KfK 3185
EUR 7054e
Juni 1981

SAGAPØ-A Code Description and User's Guide

S. Cevolani
Institut für Neutronenphysik und Reaktortechnik

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE

Institut für Neutronenphysik und Reaktortechnik

KfK 3185

EUR 7054e

SAGAPØ-A Code Description and User's Guide

S. Cevolani *)

*) Delegated from Euratom to the Kernforschungszentrum Karlsruhe

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Als Manuskript vervielfältigt
Für diesen Bericht behalten wir uns alle Rechte vor

Kernforschungszentrum Karlsruhe GmbH
ISSN 0303-4003

Abstract

This paper describes the new models inserted in the computer code SAGAPØ-A for the thermo-fluiddynamic analysis of gas cooled fuel element bundles. Moreover, it is intended to be a guide for the user of the code.

The physical background of the new models inserted in the code has been described by the author of this work in a previous paper.

A listing of the code is included in the Appendix.

SAGAPØ-A Beschreibung des Rechenprogramms und Benutzerhandbuch

Zusammenfassung

In diesem Bericht werden die in dem für die thermo- und fluiddynamische Analyse von gasgekühlten Brennelementbündeln neu eingefügten Modelle beschrieben.

Die physikalischen Grundlagen der in dem Rechenprogramm eingefügten Modelle wurden vom Verfasser dieses Berichts in einer vorherigen Arbeit vorgestellt.

Eine Liste des Rechenprogramms wird als Anhang gegeben.

Contents:

Page

1. Introduction

SAGAPØ-A is an improved version of the computer code SAGAPØ /1/ for the thermo- fluiddynamic analysis of gas cooled fuel element bundles.

The most important improvements with respect to the earlier version of the code are: 1) the insertion of models for the calculation of the heat transfer due to radiation and 2) the heat conduction within the pins and the shroud. Further modifications regard the treatment of some physical phenomena, such as laminar flow and spacer grids effect on the heat transfer.

The physical models and the mathematical procedures used in the earlier version of the code are described in ref. /2/, while the physical phenomena relative to the improvements introduced in the last version were described by the author of this work in a previous paper /3/.

Further modifications are concerned with the structural part of the code, in order to allow the insertion of the new models and the interaction between the different parts.

Purpose of this work is:

a) to describe the structure of the new developed models and the modifications performed on the code in order to allow their insertion

b) to supply a guide to the user of the code.

To avoid repetitions, no description of the unchanged parts of the code will be given here. The reader is also supposed to be familiar with the ref. /1/ and, of course, ref. /2/ and /3/.

As pointed out in ref. /5/, SAGAPØ-A can run with dynamic dimensioning. However, in this work the dimension of arrays and vectors will be referred to the whole 19-rod bundle.

2. General informations

2.1 Geometries

In axial direction we consider first the physical subdivision of the bundle in axial portions, generated by the fact that the rods are only partially heated and only partially roughened: Fig. 2.1 shows the generation of the axial portions from the superposition of the heated/unheated and of the rough/smooth subdivisions.

Furthermore, the axial portions are subdivided, for a mathematical purpose, in axial sections: these represent the mesh used for the calculation in axial direction. The length of the axial sections is defined as factor X_D times the equivalent diameter of the central channels (cfr. 3.1 in /1/).

The flow section is subdivided into channels, subchannels and sub-subchannels, as described in ref. /1,2,3/. Figures 2.2 to 2.7 show the indexing of the channels and of the subchannels for some different bundles, both in the case where the calculation is performed for the whole bundle and in the case when the calculation is limited to a symmetry section.

The pins and the liner are subdivided, in circumferential direction, into sectors; these represent the part of the pin (or of the liner) adjacent to a subchannel. Figures 2.7 and 2.9 show the sector subdivision of pin and liner.

In radial direction the pins are subdivided in different manner depending on their nature: when the pin consists of a directly heated tube (KE4 and B193D bundles), no radial subdivision is performed. In this case, by assuming the inner wall adiabatic and the thickness of the tube small in comparison with the diameter, the one-dimensional Fourier's equation is solved, neglecting the temperature gradient in radial direction at this point of the calculation; a further correction is introduced to consider the position of the thermocouples within the wall (cfr. ref./3/).

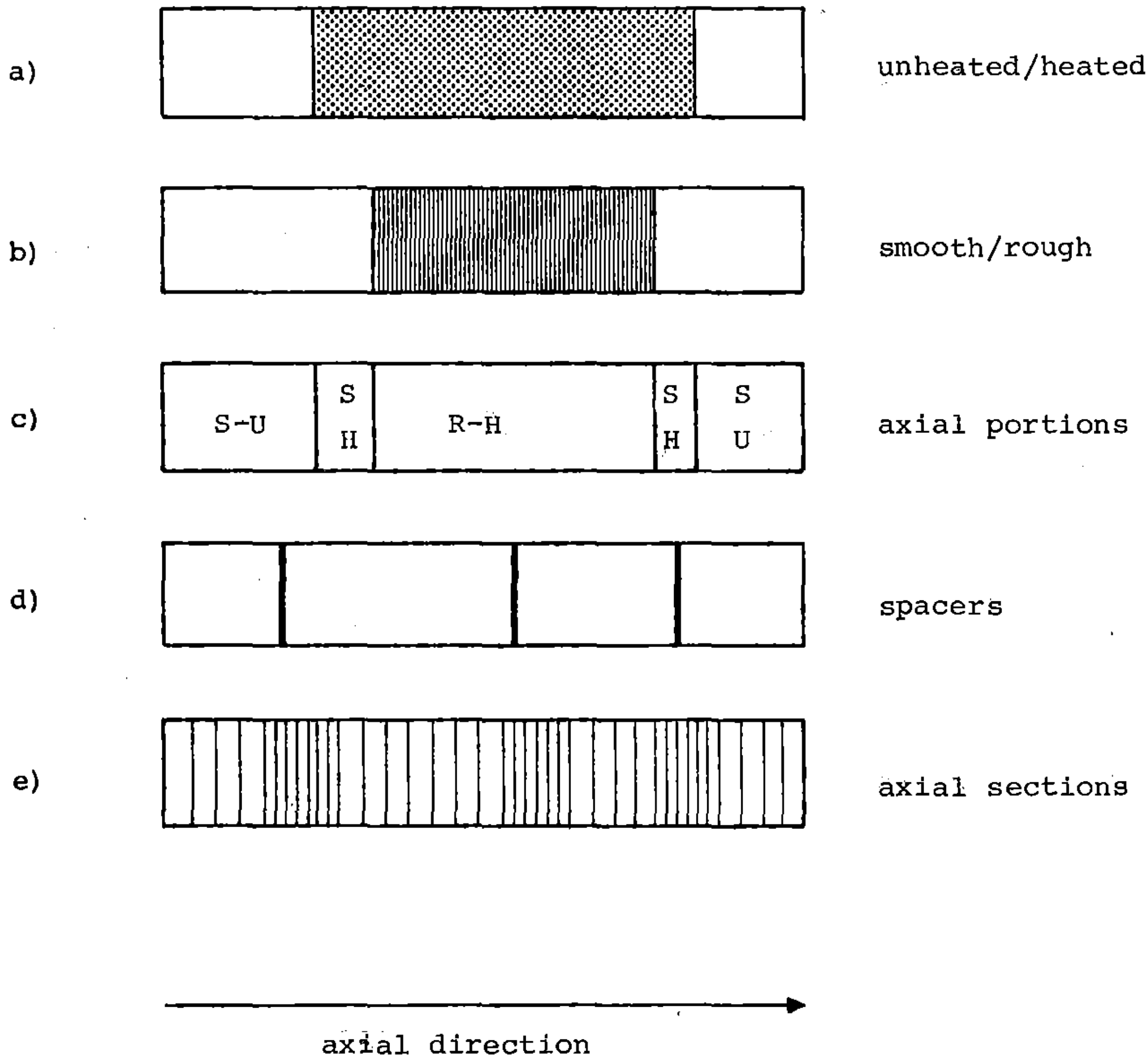


Fig.2.1: Definition of axial portions and axial sections.

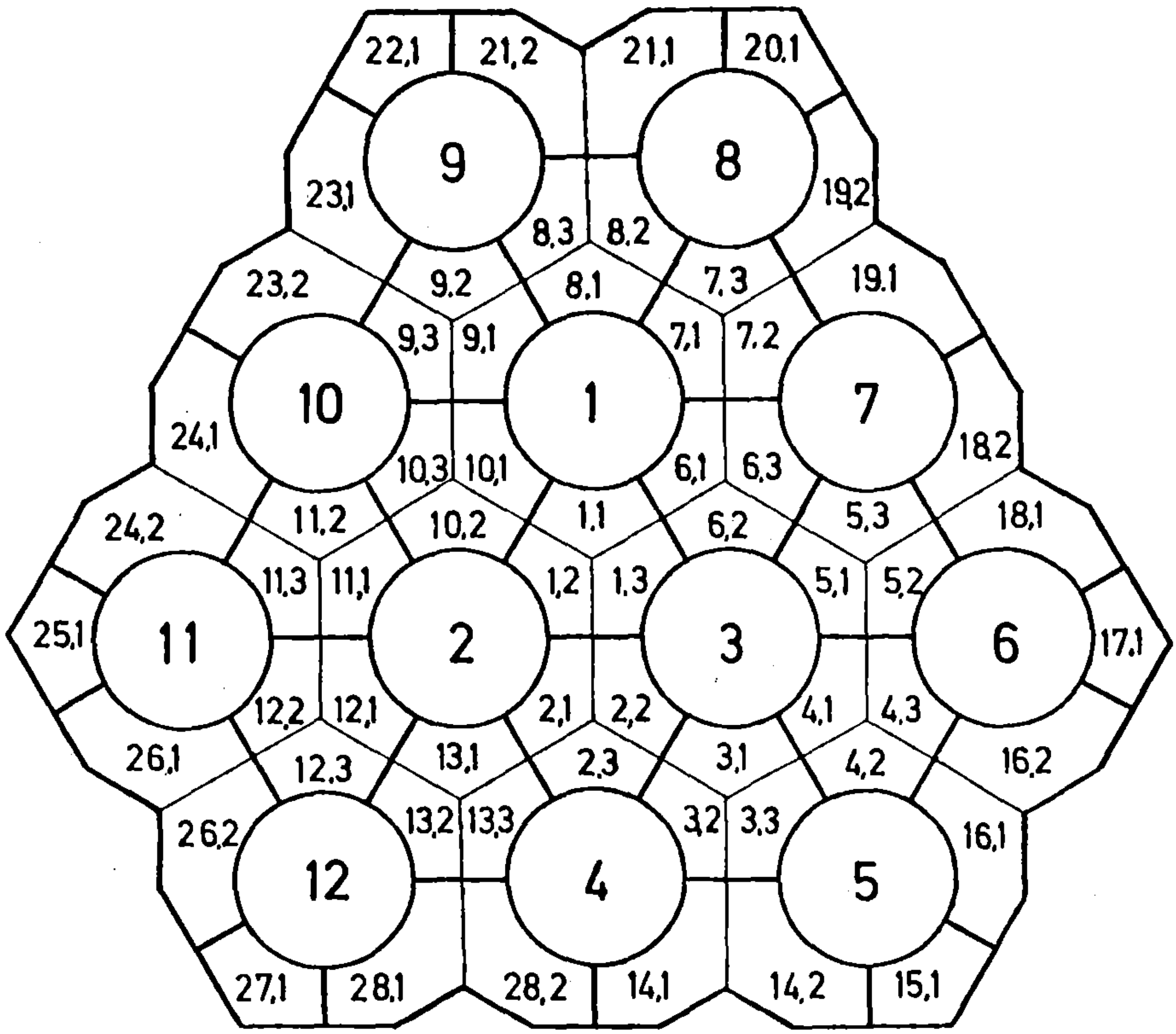


Fig.2.8: Indexing of the channels and of the subchannels for the whole flow section of the 12-rod bundles.

