTHPR	REYLEN
ACORRT	DT	GRAVY	ISETEN	KTEMP	NTMAX	TEMP
ALPHA	DTENER	GRAVZ	ISETMO	LASTDT	NTOTS	TEMP0
BCORRL	DTFUEL	HEATC1	ISTATE	LASTIT	NTPLOT	THETA
BCORRT	DTWALL	HEATC2	ISTPR	LCRES	NTPRNT	TIMAX
CCORRL	EPS1	HEATC3	IT	LMPRNT	NUMDIF	TPRNT
CCORRT	EPS2	HEATC4	ITENMX	MATWAL	OMEGA	TREST
CLENTH	EPS3	HSINK	ITMASX	MODEL	OMEGAE	TSINK
C0CP	EPS4	HYDWAL	ITMAXE	NCORR	OMEGAM	TSTART
C0K	EPS5	ICORR	ITMAXP	NEND	OMEGAR	TVAL
C0RO	FACFRE	IDRODT	ITMOMX	NEWFOR	OMEGAV	VELOC
C1CP	FACFRM	IDTIME	IXREB	NEWREB	PRES	VSLIPX
C1K	FCTHI	IFENER	IYREB	NEWTS	PRES0	VSLIPY
C1RO	FCTLO	IFITEN	IZREB	NHEATC	QK	VSLIPZ
C2CP	FORCEF	IFPLOT	I2PMUL	NMATER	RDTIME	WALLDX
C2K	FVAL	IHTWAL	KFLOW	NOFQT	RELAXE	WALLQS

(3) Namelist &TURB :

AKAPPA	BETAN	CT3	ITMAXK	OMEGAT	RELAXK	TURBV
ALFAG	CDTURB	EE	ITURMX	PRNDLD	TDIN	
ALFAN	CT1	EPS6	OMEGAD	PRNDLH	TKIN	
BETAG	CT2	HYDIN	OMEGAK	PRNDLK	TURBC	

(4) For thermal structures:

(4a) Namelist &T :

IXYZ	N	NT	OUTR	RODFR

(4b) Namelist &F :

HYD	IHT

(4c) Namelist &M :

DR	HGAP	MI	NP	Q	SGAP

(5) Namelist &REBAL :

LXSEGM	LYSEGM	LZSEGM	NUSREB	NXSEGM	NYSEGM	NZSEGM

CROSS-REFERENCE LIST OF FLUTAN ROUTINES

The name of the main (header) routine is FLUTAN.