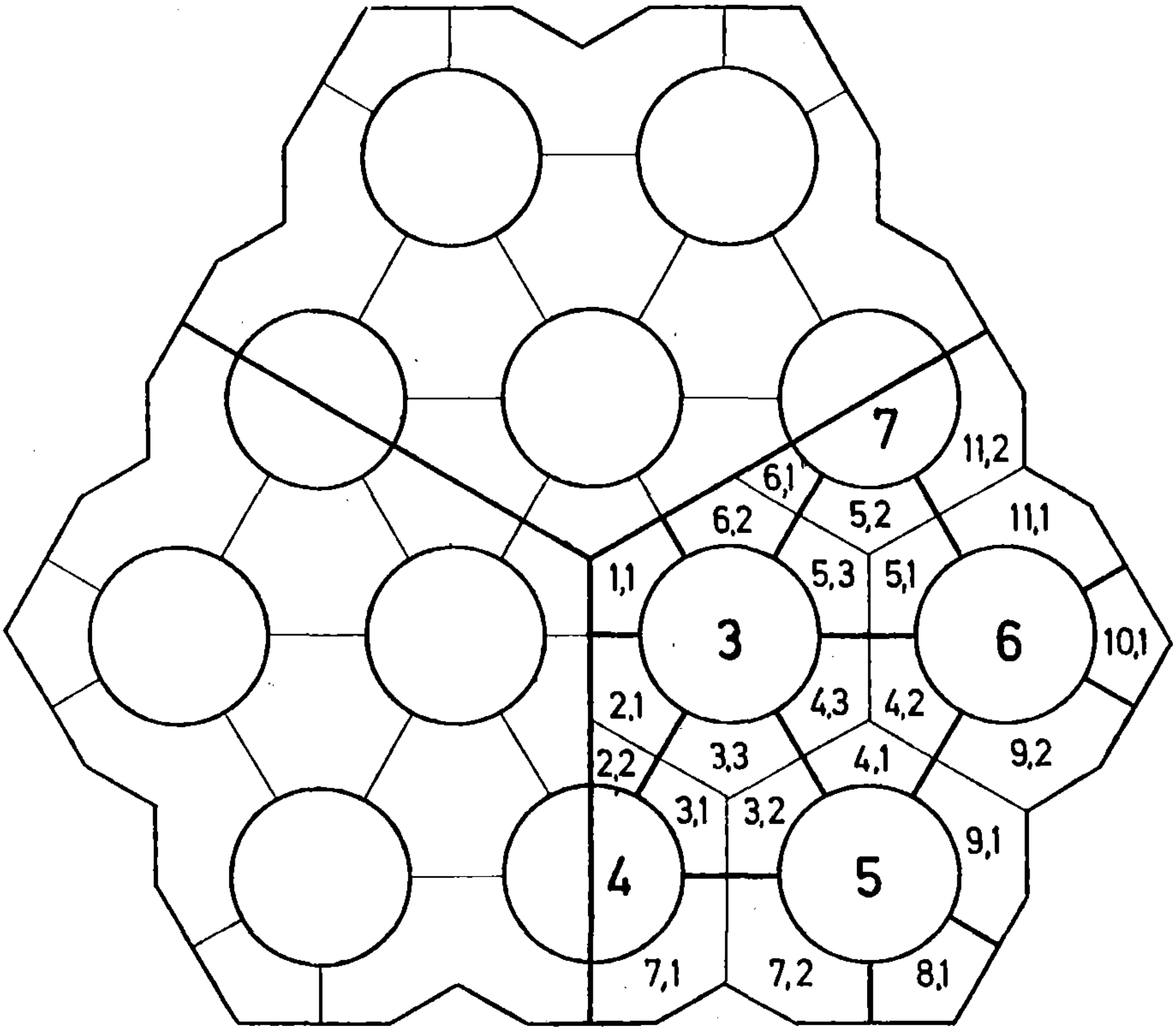


Fig.2.3: Indexing of the channels and of the subchannels for 1/3rd of the whole flow section of the 12-rod bundles.

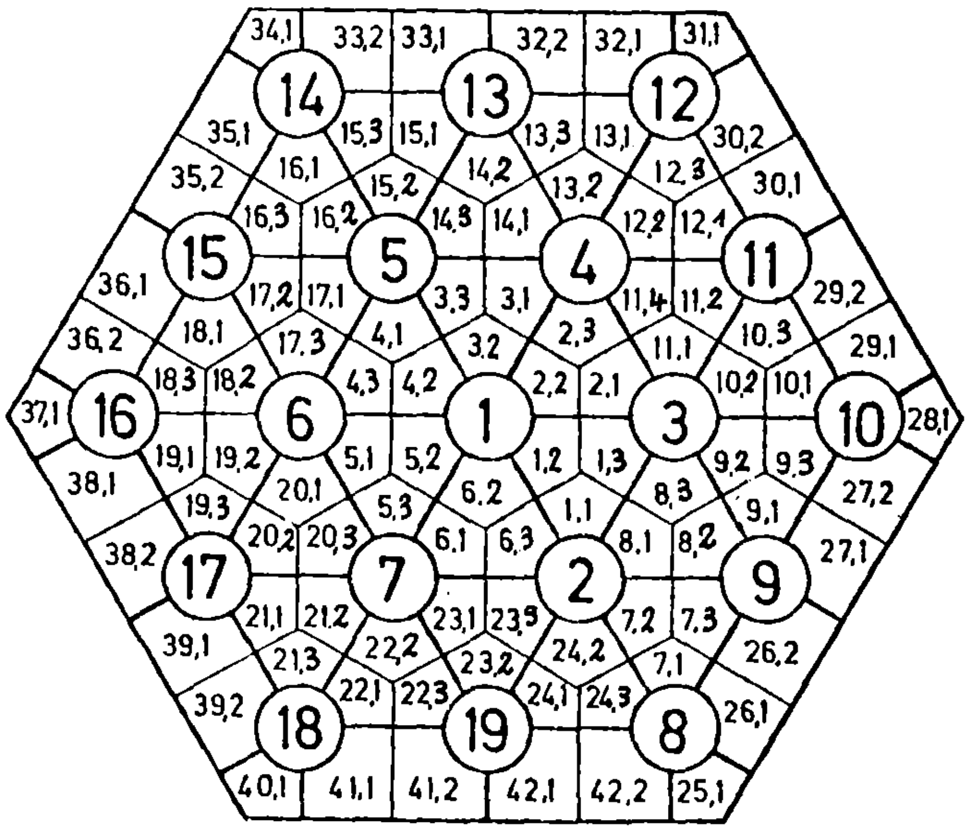


Fig.2.4: Indexing of the channels and of the subchannels for the whole flow section of a 19-rod bundle

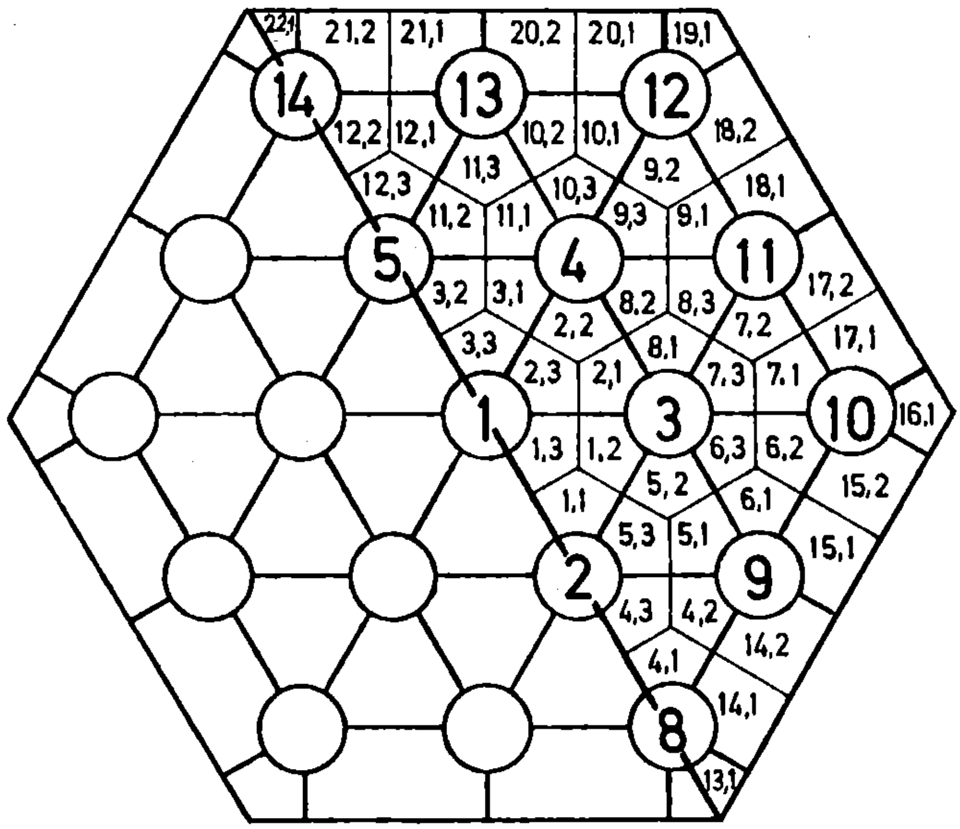


Fig.2.5: Indexing of the channels and of the subchannels for a half of the whole flow section of a 19-rod bundle.

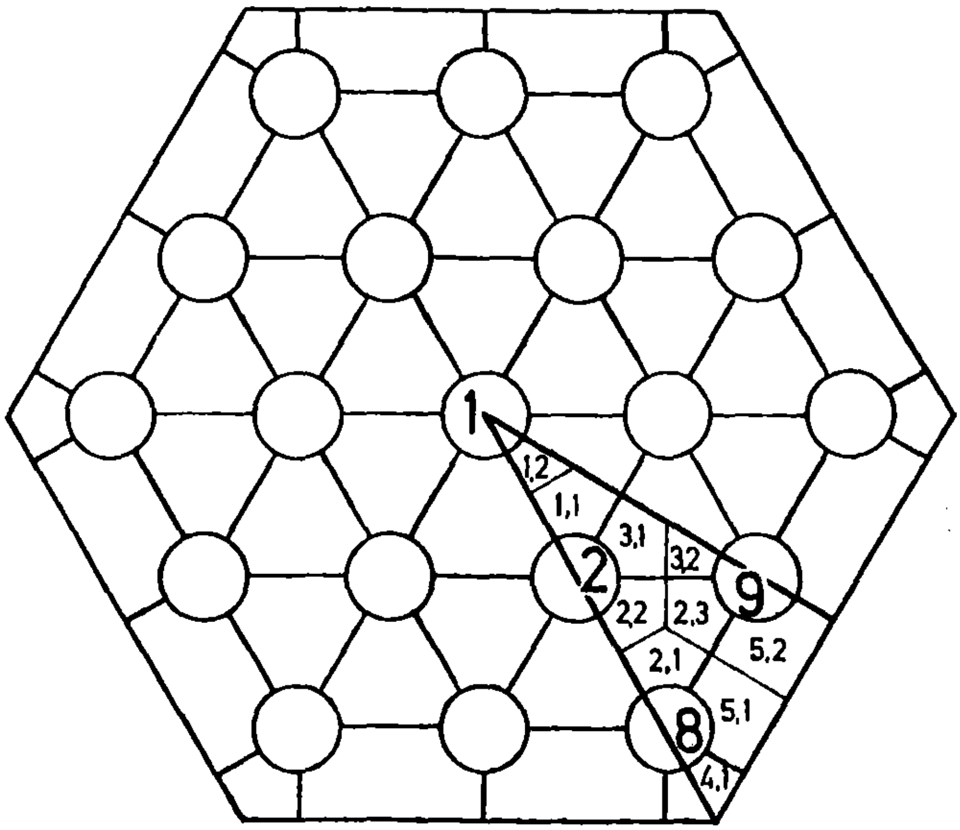


Fig.2.6: Indexing of the channels and of the subchannels for 1/12th of the whole flow section of a 19-rod bundle.

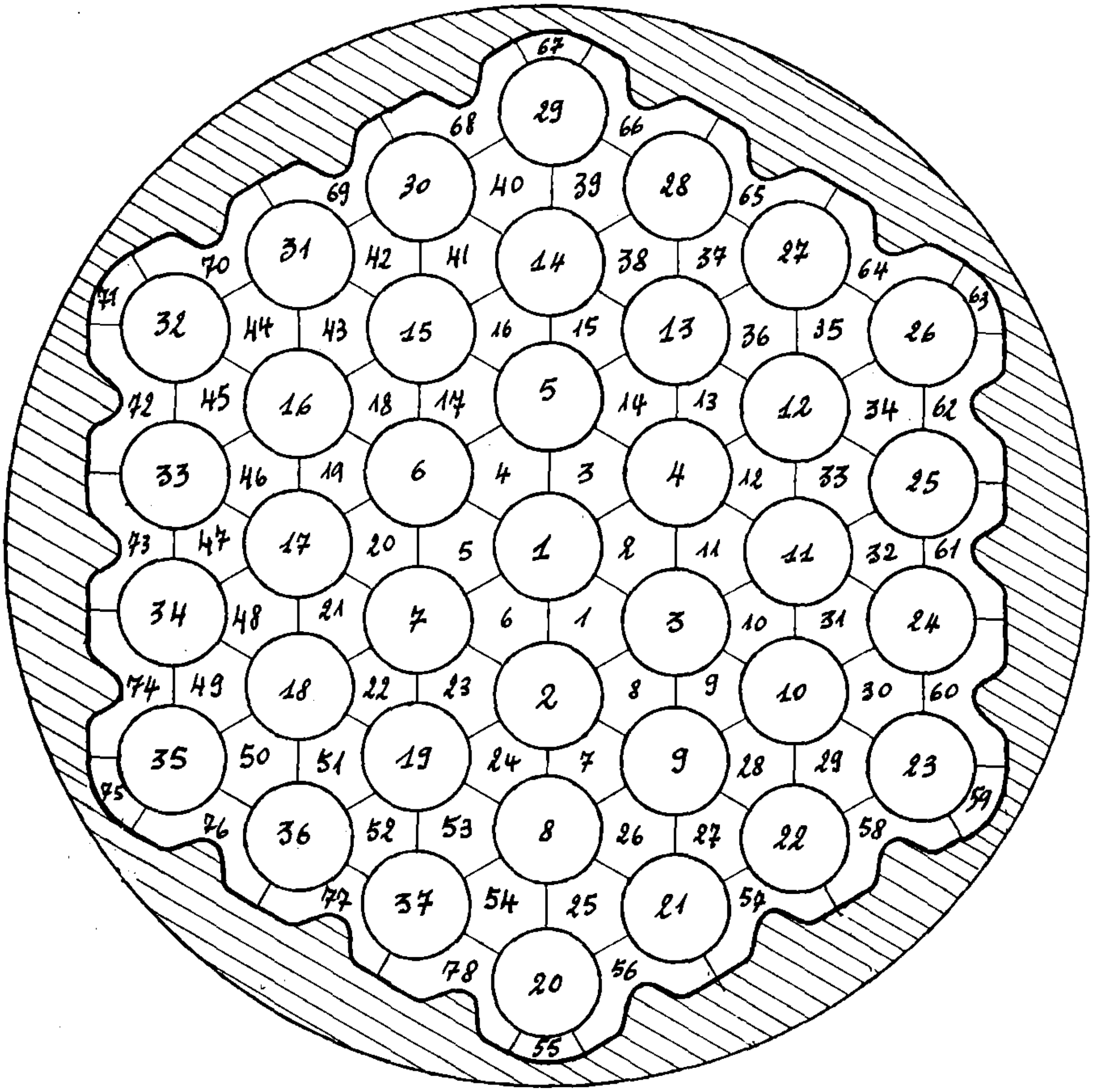


Fig.2.7: Indexing of the channels and of the rods for the whole 37-rod bundle.

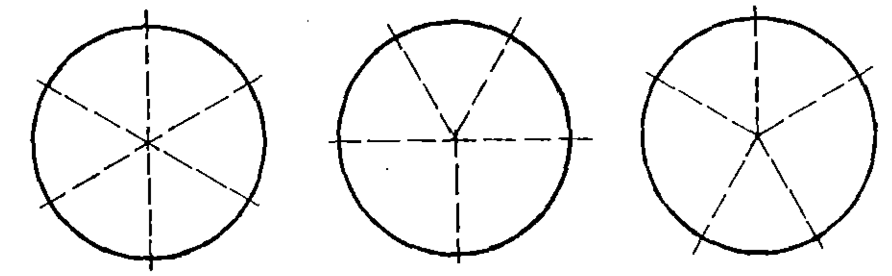


Fig. 2.8: Sector subdivision for the pins

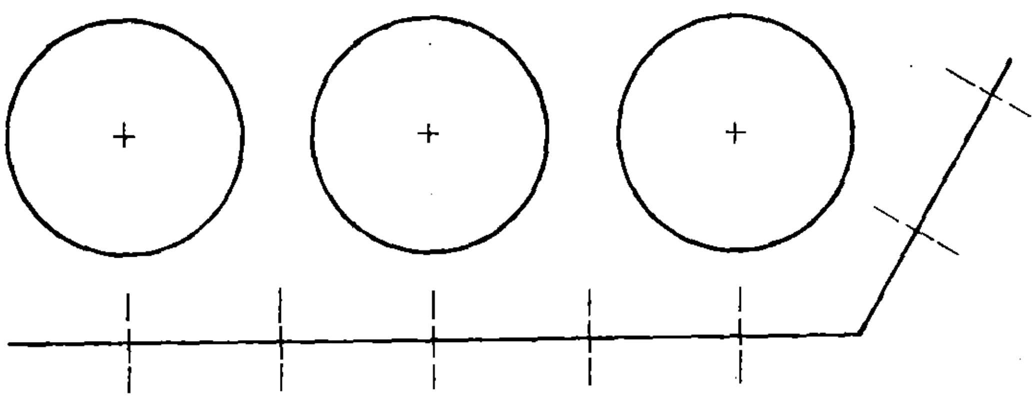


Fig. 2.9: Sector subdivision for the shroud.

If the pin consists of an inner generating core separated from the external tube by an insulated gap, the two-dimensional Fourier's equation is used.

Also for the thermal conduction within the shroud, no radial subdivision is performed: by assuming the thickness of the shroud to be small in comparison with the diameter, the problem is solved using the one-dimensional Fourier equations. The explained sector subdivision in circumferential direction is also used for the radiation calculation: the sector is the surface unit for the analysis of the radiation exchange.

The geometrical aspects of the convergence procedure are shown in fig. 2.10, where the general flow chart of SAGAPØ-A is presented: starting from the beginning of the bundle, each axial portion will be subdivided into axial sections. In each axial section the calculation is performed for each channel, subchannel and sector (SUBCHANNEL ANALYSIS). When the convergence is reached (INDICE=O), the calculations goes on to the next section, until the end of the bundle is reached.

2.2 Recall of the solution method

As described in /3/, the solution of the thermo- and fluiddynamic problem is obtained separating the different phenomena: as shown in fig.2.11, at each axial section, at the beginning of the calculation, the convective heat flux relative to each sector is computed as geometrical function of the heat generated within the pin (or within the shroud).

On the basis of these heat fluxes, the flow problem is solved for the distributions of mass flow, gas temperature, and pressure loss. In this step also the convective heat transfer coefficients are