Calling Routine		Called Routine(s)									
FLUTAN	3	AMAIN	ERRSET	TRACBK							
AMAIN	5	BMAIN	DATIME	TRACBK	TSCAN	ZEROAL					
BARIN	3	RSET2	RSET3	TRACBK							
BCFLOT	1	GETF									
BCFLOW	1	TRACBK									
BCHECK	1	TRACBK									
BCPRES	1	GETF									
BCTEMP	2	HLIQ	ROLIQ								
BCTEMT	1	GETF									
BCTEM0	2	HLIQ	ROLIQ								
BMAIN	20	BCPRHO	FLOWPR	GDCONV	GEOM3D	HEMOD	HSTRUC	HXMODB	INITAL	OUTPUT	PHYHEM
		PLTAPE	QSTRUC	RESTAR	TIMSTP	TIMSTS	TLFIX	TRACBK	TREMAN	WATSTP	WATTIM
BOXES	3	FILLM	RBSORT	TRACBK							
CRESOR	6	EXTREM	REBAZ	REBAZG	RESLOW	SELECT	SOLVPR				
DATIME	2	DATE	TIME								
DIRAPR	2	SOLVPR	TRACBK								
DIREN3	3	DECBND	DIRAPR	RESBND							
DIRPR3	3	DECBND	DIRAPR	RESBND							
DSTMSH	1	TRACBK									
ENERGI	6	ENTHAC	ENTHPR	ENTSLP	LECE00	LECE10	UPWIND				
ENLOOP	12	BCTEMP	BCTEMT	BCTEM0	DIREN3	ENERGI	EXTREM	GETF	HEMOD	HSTRUC	PHYHEM
		QSTRUC	SOLVEN								
FILLM	2	TRACBK	TREMAN								
FITIT	1	NCSPLI									
FLOWPR	1	TRACBK									
FORCES	1	TRACBK									
GDCONV	1	EXTREM									
GEOBAS	1	TRACBK									
GEOM3D	5	BOXES	IARRAY	MARRAY	RARRAY	TRACBK					
GETCPV	2	CPLIQ	VOLEXP								
GETDL	2	EXTREM	SOLVPR								
GETF	1	TRACBK									
GETRHO	1	ROLIQ									
HEMOD	1	TRACBK									
HSLIQ	1	TRACBK									
HSVAP	1	TRACBK									
HXMODA	1	TRACBK									
HXMODB	1	TRACBK									
ICTEMP	2	GETRHO	HLIQ								
INFORC	1	TRACBK									
INITAL	27	BARIN	BCFLOT	BCFLOW	BCHECK	BCPRES	BCTEMP	BCTEMT	BCTEM0	FITIT	FLOWPR
		GEOBAS	GEOEXT	GETCPV	GETF	GETRHO	HLIQ	HXMODA	ICTEMP	INFORC	INPSTR
		IREBAL	NPROPS	RESTAR	THCLIQ	TRACBK	VELADJ	VISLIQ			
INPSTR	1	TRACBK									
IREBAL	4	REBCHK	REBGEO	REBLST	TRACBK						
MOLOOP	17	BCFLOT	BCPRHO	CRESOR	DIRPR3	EXTREM	GETDL	HEMOD	MOMENI	PEQN	PEQUAL
		PHYHEM	PREADI	RMOMI	SOLVIT	XMOMI	YMOMI	ZMOMI			
MOMENI	3	BCFLOW	MAXDEV	VELADJ							
NCSPLI	1	TRACBK									
OUTPUT	5	GETF	ISURFO	PSTRUC	RARRAY	RSURFO					
PEQN	1	TRACBK									
PEQUAL	1	GETDL									
PHYHEM	5	HSLIQ	HSVAP	TSLIQ	TSMIX	TSVAP					
PREADI	1	TRACBK									
QSTRUC	2	GETF	TRACBK								
RBSORT	2	TRACBK	VISORT								
REBAZ	1	TDMA									
REBAZG	3	REBMAT	RESBND	TRACBK							
REBCHK	1	TRACBK									
REBGEO	2	CONMSH	DSTMSH								
REBLST	1	TRACBK									
REBMAT	1	DECBND									
RESTAR	2	DATIME	PLTAPE								
RMOMI	18	FORCES	LECM00	LECM10	LECM11	LECM30	LECM31	LECM32	LECM33	LECY20	LECY21
		LECY40	LECY41	LECY42	LECY43	RMOMRC	UPWIND	WLFNCV	XMOSLP		
SELECT	2	SOLVPR	TRACBK								
SOLVEN	2	EXTREM	SOLVPR								
SOLVIT	4	EXTREM	REBAZ	REBAZG	SOLVPR						
TDNER	3	ENTHAC	FHCALC	FLOWPR							

```
TIMSTP  7   BCPRES   ENLOOP   MOLOOP   TREMAN   TUR0EQ   TUR1EQ   TUR2EQ
TIMSTS  1   TRACBK
TKENER  4   ENTHAC   EXTREM   FHCALC   FLOWPR
TLFIX   3   EXTREM   FLOWPR   TRACBK
TRACBK  1   CALSEQ
TREMAN  1   RTIME
TSCAN   1   TRACBK
TSLIQ   1   TRACBK
TSMIX   1   TRACBK
TSVAP   1   TRACBK
TUR0EQ  1   TKSORC
TUR1EQ  5   BCTURB   DIREN3   SOLVEN   TKENER   TKSORC
TUR2EQ  6   BCTURB   DIREN3   SOLVEN   TDNER    TKENER   TKSORC
UPWIND  1   TRACBK
VISORT  1   TRACBK
WLFNCV  1   TRACBK
XMOMI  17   FORCES   LECM00   LECM10   LECM11   LECM20   LECM21   LECM30   LECM31   LECM32   LECM33
            LECM40   LECM41   LECM42   LECM43   UPWIND   WLFNCV   XMOSLP
YMOMI  18   FORCES   LECM00   LECM10   LECM11   LECM20   LECM21   LECM30   LECM31   LECM32   LECM33
            LECM40   LECM41   LECM42   LECM43   UPWIND   WLFNCV   YMOMRC   YMOSLP
ZMOMI  17   FORCES   LECM00   LECM10   LECM11   LECM20   LECM21   LECM30   LECM31   LECM32   LECM33
            LECM40   LECM41   LECM42   LECM43   UPWIND   WLFNCV   ZMOSLP
```

```
Called          Calling
Routine:        Routine(s):

AMAIN       1   FLUTAN
BARIN       1   INITAL
BCFLOT      2   INITAL   MOLOOP
BCFLOW      2   INITAL   MOMENI
BCHECK      1   INITAL
BCPRES      2   INITAL   TIMSTP
BCPRHO      2   BMAIN    MOLOOP
BCTEMP      2   ENLOOP   INITAL
BCTEMT      2   ENLOOP   INITAL
BCTEM0      2   ENLOOP   INITAL
BCTURB      2   TUR1EQ   TUR2EQ
BMAIN       1   AMAIN
BOXES       1   GEOM3D
CALSEQ      1   TRACBK
CONMSH      1   REBGEO
CPLIQ       1   GETCPV
CRESOR      1   MOLOOP
DATE        1   DATIME
DATIME      2   AMAIN    RESTAR
DECBND      3   DIREN3   DIRPR3   REBMAT
DIRAPR      2   DIREN3   DIRPR3
DIREN3      3   ENLOOP   TUR1EQ   TUR2EQ
DIRPR3      1   MOLOOP
DSTMSH      1   REBGEO
ENERGI      1   ENLOOP
ENLOOP      1   TIMSTP
ENTHAC      3   ENERGI   TDNER    TKENER
ENTHPR      1   ENERGI
ENTSLP      1   ENERGI
ERRSET      1   FLUTAN
EXTREM      9   CRESOR   ENLOOP   GDCONV   GETDL    MOLOOP   SOLVEN   SOLVIT   TKENER   TLFIX
FHCALC      2   TDNER    TKENER
FILLM       1   BOXES
FITIT       1   INITAL
FLOWPR      5   BMAIN    INITAL   TDNER    TKENER   TLFIX
FORCES      4   RMOMI    XMOMI    YMOMI    ZMOMI
GDCONV      1   BMAIN
GEOBAS      1   INITAL
GEOEXT      1   INITAL
GEOM3D      1   BMAIN
GETCPV      1   INITAL
GETDL       2   MOLOOP   PEQUAL
GETF        7   BCFLOT   BCPRES   BCTEMT   ENLOOP   INITAL   OUTPUT   QSTRUC
GETRHO      2   ICTEMP   INITAL
HEMOD       3   BMAIN    ENLOOP   MOLOOP
HLIQ        4   BCTEMP   BCTEM0   ICTEMP   INITAL
HSLIQ       1   PHYHEM
```