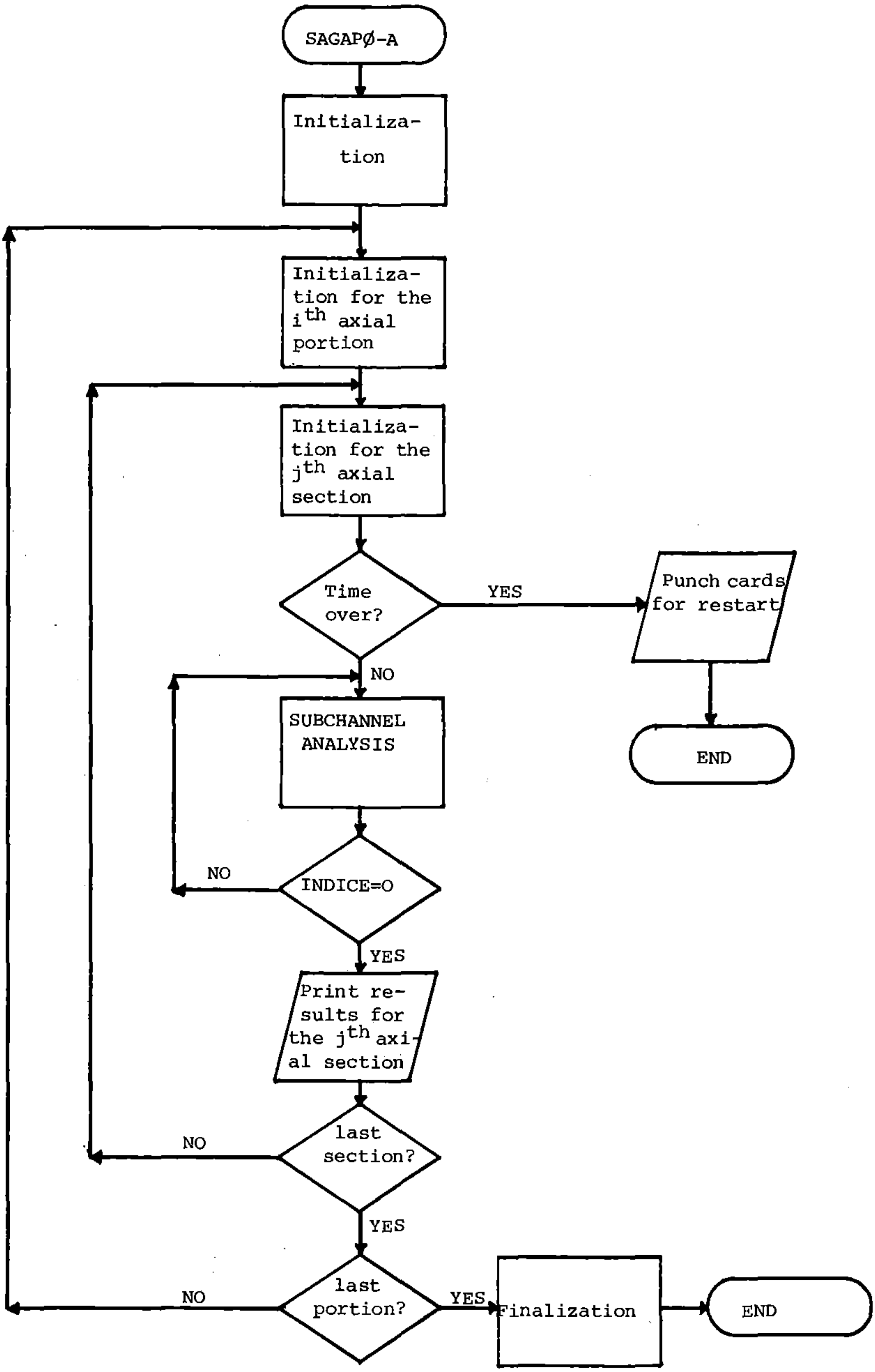
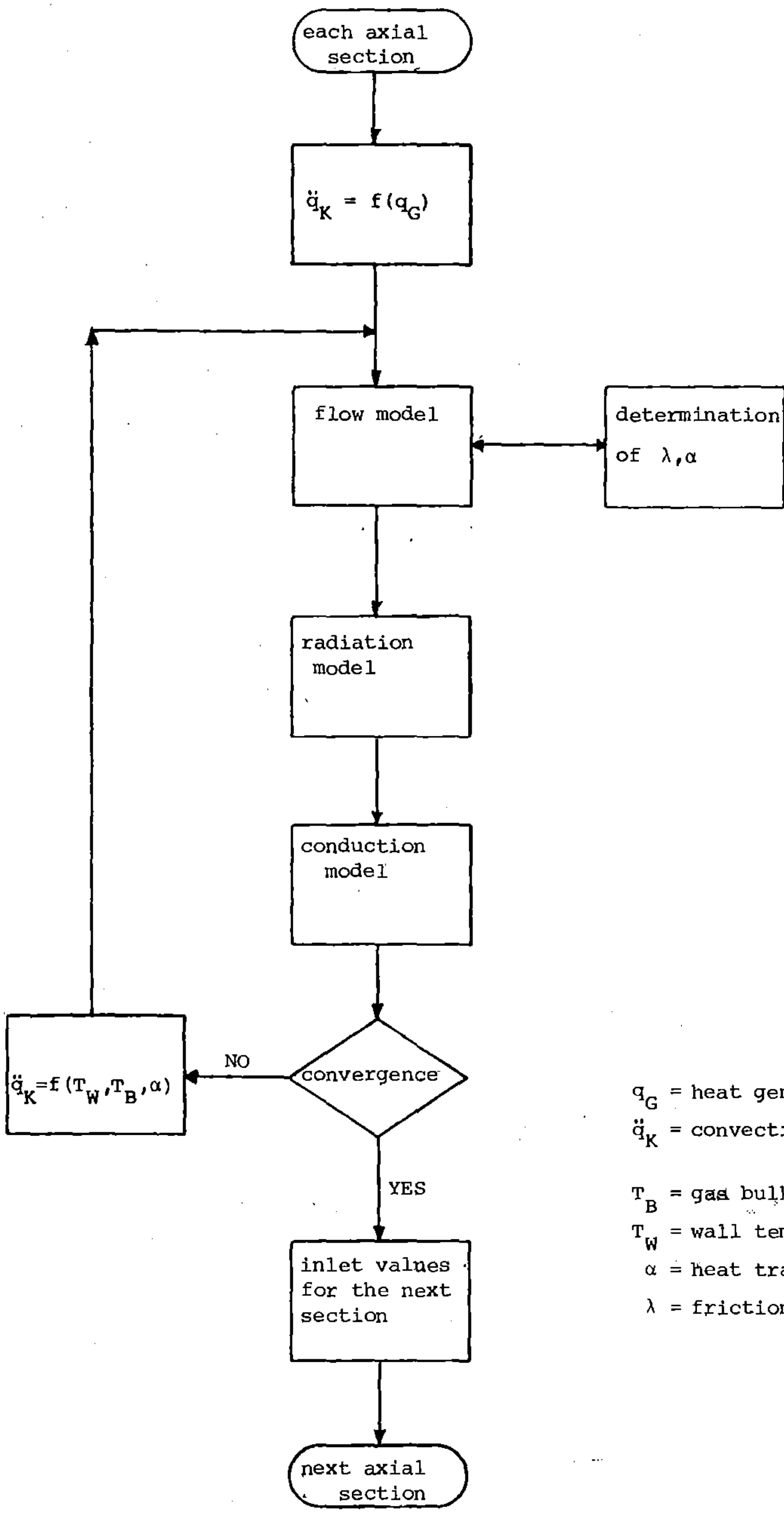


Fig. 2.10: Schematical flow chart of SAGAPØ-A



q_G = heat generated
$\ddot{q}_K$ = convective heat flux

T_B = gas bulk temperature
T_W = wall temperature
α = heat transfer coefficient
λ = friction factor

Fig. 2.11: Separation process for the different heat transfer modes.

computed. It is therefore possible to compute approximate values for the wall temperatures. Using these wall temperatures as boundary conditions, it is possible to solve the radiation problem for the heat exchanged by radiation.

At this point, the thermal conduction equations are solved, using the bulk temperatures, the radiative heat fluxes and the heat transfer coefficients as boundary conditions.

The convergence is now tested (for pressure drop, bulk and wall temperatures), and, if no convergence is achieved, the convective heat fluxes are calculated one more time as a function of the heat transfer coefficient and the bulk and wall temperatures. The process is now repeated, until the convergence is reached. With reference to the figures 2.12 and 2.10 it must be pointed out that the flags INDICE, IFLØW and IHEAT are used just to explain the convergence process: they do not correspond to the flags really used in the code. This is due to the fact that the convergence process in the code is slightly (but merely formally) different from the process shown here.

2.3 Structure of the code

The flow chart of the procedure explained in the previous paragraph is shown in the figures 2.12 to 2.14.

The main connexion between the earlier version of the code and the new models is given by the subroutine TEMCØN, which organizes the thermal calculation.

At the beginning, TEMCØN (see fig. 2.13) calls the subroutine WALLTE, which computes the heat transfer coefficients and a first approximate value for the wall temperatures. WALLTE is a modified version of the subroutine with the same name present in the earlier version of the code. Then TBRTBS computes the dummy bulk temperatures (cfr. 5.5 in ref. /3/) and RADIA organizes the calculation for the radiation problem. The option IRAD allows neglecting the radiation calculation.

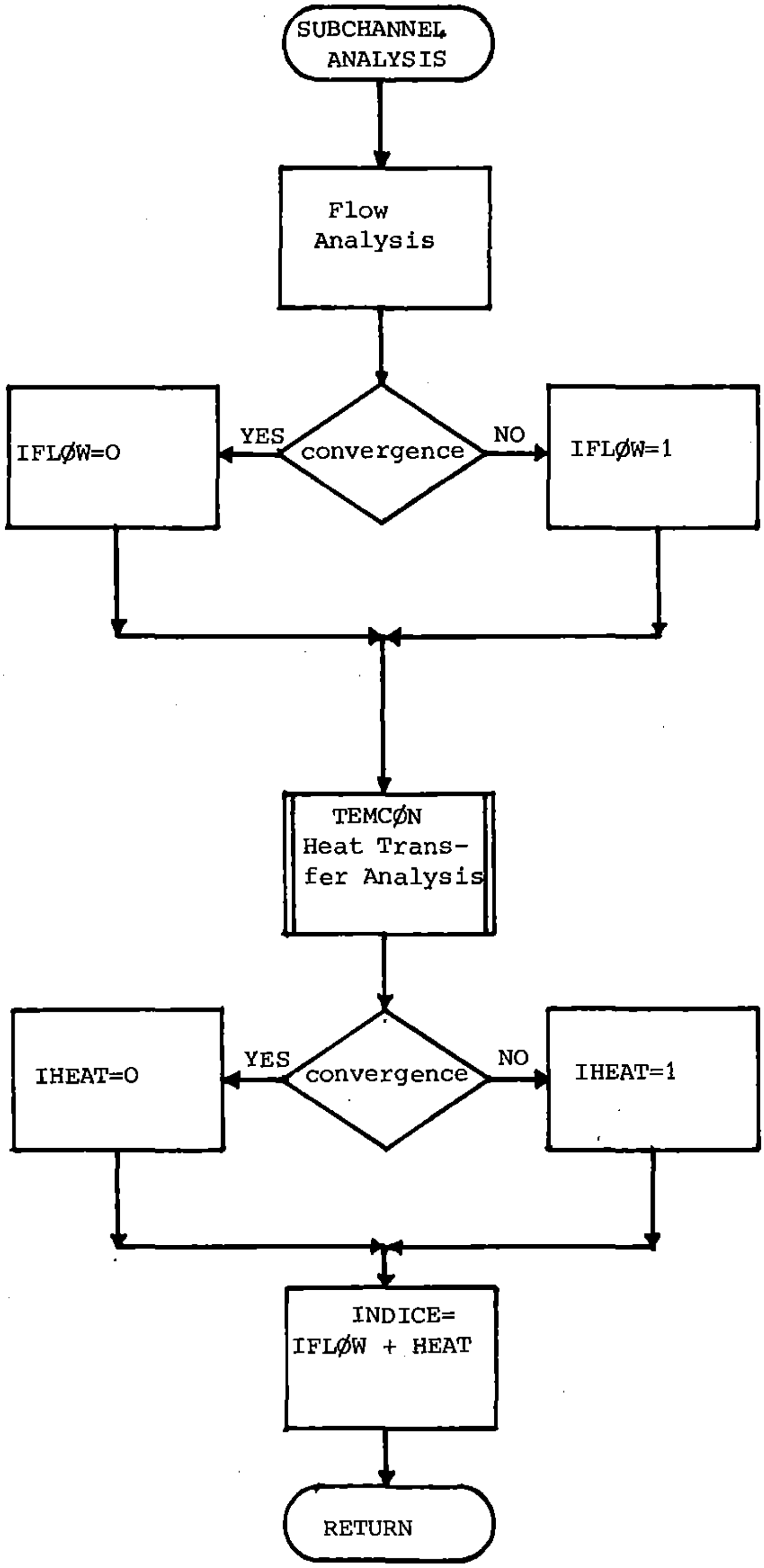


Fig. 2.12: Schematic flow chart of the loop ITCØRR (subchannel analysis).