HSTRUC	2	BMAIN	ENLOOP				
HSVAP	1	PHYHEM					
HXMODA	1	INITAL					
HXMODB	1	BMAIN					
IARRAY	1	GEOM3D					
ICTEMP	1	INITAL					
INFORC	1	INITAL					
INITAL	1	BMAIN					
INPSTR	1	INITAL					
IREBAL	1	INITAL					
ISURFO	1	OUTPUT					
LECE00	1	ENERGI					
LECE10	1	ENERGI					
LECM00	4	RMOMI	XMOMI	YMOMI	ZMOMI		
LECM10	4	RMOMI	XMOMI	YMOMI	ZMOMI		
LECM11	4	RMOMI	XMOMI	YMOMI	ZMOMI		
LECM20	3	XMOMI	YMOMI	ZMOMI			
LECM21	3	XMOMI	YMOMI	ZMOMI			
LECM30	4	RMOMI	XMOMI	YMOMI	ZMOMI		
LECM31	4	RMOMI	XMOMI	YMOMI	ZMOMI		
LECM32	4	RMOMI	XMOMI	YMOMI	ZMOMI		
LECM33	4	RMOMI	XMOMI	YMOMI	ZMOMI		
LECM40	3	XMOMI	YMOMI	ZMOMI			
LECM41	3	XMOMI	YMOMI	ZMOMI			
LECM42	3	XMOMI	YMOMI	ZMOMI			
LECM43	3	XMOMI	YMOMI	ZMOMI			
LECY20	1	RMOMI					
LECY21	1	RMOMI					
LECY40	1	RMOMI					
LECY41	1	RMOMI					
LECY42	1	RMOMI					
LECY43	1	RMOMI					
MARRAY	1	GEOM3D					
MAXDEV	1	MOMENI					
MOLOOP	1	TIMSTP					
MOMENI	1	MOLOOP					
NCSPLI	1	FITIT					
NPROPS	1	INITAL					
OUTPUT	1	BMAIN					
PEQN	1	MOLOOP					
PEQUAL	1	MOLOOP					
PHYHEM	3	BMAIN	ENLOOP	MOLOOP			
PLTAPE	2	BMAIN	RESTAR				
PREADI	1	MOLOOP					
PSTRUC	1	OUTPUT					
QSTRUC	2	BMAIN	ENLOOP				
RARRAY	2	GEOM3D	OUTPUT				
RBSORT	1	BOXES					
REBAZ	2	CRESOR	SOLVIT				
REBAZG	2	CRESOR	SOLVIT				
REBCHK	1	IREBAL					
REBGEO	1	IREBAL					
REBLST	1	IREBAL					
REBMAT	1	REBAZG					
RESBND	3	DIREN3	DIRPR3	REBAZG			
RESLOW	1	CRESOR					
RESTAR	2	BMAIN	INITAL				
RMOMI	1	MOLOOP					
RMOMRC	1	RMOMI					
ROLIQ	3	BCTEMP	BCTEM0	GETRHO			
RSET2	1	BARIN					
RSET3	1	BARIN					
RSURFO	1	OUTPUT					
RTIME	1	TREMAN					
SELECT	1	CRESOR					
SOLVEN	3	ENLOOP	TUR1EQ	TUR2EQ			
SOLVIT	1	MOLOOP					
SOLVPR	6	CRESOR	DIRAPR	GETDL	SELECT	SOLVEN	SOLVIT
TDMA	1	REBAZ					
TDNER	1	TUR2EQ					
THCLIQ	1	INITAL					
TIME	1	DATIME					
TIMSTP	1	BMAIN					
TIMSTS	1	BMAIN					
TKENER	2	TUR1EQ	TUR2EQ				

TKSORC	3	TUR0EQ	TUR1EQ	TUR2EQ							
TLFIX	1	BMAIN									
TRACBK	42	AMAIN	BARIN	BCFLOW	BCHECK	BMAIN	BOXES	DIRAPR	DSTMSH	FILLM	FLOWPR
		FLUTAN	FORCES	GEOBAS	GEOM3D	GETF	HEMOD	HSLIQ	HSVAP	HXMODA	HXMODB
		INFORC	INITAL	INPSTR	IREBAL	NCSPLI	PEQN	PREADI	QSTRUC	RBSORT	REBAZG
		REBCHK	REBLST	SELECT	TIMSTS	TLFIX	TSCAN	TSLIQ	TSMIX	TSVAP	UPWIND
		VISORT	WLFNCV								
TREMAN	3	BMAIN	FILLM	TIMSTP							
TSCAN	1	AMAIN									
TSLIQ	1	PHYHEM									
TSMIX	1	PHYHEM									
TSVAP	1	PHYHEM									
TUR0EQ	1	TIMSTP									
TUR1EQ	1	TIMSTP									
TUR2EQ	1	TIMSTP									
UPWIND	5	ENERGI	RMOMI	XMOMI	YMOMI	ZMOMI					
VELADJ	2	INITAL	MOMENI								
VISLIQ	1	INITAL									
VISORT	1	RBSORT									
VOLEXP	1	GETCPV									
WATSTP	1	BMAIN									
WATTIM	1	BMAIN									
WLFNCV	4	RMOMI	XMOMI	YMOMI	ZMOMI						
XMOMI	1	MOLOOP									
XMOSLP	2	RMOMI	XMOMI								
YMOMI	1	MOLOOP									
YMOMRC	1	YMOMI									
YMOSLP	1	YMOMI									
ZEROAL	1	AMAIN									
ZMOMI	1	MOLOOP									
ZMOSLP	1	ZMOMI									

Notes:

1. CALSEQ is a special ASSEMBLER-tool. If not available, the only call, in the routine TRACBK, may be discarded.
2. The following routines belong to the FORTRAN environment:
   DATE, ERRSET, RTIME, TIME.