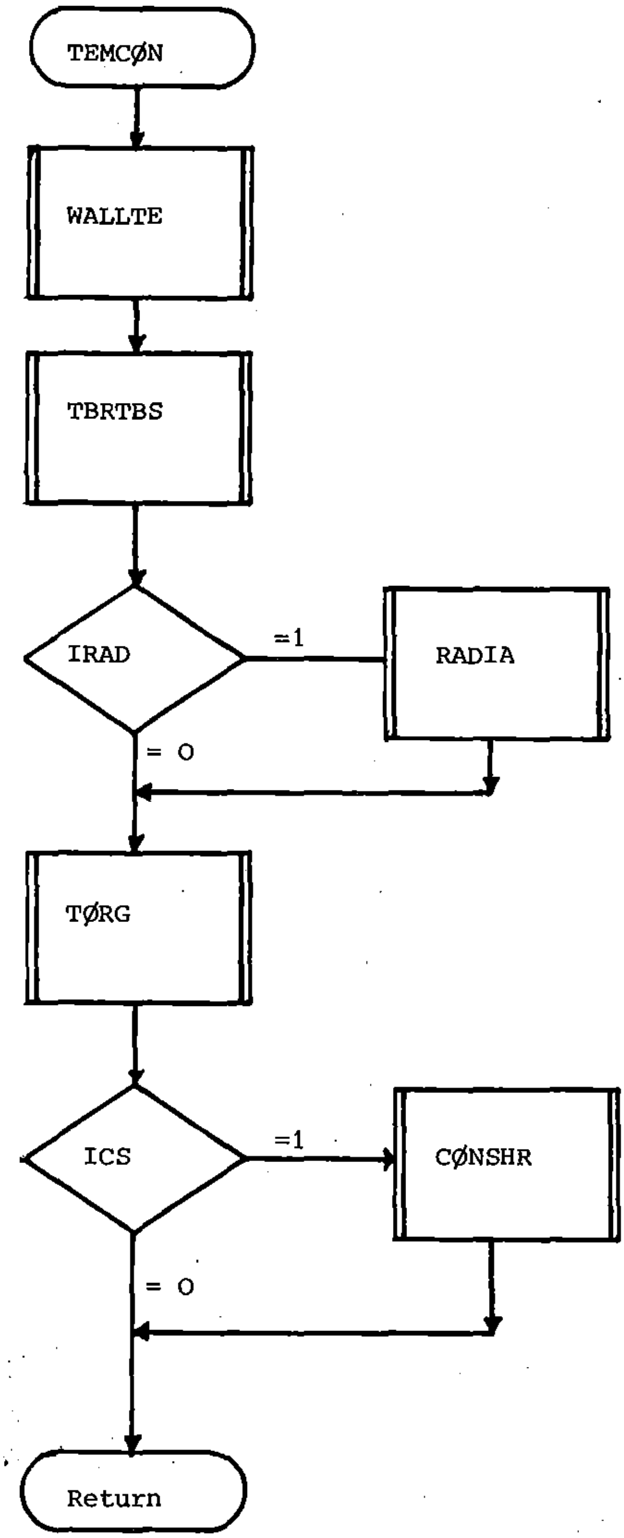


Fig. 2.13: Schematic flow chart of the subroutine TEMCØN

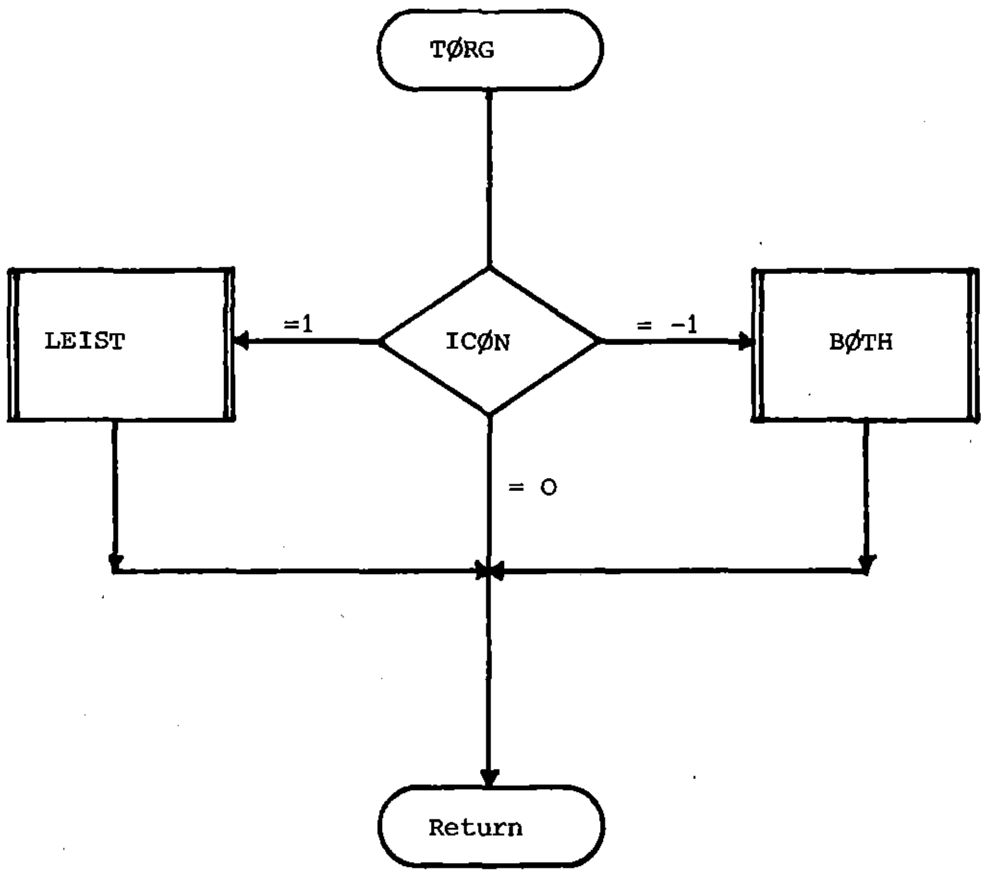


Fig. 2.14: Flow chart of the subroutine TØRG

The subroutine TØRG (see also fig. 2.14) determines the conduction model to be used depending on the option ICØN: the subroutine LEIST performs the one-dimensional analysis, while BØTH uses the two-dimensional approach in the case of central heated core. Also the thermal conduction can be neglected (ICØN=0), but in this case no radiation calculation is allowed. Finally, the subroutine CØNSHR organizes the calculation for the thermal conduction within the shroud, also under the option ICS.

3. Input/Output

In fig. 3.1 the input/output files of SAGAPØ-A are shown.

The input is provided as "INPUT DATA", whose records are read from the code, and "BLØCK DATA". The main modifications of the input parameters, with respect to the earlier version of the code, have been performed on the "BLOCK DATA".

The output files consist of a LISTING, partly modified with respect to the earlier version, of the files RESULTS*4, where the results of the calculation are stored for further plotting and comparison with experimental results, and of the file RESTART which are automatically written by the code if the available calculation time is going to be exceeded.

3.1 INPUT DATA

The geometrical data of the bundle and the thermo- and fluiddynamic conditions of the case to be computed are mainly given in this file.

No significant modifications have been performed with respect to the earlier version of the code. The changes are concerned with the following items:

a) A new card must be inserted before the first card in ref. /1/. Any text can be written in this new card, no use of which will be done by the code: the card allows the identification of the block of input cards.

b) Cards 39,40 and 41 (and 39a, 40a, 41/a etc) have been eliminated. Moreover no use of the parameters GRIP, GRI1 and GRI2 is done in the present version of the code. These parameters are automatically set to 1.

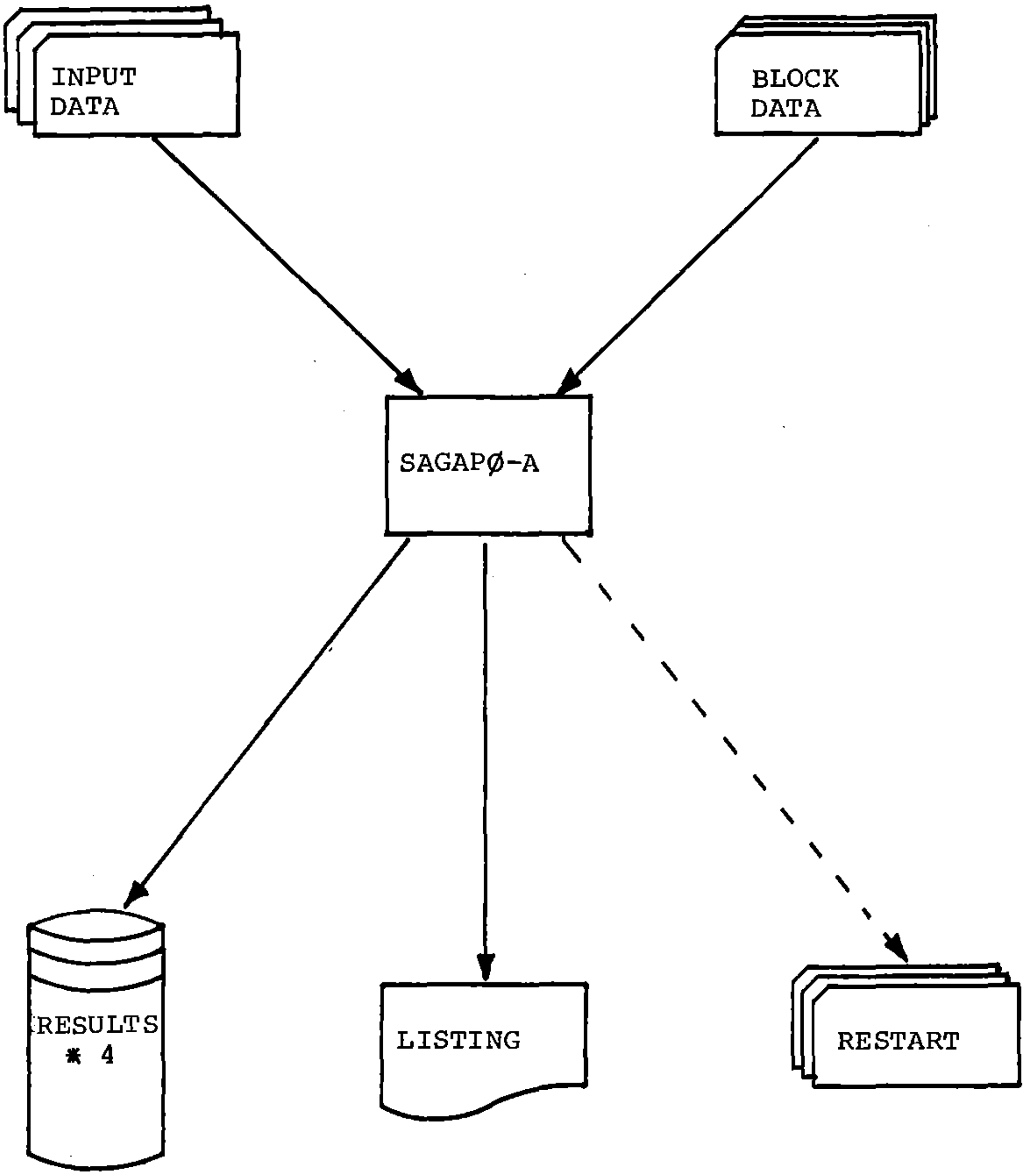


Fig. 3.1: Input/output for SAGAPØ-A

3.2 BLØCK DATA

The BLØCK DATA has been strongly expanded with respect to the earlier version. Actually, it provides the options for the calculations, the coefficients for some functions and, in some cases, arrays concerned with the indexing of the bundle.

The new parameters to be provided in the BLØCK DATA (beyond the parameters already described in /1/) are listed in Tab.3.1.

According to /1/, in case of a 12-rod bundle further parameters are to be provided in BLØCK DATA.

If a symmetry section of the bundle (12-rod bundle case) is computed, also the parameters listed in Tab. 3.2 are to be provided in BLØCK DATA. Differently from the other parts of the paper, in this table the dimensions of the arrays are referred to 1/3 of the whole 12-rod bundle.

COMMON BLØCK	ARGUMENT	EXPLANATION
SCO4C	RFUEL	Radius of the heated pin core (fuel); it corresponds to r_f of fig. 5.1 in /3/ (cm)
SCO6C	D3(3)	Coefficients for the thermal conductivity of the cladding material (K=D3(1)+T*D3(2)+T^2*D3(3)).
SCO9C	IS	IS=1 The superposition principle for the external channels (cfr.5.5 in /3/) is to be used. IS=0 no superposition principle
SC18C	RE, RI, ALFW, ALFC	RE = outer radius of the ring corresponding to the liner (cfr. 5.3 in /3/) (cm) RI = inner radius of the liner equivalent ring (cm) ALFW = angle (in radiants) corresponding to 1/2 of a wall channel: 2*RI*ALFW = perimeter of a wall channel ALFC = angle in radiants corresponding to a corner channel.
SC19C	ICØN	ICØN = 1 Conduction within the clad ICØN = -1 Conduction within core and clad ICØN = 0 no conduction
SC22C	NTØT	Total number of sectors in the shroud
SC23C	ICS	ICS = 1 conduction in the liner ICS = 0 no conduction in the liner
SC25C	NPRINT AR(10)	NPRINT < 0 Print results at each axial section NPRINT = 0 Print results between the positions defined by AR(1) ≤ x ≤ AR(2) NPRINT > 0 (≤10) Print results at the axial sections corresponding to AR(1), AR(2),...,AR(NPRINT)
SCO2L	JLAM	JLAM = 1 The K values for the laminar friction factor for corner and wall channel are averaged (cfr. 3.3.2 in /3/). JLAM = 0 no average

Tab. 3.1: New CØMMØN BLØCKS in BLØCK DATA

COMMON BLØCK	ARGUMENT	EXPLANATION
SCO3L	IPHUD	IPHUD = 1 Heat losses at the shroud are computed by means of the EIR method. IPHUD = 0 Heat losses at the shroud are given by input.
SCO4L	TAMB, HUDFAC, TMS	TAMB: Room temperature during the experiment (C) HUDFAC: total heat transfer coefficient between liner and room (cal $sec^{-1}cm^{2}$ C^{-1}) TMS: average shroud temperature (C)
SCO5R	IRAD	IRAD = 1 Thermal radiation is taken into account IRAD = 0 no thermal radiation
SC15R	IEPS	IEPS = 0 constant emissivity for the pins and on the shroud sectors. IEPS = 1 emissivity as function of the temperature
SCO7R	EPSR, EPSS, SIGMA	EPSR = constant value assumed for the rod emissivity (if IEPS = 0) EPSS = the same as EPSR, but for the shroud SIGMA = Stephan-Boltzmann constant ($1.35*10^{-12}$ cal cm^{-2} sec^{-1} K^{-4})
SCO1S	IHAS	IHAS = 1 the correlation for the heat transfer improvement due to the spacer is based on the local Re, Pr, ε, D_h IHAS = 0 the spacer correlation is based on average Re, Pr, ε, D_h
SC16R	ICØMP	ICØMP = 1 The view factors are normalized by modifying each factor. ICØMP = 0 The normalization is performed by introducting the dummy factor f_{ii} (cfr. 4.3.2 in /3/).

Tab. 3.1: Cont.

CØMMØN BLØCK	ARGUMENT	EXPLANATION
SCO1P	S(19)	Alternative method to define non-uniform power distribution in the pins: instead to read Q(I), I=1,NRØDS als input data, it is possible to read only one Q and to define the vector S; for each pin it will be Q(I) = Q + S(I) This method was used for the 19-rod bundle calculation by uniform heating, to consider the slight power differences between the pins.
CEVO1	LAMØP1	= 2 in laminar flow, the spacer smooth correlation for the pressure loss is used in the rough part too = 1 spacer rough correlation in the rough part
CEVO3	LAMOP2	= 1 corrections for gas thermal conduction between the ribs = 2 no correction
CEVO4	LAMOP3	= 1 for laminar flow = 2 for turbulent flow
CEVO2	RTIP4	Tip radius of the rough part of the pin (cm).
SC34C	ISUC	= 1 Superposition principle for the central sectors (cfr. 7.45) = 0 No superposition principle for the central sectors
SCO2P	IPSUB	= 1 New and old print for the subchannel variables = 0 only new print

Tab. 3.1: Cont.

CØMMØN BLØCK	ARGUMENT	EXPLANATION
SCO1C	NCAN(12): LIPS(12,10):	NCAN(J): number of channels adjacent to pin 5 LIPS(J,K): index of the K^{TH} channel adjacent to pin J. LIPS is used for the conduction calculation: the channels LIPS(J,K) must be contigues
SC13R	NAFF(30)	NAFF(J) number of sectors interacting with sector J through radiation
SC14R	KAFF(30,13)	KAFF(J,K): index of the K^{TH} sector interacting through radiation with sector J.

Tab. 3.2: Parameters to be provided in BLØCK DATA only in the case of 1/3 of the whole 12-rod bundle.

3.3 Listing

Many modifications have been introduced in the listing produced by the code. The main modifications regard the following items:

a) View factors information : at the beginning of each axial portion the computed view factors are printed together with their sum and the result of the normalization.

b) The subchannels variables can be printed by means of a new method using the three subroutines JELLA, JELLB, JELLC (cfr. also CØMMØN Block SCO2P in 3.2).

The subroutines JELLA supply the following results:

CHA Channel index

ROD rod index

QGEN Heat generated in the portion of pin corresponding to that sector /W/.

QFLUX Convective heat flux, also $\frac{\text{QGEN + QJ}}{\text{AREA}}$ /W cm^{-2}/

QRAD Heat exchanged through radiation /W/.
QRAD > O means heat transmitted to the sector.

NU Nusselt number: it corresponds to the final value, i.e. it is modified by spacer effect and inlet effect (if any).

YH Spacer effect: NU = YH * NUO

ALFA Heat transfer coefficient based on NU. /W $cm^{-2}C^{-1}$/

TBULK Subchannel bulk temperature /^{o}C/

TBR Subchannel reference temperature /^{o}C/; as described in 5.5 of /3/ it is:
QFLUX = ALFA * (TWINF - TBR)

TWINF Sector wall temperature at the infinite conductivity of the canning material /^{o}C/.

TWALL Temperature at the position where the thermocouple is placed (also TWALL is corrected for the Biot effect, cfr. 5.8 in /3/, and for the conduction in radial direction cfr. 5.9 in /3/).

The same results are printed for the shroud sectors; in this case YH is always set to 1 because no spacer effect on the Nu number of the shroud is considered.

The subroutine JELLAC supplies the following results:

CHA	channel index
RØD	rod index
QTRA	Heat transmitted to the subchannel by the adjacent pins and shroud /W/.
MASS	Mass flow rate in the subchannel, average on the axial section /g sec^{-1}/
CP	Specific heat of the coolant /J g^{-1} C^{-1}/
T2-T1	Temperature increase in the coolant between the inlet and the outlet of the axial section /^{o}C/
LAMDA	Subchannel friction factor (4 times the Fanning friction factor).
DP	Pressure loss in the subchannel /Bar/
REB	Subchannel Reynold's number
DENSITY	Density of the coolant /g cm^{-3}/
TH.CØND.	Thermal conductivity of the coolant /W cm^{-1} $^{o}C^{-1}$/
ETA	Dynamic viscosity of the coolant /g cm^{-1} sec^{-1}/.

Subroutine JELLC supplies the following data:

CHA	channel index
RØD	rod index
H+B	h^+ evaluated at the bulk temperature of the coolant
H+W	h^+ evaluated at the wall temperature
RSM	$R(h_W^+)$ in case of hydraulically smooth flow
RO1	Parameter R_{01} (h_W^+)
RH+	Parameter R (h_W^+)
GO1	Parameter G_{01} (h_W^+)
GH+	Parameter G (h_W^+)
TW/TB	Wall to bulk temperature ratio
TW/TE	Ratio between wall and inlet temperature
BIØT	Biot number (cfr. 5.8 in /3/).

These subroutines are called only in case of heated sections. Once more, JELLC is called only in case of rough surfaces.

c) At the end of the calculation some data referred to the general heat balance are printed.

3.4 RESULTS*4

To allow the generation of a graphical output, the results of SAGAPØ-A (wall temperatures and pressure drop) are stored in 4 files on a disk. The plot procedure, which also allows the comparison with the experimental results, is described in Ref. /5/.

The four files are written in free format and contain index, axial position and value to be plotted. A description of these files is shown in Tab.3.3; for more information cfr. ref. /5/.

File	Format	Contents	Note
FTO2FOO1	free	X,DP	DP = pressure drop at heigh x
FTO3FOO1	free	NS,M,X,T	T = wall temperature of the sector of pin adjacent to subchannel M of channel NS at the height X
FTO4FOO1	free	NW,M,X,T	T = wall temperature of the sector of liner adjacent to subchannel M of wall channel NW at the height X
FTO9FOO1	free	NW,M,J,X,T	T = wall temperature of the surface adjacent to the part J of subchannel M of wall channel NW at the height X

Tab. 3.3: Output files RESULTS*4

3.5 RESTART

This procedure was already present in the earlier version of the code. If the given calculation time is almost used up, the cards with the mass flow rate - and temperature distribution are punched. This procedure allows the restart of the calculation. Because this procedure was not modified, the description given in Ref. /1/ is still valid and no more details will be presented here.

4. Thermal conduction model

4.1 Conduction within the pins

As previously pointed out, two conduction models are available: the first one (one-dimensional model) is organized by the subroutine LEIST, while the calculations for the two-dimensional model are performed by the subroutine BØTH.

In fig.4.1 the flow chart of the subroutinte LEIST is shown. The subroutines ALFAC and FGEØ determine the heat transfer coefficients (in the form requested by the following part of the code) and some geometrical factors. Then the coefficient matrix is built up and the resulting system of linear equations is solved by the subroutine GAUSS1. GAUSS1 solves the system of linear equations by means of the Gauss elimination method with pivotal condensation /8/.

TNEW assignes the computed temperature values to the arrays used in the other parts of the code and QCØC builds up an array (QJ) for the definition of the modified convective heat fluxes. In fact, the convective heat fluxes are defined as

$$\ddot{q} = \frac{q_{GEN} + QJ}{AREA}$$

At the first iteration is QJ = 0.0 (cfr. 2.2 and fig.2.12).
In the following iterations, QJ is defined by QCØC and $\ddot{q}$ results a function of radiation and conduction as well. QCØC also performs the control on the thermal balance.

In case of two-dimensional model, as shown in fig.4.2, together with ALFAC and FGEØ, also the subroutine KGAP is called, which computes the heat transfer coefficient at the gap between core and clad.