INPUT EXAMPLE

```
                                                                  Comm. Records
      +-------------------------------------------------+
      |     SAMPLE INPUT FOR STEADY-STATE CALCULATION   |
      +-------------------------------------------------+
 &GEOM                                                            Namelist GEOM
  IGEOM=0, IMAX=52, JMAX=10, KMAX=29,
  NM1=2773, NL1=2084, NSURF=19,
  ISTRUC=1, NFORCE=7, ITURKE=12,

  DX=.035, 2*.03, .025, .02, .01, .03, .04, .08, ...
  DY= 2*.02, .0263, .0237, .06, .10, .15, 2*.20, .30,
  DZ=.20, .30, 2*.40, .30, .20, .35, ...

  XNORML=  3*-0.707107, 0.8660, -0.8660, 1.0, -1.0, ...
  YNORML=  0.0, -0.707107, 3*0.0, 2*0.0, 1.0, -1.0, ...
  ZNORML=  -0.707107, 0.0, 0.707107, -0.5, 0.5, ...

   IPRES0=1, JPRES0=1, KPRES0=1,

  &END
IREG 7.0711E-4   7   7   1   2  24  24   1   NOZZLE TOP          Boundary
IREG 5.5791E-4   7   7   3   3  23  23   1                      Surface
IREG 1.0055E-3   7   7   4   4  22  22   1                      Identification
IREG 1.2728E-3   7   7   5   5  21  21   2   NOZZLE SIDE        Records
       .
       .
REG              1   1   1  10   1  29   6   +X DC C.B. SIDE
REG              6   6   1  10   1  15   7   -X DC P.V. SIDE
REG              6   6   1  10  25  29   7
       .
       .
REG 2.08400E-4  35  35   1   1  25  25  19   -Z INJ INLET
REG             36  37   1   1  25  25  19
REG 3.25230E-4  38  38   1   1  25  25  19
END   BOUNDARY SURFACE IDENTIFICATION
                                                                  Comm. Record
 &DATA                                                            Namelist DATA
  ISTATE=0, IFENER=0,
  IDTIME=0, DT=1.0,  IXREB= 1, IREBIT= 10,

  KFLOW= 7*1, 2*-3, 1, -5, 3*1, -3, 2*1, -3, 1,
  KTEMP=5*400, 1, 2*400, 2*1, 5*400, 1, 2*400, 1,
  VELOC= 18*0.0,  0.010,
  TEMP =19*300.0,
  PRES0=110.0E5, TEMP0=300.0,  GRAVZ=-9.8,

    FORCEF=6*0.5, 5.0E4,
    REYLEN=0.05, 0.19, 2*0.3, 2*1.0, 0.3,
    CLENTH=0.05, 0.19, 2*0.3, 3*    ,
      ICORR=1,1,2,2,3,3,1,
      NCORR=3,
    ACORRL=  64.0,  96.0,   0.0,      ACORRT= ...,
    BCORRL=  -1.0,  -1.0,  -1.0,      BCORRT= ...,
    CCORRL=   0.0,   0.0,   1.0,      CCORRT= ...,
```

```
    NHEATC=1,
    HEATC1=3.66, HEATC2=0.0240, HEATC3=0.8, HEATC4=0.4,
        NMATER=2,

      C0K =   38.0,   50.0,   C1K = ...,
      C0CP=  453.0,  440.0,   C1CP= ...,
      C0RO= 7856.0, 7850.0,
  NTPRNT= 1,-9999,
  NTHPR= 1201, 3201, 5201, 8201,
  &END

 &TURB
  HYDIN=0.19, TDIN=12.355, ITMAXK= 299,
  &END

ZFOR   1  33  36   1   1  23  23    INJ      )
XFOR   2   8  51   1   2  17  23    CL       )  FRICTION
  .        .
  .        .
  .        .
END   FORCE STRUCTURE LOCATION

      THERMAL STRUCTURE PROTOTYPE RECORDs (PLUS END)
 &T   N= 1, IXYZ=3, RODFR=0.01421, OUTR=0.224, &END
 &F         IHT=1, HYD=0.3,                  &END
 &M         MI=1, NP=2, DR=0.112, Q=0.0,     &END
 &T   N= 2, IXYZ=3, RODFR=0.01421, OUTR=0.224, &END
 &F         IHT=1, HYD=0.3,                  &END
 &M         MI=1, NP=2, DR=0.112, Q=0.0,     &END
  .     .          .
  .     .          .
END   THERMAL STRUCTURE PROTOTYPE

OUT    1   6   6   1   1   2  29    DC PRESS. VESS. SIDE
OUT    2   6   6   2   2   2  29
OUT    3   6   6   3   3   2  29
OUT    4   6   6   4   4   2  29
 .         .          .
 .         .          .
 .         .          .
OUT   19   7   7   1   1  16  16
END   THERMAL STRUCTURE LOCATION

      THE FOLLOWING END-RECORD IS MANDATORY
END   BOUNDARY VALUE INITIALIZATION
AL    0.41670    7   7   1   2  24  24     IRREG. CELLS
AL    0.75000    7   7   3   3  23  23     JUNCTION BETW.
AL    0.50000    7   7   4   4  22  22     CL AND DC
 .        .              .
 .        .              .
 .        .              .
ALZ   0.9622    38  38   1   1  25  25
ALX   0.0        1   5   1  10   1   1     DC OUTLET
ALY   0.0        1   6   1  10   1   1
END   INTERNAL CELL INITIALIZATION
```

Namelist DATA continued

Comm. Record Namelist TURB

Comm. Record Force Locat. Records (END incl.) (Cf. Fig.2)

Optional END Comm. Records

Namelist T
Namelist F
Namelist M

Namelist M

Optional END Comm. Record Ther. Struc. Location Records (END incl.) (Cf. Fig.2,3)