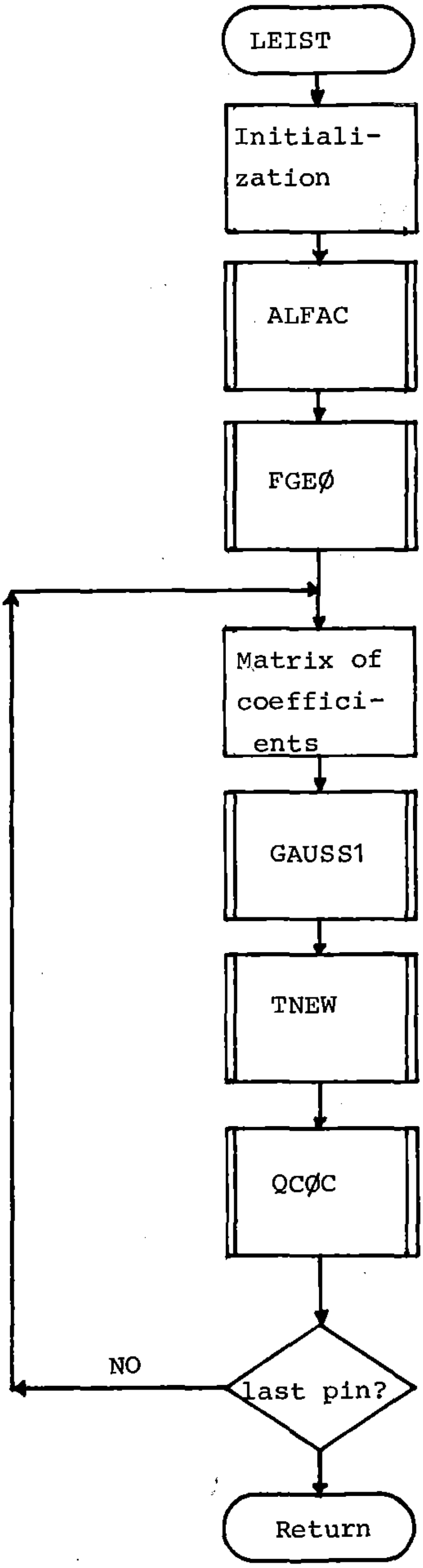


Fig.4.1: Flow chart of the subroutine LEIST

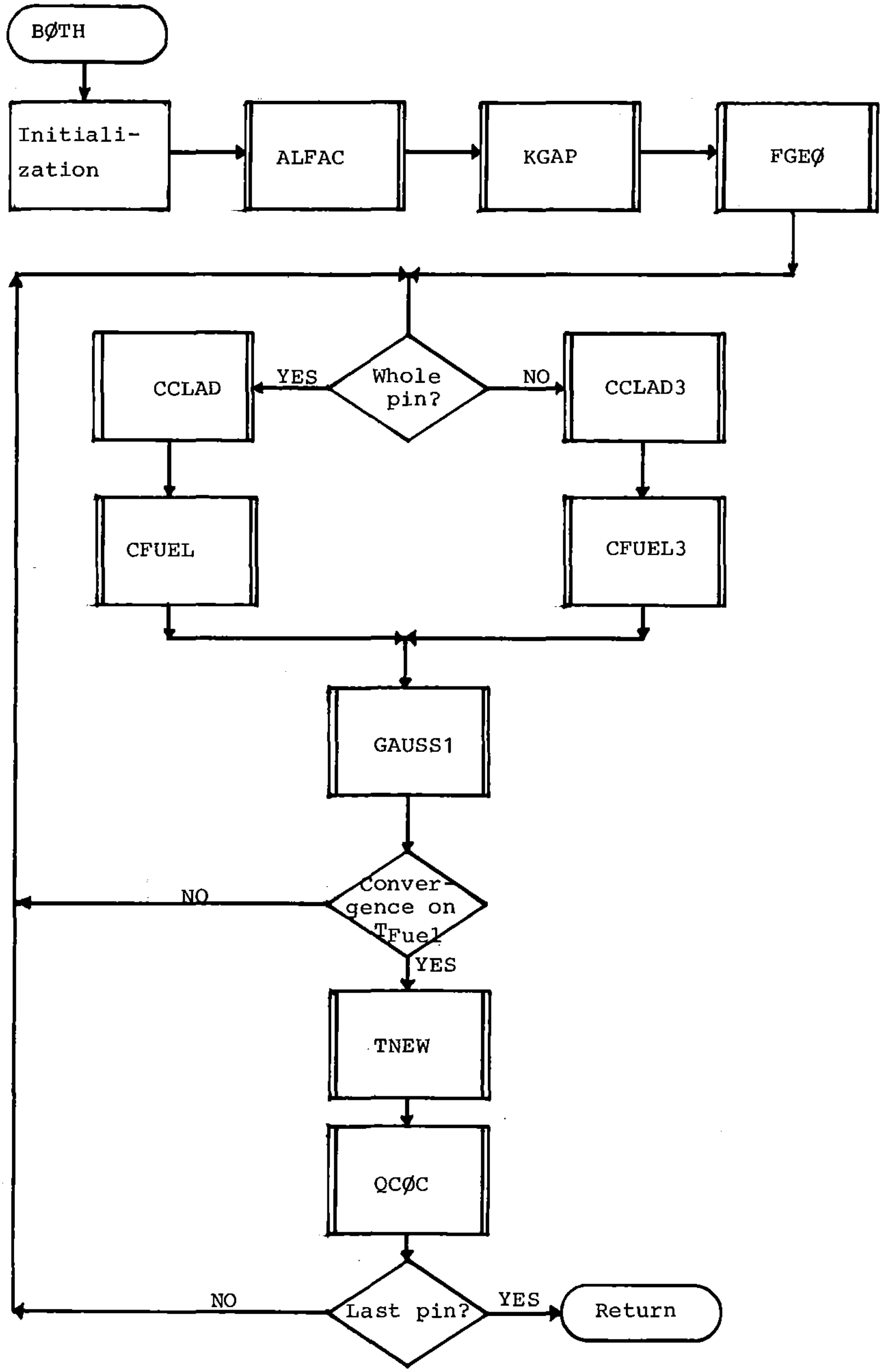


Fig. 4.2: Flow chart of the subroutine BØTH

The coefficient matrix is built up by the subroutines CCLAD (coefficient for the clad) and CFUEL (coefficient for the heated core). When the calculation is limited to a symmetry section of the whole bundle, for some of the pins the calculation is also restricted to a symmetry section of the pin. In this case the subroutines CCLAD3 and CFUEL3 are called.

Then the subroutine GAUSS1 solves the system and the convergence on the core temperature is tested. When the convergence is reached, the calculation goes on with TNEW and QCØC.

4.2 Conduction in the shroud

The calculation of the thermal conduction within the wrapper tube is organized by the subroutine CØNSHR (see fig. 4.3).

As pointed out in 5.3 of /3/, the shroud is considered to consist of a ring, whose dimension must be given in BLØCK DATA (cfr. CØMMØN block /SCISC/), together with the angles corresponding to a half wall sector and to a whole corner sector.

The coefficient array is built up by the subroutine MATBUS and the system is solved by the subroutine GAUSS. GAUSS is a slight modification of the subroutine GAUSS1, the difference consisting in the method used to transfer the matrix of the coefficients.

Subroutine TNEWS assignes the computed values to the temperature array and QDEFIS computes the array for the definition of the modified convective heat fluxes (in analogy to QCØC in 4.1). QDEFIS performs also the control of the thermal balance.

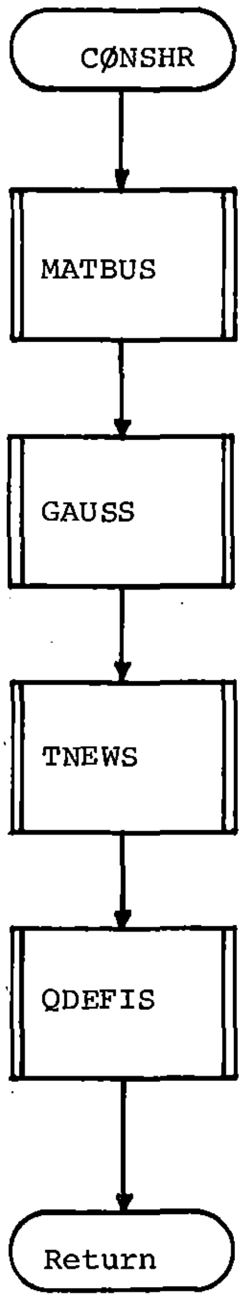


Fig. 4.3: Flow chart of the subroutine CØNSHR

4.3 New subroutines and modifications

The subroutines concerned with the conduction problem are:

LEIST, ALFAC, JZURU, KGAP, CCLAD, FGEØ, KFUEL, CFUEL,TESTNU, TNEW, CCLAD3, CFUEL3, QCØC, TESTB, TESTW, ASSE, TEST1, MEZZI, TTØT, TØRG, BØTH, CØNSHR, MATBUS, TNEWS, QDEFIS, GAUSS, PIVØT, GAUSS1, PIVØT1, DELIP, TBRTBS.

To allow the insertion of the conduction model the following subroutines have been modified:

MAIN, ANGCA1, BALA, CEWA, RECANG, RECCA1, RECCA2, SUBBAL, TRICA1, WALLTE.

5. Radiation model

5.1 General information

The radiation calculation is subdivided into two parts: the determination of the connexion arrays (which sector exchanges with which etc.) and the solution of the problem (determination of the heat transferred by radiation).

For the first purpose, five subroutines are called by the main program at the beginning of each axial portion.

TØTSEC: computes the total number of sectors.

TARRAY: builds up the arrays (ISU,IGI,ISS) which allow the interaction between the flow model and the radiation model.

VFXAL: computes the view-factors

VFDET: determines the sectors exchanging with each other the relative view-factor type.

VFCTR: controls the produced arrays

The problem is then solved by the subroutine RADIA (see fig. 5.1): within it, MATBUI builds up the array of coefficients and SYSØL solves the system of linear equations by means of the Gauss-Seidel method /6/. For the solution of the radiation system, the Gauss-Seidel iterative method has been preferred to the Gauss elimination method because of the sensible decrease of storage requirement allowed by the iterative method.

It must be pointed out, that if EPSR ≠ EPSS (see later 4.2) the sufficient convergence condition for the Gauss-Seidel method is not fulfilled: in this case you will get a warning on the listing, but the calculation will nevertheless be performed by means of the Gauss-Seidel method: in all the calculations performed up to now no convergence problems occurred also if the sufficient convergence condition was not fulfilled. QDEFI builds up the arrays QPR, QSR (heat exchanged by radiation for pin and shroud sectors) and performs the thermal balance.

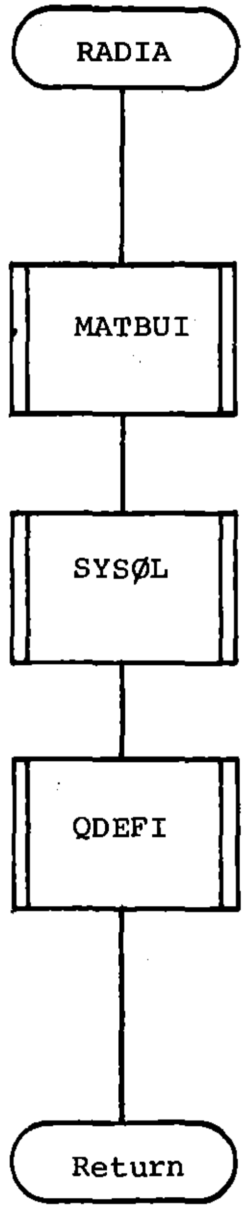


Fig. 5.1: Flow chart of the subroutine RADIA

For the solution of the radiation problem, the emissivity must be provided (see later 5.2) and the option ICØMP must be defined in BLØCK data (see 3.2). The option ICØMP is concerned with the problems discussed in 4.3.2 of ref. /3/. No great differences between the two approaches have occured up to now in the SAGAPØ-A calculations; however, the analysis of the radiation exchange performed on a simplified model shows that slightly better results are obtained by putting ICØMP=0 (normalization of all the computed view factors).

It must be pointed out that a radiation calculation for a symmetry section (1/2, 1/3,... etc. of the whole bundle) is not possible, because in this case the reciprocity rule for the view factors is nolonger fulfilled.

However, a radiation model for 1/3 of the 12-rod bundle has been developed, because, due to the particular geometry of the 12-rod bundle, the error introduced by this schematization should be relatively small.

5.2 Emissivity

The emissivity is the parameter defining the radiative exchange. It depends on many factors, mainly on the temperature of the surfaces. In SAGAPØ-A two different methods are allowed to define the emissivity value to be used, depending on the option IEPS (cfr. 3.2):

IEPS=0: the emissivity is assumed to be constant for the pins and for the shroud. In this case no dependence of the emissivity on the temperature (i.e. on the axial position of the surface) is considered. The constant values for the pin emissivity (EPSR) and for the liner emissivity (EPSS) must be defined in BLØCK DATA.