Optional END Comm. Records

Mandatory END Internal Cell Initialization Records

Mandatory END

```
+-----------------------------------------+
|    SAMPLE INPUT FOR TRANSIENT CALCULATION    |
+-----------------------------------------+
```

```
&GEOM                                                          Mandatory
 IFRES=3, ( or: IFRES=2, )                                     Mandatory
 &END                                                          Mandatory

&DATA                                                          Mandatory
 ISTATE=2,                                                     See note
        IFENER=1, TSTART=0.0,                                   Optional
 IDTIME=0, DT=0.020, 0.050,  LASTDT=5,                         Optional
 NTMAX=20000, IT=99,                                           Optional

 KFLOW(19)=101, VELOC(19)=0.4348,                              Optional
 KTEMP(19)=102,                                                Optional
 NEND=4,4,                                                     Optional

   TVAL=     0.0,    0.10,    0.10,  600.00,                   Optional
             0.0,    0.10,    0.10,  600.00,                   Optional
   FVAL=    0.23,    1.00,    1.00,    1.00,                   Optional
             1.0,   0.06667,  0.06667,  0.06667,               Optional

 NTPRNT= -9999,                                                Optional
 NTHPR=1201, 3201, 5201, 8201, 17201,                          Optional
 &END                                                          Mandatory

&TURB                                                          Mandatory
 HYDIN=0.19, TDIN=12.355, ITMAXK= 299,                         Optional
 &END                                                          Mandatory

END                                                            Mandatory
END                                                            Mandatory
```

Note:
On restart, ISTATE must be set at the

- first continuation of a steady state calculation
- commencement of a transient calculation
- first continuation of a transient calculation

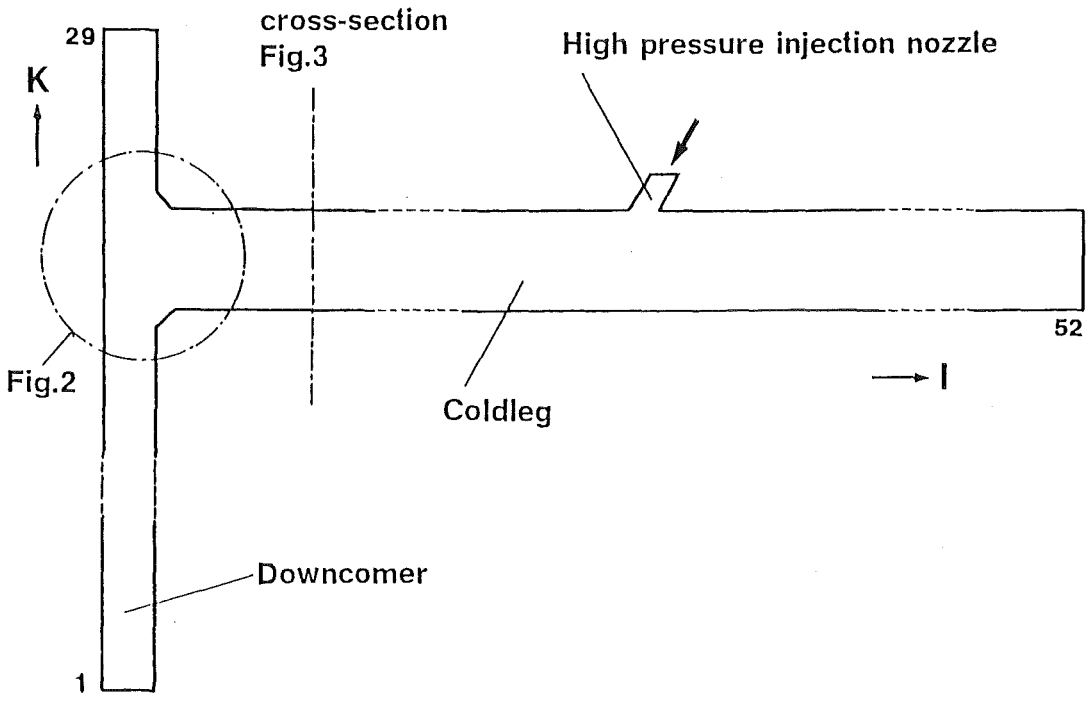

Figure 1. Input Sample Model. (Overall View).

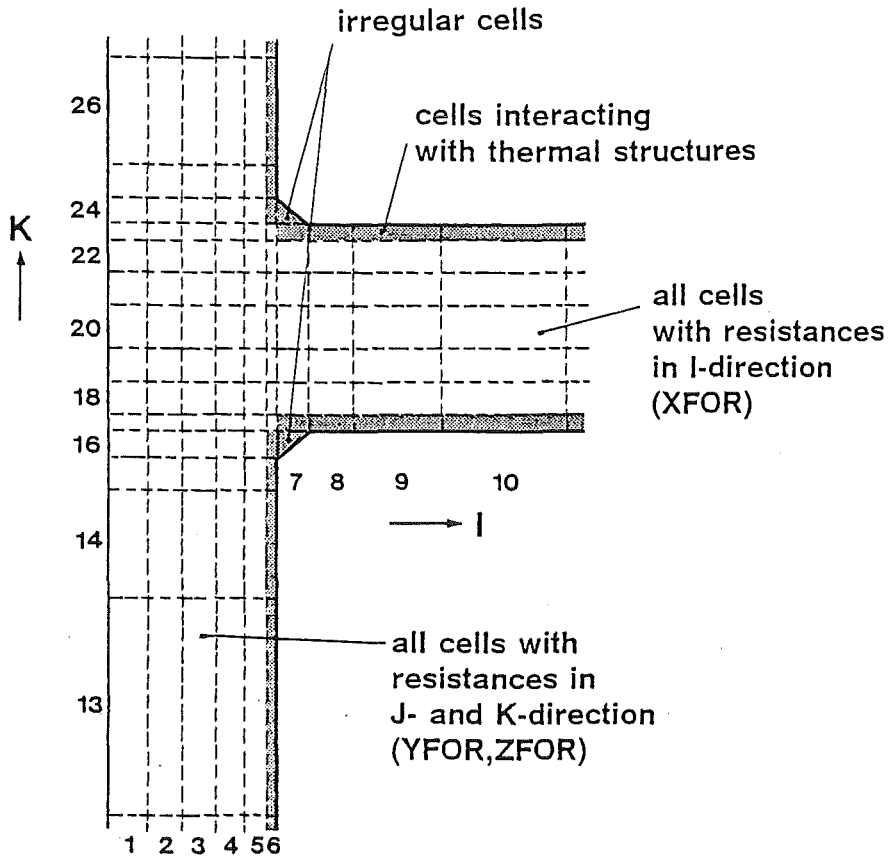

Figure 2. Input Sample Detail. Junction of Cold Leg and Downcomer.

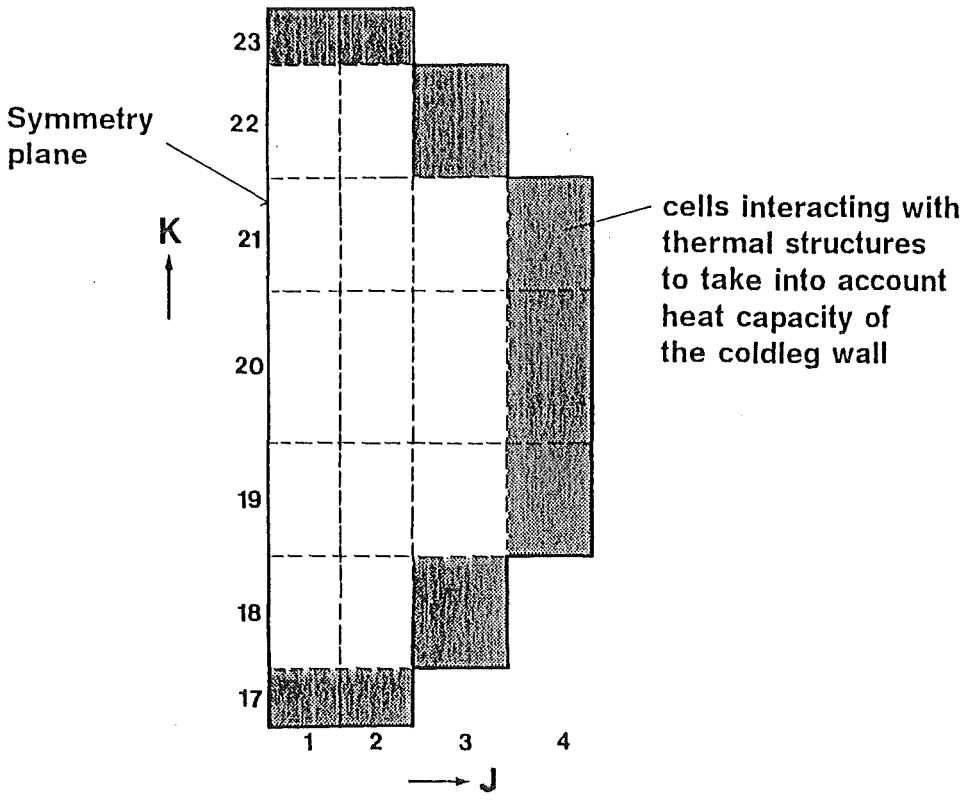

Figure 3. Input Sample Detail. Section across the Cold Leg.

Page intentionally left blank.

REFERENCES

1. H. M. Domanus, W. T. Sha, R. C. Schmitt, V. L. Shah, F. F. Chen, C. C. Miao
"COMMIX-1B: A Three-Dimensional Transient Single-Phase Computer Program for Thermal Hydraulic Analysis of Single and Multicomponent Systems, Volume I: Equations and Numerics",
NUREG/CR-4348 Vol. I, ANL-85-42 Vol. I, September 1985

2. H. M. Domanus, W. T. Sha, R. C. Schmitt, V. L. Shah, F. F. Chen, C. C. Miao
"COMMIX-1B: A Three-Dimensional Transient Single-Phase Computer Program for Thermal Hydraulic Analysis of Single and Multicomponent Systems, Volume II: Users Manual",
NUREG/CR-4348 Vol. II, ANL-85-42 Vol. II, September 1985

3. M. Bottoni, H. N. Chi, T. H. Chien, H. M. Domanus, R. W. Lyczkowski, W. T. Sha, V. L. Shah
"COMMIX-2: A Three-Dimensional Transient Computer Program for Thermal-Hydraulic Analysis of Two-Phase Flows",
NUREG/CR-4371, ANL-85/47, September 1985

4. H. M. Domanus, Y. S. Cha, T. H. Chien, R. C. Schmitt, W. T. Sha
"COMMIX-1C: A Three-Dimensional Transient Single-Phase Computer Program for Thermal Hydraulic Analysis of Single and Multicomponent Engineering Systems, Volume I: Equations and Numerics",
NUREG/CR-5649 Vol. I, ANL-90/33 Vol. I, November 1990

5. H. M. Domanus, Y. S. Cha, T. H. Chien, R. C. Schmitt, W. T. Sha
"COMMIX-1C: A Three-Dimensional Transient Single-Phase Computer Program for Thermal Hydraulic Analysis of Single and Multicomponent Engineering Systems, Volume II: Users Guide and Manual",
NUREG/CR-5649 Vol. II, ANL-90/33 Vol. II, November 1990

6. H. Borgwaldt
"CRESOR, a Robust Vectorized Poisson-Solver Implemented in the COMMIX-2(V) Thermal-Hydraulics Code",
Paper AC064, Int. Conf. on Supercomputing in Nuclear Applications (SNA '90), Mito City, Japan
March 1990

7. Cl. Günther
"A Consistent Upwind Method of Second Order for the Convection-Diffusion Equation",
Proceedings of the 1987 International Conference on Computational Techniques and Applications (CTAC-87), p. 249
Sydney, Australia
August 1987

8. Cl. Günther
"Vergleich verschiedener Differenzenverfahren zur numerischen Lösung der 2-d-Konvektions-Diffusionsgleichung anhand eines Beispiels mit bekannter exakter Lösung",
KfK 4439
Kernforschungszentrum Karlsruhe
August 1988

9. K. Sakai, D. Weinberg
"Investigation on the LECUSSO (Local Exact Consistent Upwind Scheme of Second Order) Scheme - Extension to Nonuniform Mesh Size Grids and Numerical Stability Analysis - and Application for Natural Circulation Test Analyses using COMMIX-2(V)",
Paper AS081, Int. Conf. on Supercomputing in Nuclear Applications (SNA '90), Mito City, Japan
March 1990

10. M. Bottoni, G. Willerding, W. Baumann
"Erstellung einer Referenzversion des 3D-Thermohydraulik-Rechenprogramms COMMIX",
unpublished report, Kernforschungszentrum Karlsruhe (1986)

11. G. Willerding, W. Baumann
"FUNGEO - Eine Prozedur zur Erzeugung der FLUTAN-Eingabe für reguläre Rechtwinkel- und Zylindergeometrie",
unpublished report, Kernforschungszentrum Karlsruhe (1992)

12. G. Willerding, W. Baumann
"HEXPIN - Ein Input-Generator für COMMIX-2(V)-Rechnungen mit hexagonaler Stabbündelgeometrie",
unpublished report, Kernforschungszentrum Karlsruhe (1990)

13. G. Willerding
"COMCHK - Ein Eingabeprüfer für COMMIX-2(V)",
unpublished report, Kernforschungszentrum Karlsruhe (1990)

A

B

C

D

E

F

G

H

I

J

K

L

M

N

Z