IEPS=1: The emissivity is assumed to depend on the temperature. In this case a function EPS must be defined, which provides the necessary correlation. If IEPS=1 the values EPSR and EPSS are not used at all. On the contrary, if IEPS=0 the function EPS is not used.

Due to the difficulty of determining the emissivity, both methods are approximated. The choice between the two methods is therefore mainly depending on operative problems (availability of data on the bundle to be analyzed, calculation at different emissivity values to evaluate the radiation effect, etc). In most of the performed calculations for the 12-rod bundle it was assumed IEPS=0 by setting

EPSR=0.6 EPSS=0.3.

while in the calculation for the 19-rod bundle it was generally assumed

EPSR=ESPS=0.42

5.3 New subroutines and modifications

The radiation model consists of the following subroutines:

RADIA, TARRAY, VFCAL, SYSØL, SUFCØN, GAUDEL, MATBUI, VECTR, QDEFI, NØRMA, DAREA, VFDET, VFDE1, VFD3, EPS, CFC1,..., FINDF1... .

To allow the insertion of this model the MAIN program has been modified.

6. Other modifications

6.1 Spacers effect on Nu number

A new method has been developed for the evaluation of this effect. It is based on the last experiment performed at the KfK in this field /7/.

The axial bundle subdivision has been left as it was in the earlier version of the code (also depending on the presence of the spacer), but the spacer effect is now computed at each section, inespective of the presence of a spacer grid. This is performed by calling the subroutine SPANU, which in turn calls SPANUG (in case of smooth surfaces) or SPANUR (rough surfaces). The subroutines SPANUG und SPANUR are a slight modification of the subroutines used to elaborate the data presented in /7/ and are due to the courtesy of Md.A. Hassan.

To allow the insertion of this new method, modifications have been performed on the following subroutines:
MAIN, CEWA, RECANG, WALLTE, RTRI, RTSI, TEMLAM.

6.2 Laminar flow

As pointed out in 3.3.2 of /3/ in case of laminar flow the corner channel and the adjacent parts of wall channels are combined together for the calculation of pressure loss. This procedure consists of a modification of the K factors (eq. 3.20 in /3/) and is performed by the subroutine CØRKA (see later 7.11).

To allow the use of this method, the subroutine KAPCØR has been modified.

6.3 Superposition principle

The superposition principle is used in two different cases:

1. Determination of the temperature of a pin sector facing the liner and of the temperature of the correspondent liner sector (cfr. 5.5 in /3/). In this case the superposition principle is switched on by the option IS (cfr. SC09C in 3.2). As already pointed out (2.3) some "dummy" gas temperatures are defined by the subroutine TBRTBS.

2. Determination of the temperature of the sectors facing a central channel (only in case of rough surfaces and turbulent flow) /9/. This method is switched on by the option ISUC (cfr. SC34C in 3.2). The calculations are performed by the subroutine SUPCEN, and the subroutine TBRTBS is also used.

7. New subroutines

In the present chapter a short description of the new inserted subroutines and functions is presented together with the explanation of their arguments. No description of the subroutines and functions contained in the earlier version of the code is given here.

7.1 Subroutine ALFAC

Builds up the array FALFA (42,3), to be used as boundary condition for the conduction equations.
ALFAC is called by LEIST and BØTH.

Arguments:
VDIAM: volumetric diameter of the pin (cm)
H : length of the axial section (cm)

7.2 Subroutine ASSE

Attributes a start value to the temperatures within the heated core (fuel).
ASSE is called by BØTH.

Arguments:

NCA : number of sectors in the actual pin
XF : temperatures.

7.3 Subroutine BØTH

BØTH organizes the calculation of the thermal conduction within the pins in the case of power generated within a heated core (two-dimensional approach to the problem). BØTH is called by the subroutine TØRG.

Arguments:

VDIAM: volumetric diameter of the pin (cm)
PIG : 3.1415.
H : length of the axial section (cm)
VDIA1: Tip or volumetric diameter of the pin (cm), depending on the option LAMØP3.
RINT: inner radius of the clad (cm).

The flow chart of this subroutine is presented in fig. 4.2.

7.4 Subroutine CCLAD

CCLAD computes the coefficients for the thermal conduction equations within the clad in case of power generated within a inner core (two-dimensional model).

CCLAD is called by BØTH.

Arguments:

NCA : number of sectors in the considered pin
A : array containing the a_{ij} coefficients (left hand six of the equations).
B : vector for the b_i coefficients (right hand side of the equations).
RSTAR : reference radius for the thermal exchange across the gap (cm). Cfr. 5.2.1 in /3/.
R : average radius of the clad (cm)
H : length of the axial section (cm)
J : index of the considered pin
S : clad thickness (cm)

7.5 Subroutine CCLAD3

CCLAD3 computes the coefficients for the thermal conduction equation within the clad when the two-dimensional model is used for a half of a pin. It is called by BØTH. The arguments are the same as in the case of CCLAD.

7.6 Subroutine CFUEL

CFUEL computes the coefficients for the thermal conduction equations within the core. It is called by BØTH.

Arguments:

NCA : number of sectors in the pin J
A : array of the coefficients $a_{i,k}$ (left hand side of the equations)
B : vector of coefficients b_i (right hand side of the equations)
RSTAR: reference radius (cm) cfr. 5.2.1 in /3/.
H : length of the axial section (cm)
J : index of the considered pin
XF : vector containing the temperatures within the heated core (fuel).

7.7 Subroutine CFUEL3

CFUEL3 is the analog to CFUEL (see) in the case when only a half of a pin is computed.

7.8 Subroutines CFC1, CFC3,...

CFC1 computes the view factor of type F1 (pg. 134 in ref. /3/). Analogously CFC3 computes the view factor of type F3 (pg. 135 in /3/) and so on. Because of the similarity of these subroutines, no individual description will be presented. They all are called by the subroutine VFCAL.

Arguments:

P : pitch of the rods (cm)
D : volumetric diameter (cm)
PIG : 3.1415
FCX : view factor corresponding to the numerator of definition 4.27 in /3/:i.e.

$$FCX = f_{ij} * 2A_i = f_{ji} * 2A_j$$

where A_i is the surface of sector i.
R = volumetric radius (cm)
Z = distance between the center of a wall pin and the shroud wall (cm).
ZWC = height of the blocking triangles (cm).

7.9 Subroutine CØNSHR

Organizes the calculation of the thermal conduction within the shroud. It is called by TEMCØN and has no arguments. The flow chart is presented in fig. 4.3.

7.10 Subroutine CØNTRØ

Prints information about the convergence process. It is called by the MAIN program.

Arguments:

FA : friction factor in the precedent iteration
FA1 : friction factor in the present iteration
ITCØRR: index of the ITCØRR loop (cfr. /1/)
INDICE: convergence flag for the thermal calculation.

7.11 Subroutine CØRKA

CØRKA modifies the computed K-values (for the calculation of the friction factor in laminar flow) if the corner channel and the adjacent part of wall channel are computed together (case of JLAM=1) according to the procedure described in 3.3.2 of /3/. CØRKA is called by KAPCØR and has no arguments.

7.12 Function DAREA

The function DAREA computes the area of the sector L. To allow its direct use in the equation (4.27) of /3/, it supplies the double of this area, i.e.:

DAREA(L) = 2 * Area of the sector L.

DAREA is called by MATBUI, VFCTR, NØRMA and has the following argument

L : index of the sector.

7.13 Subroutine DECP

DECP fixes the switch IPRINT, which determines the print (IPRINT=1) or not (IPRINT=0) of the thermo- fluiddynamic variables at the actual axial section. The value of IPRINT depends on the input data NPRINT and AR (see CØMMØN/SC25C/ in 3.2).
DECP is called by the MAIN program.

Arguments:

IPRINT: = 1 The results for the actual axial section are printed on the output listing

= 0 no print

X1 : distance between the beginning of the actual axial section and the point where the calculation is started (cm)

X2 : distance between the end of the actual axial section and the point where the calculation is started (cm).

STLEN: Distance between the bundle inlet and the point where the calculation is started (cm).

7.14 Subroutine DELIP

DELIP builds up the array LIPS (19,10) and the vector NCAN (19):

NCAN(J): number of channels adjacent to the J^{th} pin, J = 1, NRØDS

LIPS(J,I): index of the I^{th} channel adjacent to the pin J, I=1, NCAN(J)

DELIP is called by the MAIN program and has no arguments.

7.15 Function DPIN

DPIN represents the coefficient for the pressure loss at the bundle inlet (cfr. equation 6.9 in /3/).
It is called by the MAIN program.

Arguments:

MFLØW: mass flow rate relative to the whole bundle (g/sec^{-1})

PE : inlet pressure ($kg\ cm^{-2}$)

TE : inlet temperature (^{o}C)

CINL : limit value for DPIN at high turbulent flow: 1.2.

7.16 Function EPS

EPS represents the emissivity of the sector L. The values of EPS are computed depending on the option IEPS as described in 5.2. EPS is called by the subroutine MATBUI.

Arguments:

T : temperature of the sector L (^{o}C)
L : index of the sector for which the emissivity has to be determined.

7.17 Subroutine FGEØ

FGEØ computes the average radius of the clad (R), his thickness (S) and the reference radius RSTAR (cfr. 4.1). It is called by LEIST and BØTH.

Arguments:

VDIAM: volumetric diameter of the considered pin (cm)
RSTAR: reference radius (cm)
S : clad thickness (cm)
R : average radius of the clad (cm)
RINT : inner radius of the clad (cm)

7.18 Subroutines FIND ...

The subroutines of this group fill the arrays KAFF(132,13),VFAC(132,13):

KAFF (L,N): index of the N^{th} sector having a radiation exchange with the sector L
VFAC (L,N): view factor $f_{L,N}$

There is a subroutine of this type for each type of view factor, as it is shown in Tab. 7.1.
All these subroutines are called by VFDET.

Arguments:

L : index of the sector
NAFF : counter of the sectors interacting with sector L through radiation
NS : index of the channel adjacent to the sector L
J : index of the rod to which the sector L belongs.

Subroutine	Detects the view factor(s) of type
FINDF1	F1
FINDF3	F3,F15
FINDF4	F4
FINDF5	F5
FINDF9	F9
FINDW2	F2
FW7	F7
FW8	F8
FW1112	F11,F12
FW13	F13
FW16	F16
FW1718	F17,F18
FINDA	F24
FS1112	F11,F12 *
FFS13	F13 *
FFS24	F24 *
FSA	F17,F18 *
FF32	F32
FFW31	F31 (relative to wall sectors)
FFA31	F31 (relative to corner sectors)
FS2	F15 *

Tab. 7.1: Reference table for the subroutines of type FIND....

* relative to shroud sectors.

7.19 Subroutine GAUDEL

This subroutine solves the system of linear equations for the radiation problem using the GAUSS-SEIDEL method /6/. In order to reduce the requested storage amount, the coefficient matrix has been compressed from A(132,132) to A(132,13) (case of 19-rod bundle) by means of a particular index manipulation. The subroutine GAUDEL has been written to solve this particular system of equation. It cannot be used for general purpose. The accuracy of the convergence process and the maximum number of allowed iteration are fixed in the calling subroutine, SYSØL.

Arguments:

A	: matrix of coefficients (left hand side of the equations)
B	: vector of coefficients (right hand side of the equations)
X	: solution of the system: X(L) is the net heat quantity (cal/sec) exchanged through radiation at the sector L.
PERC	: accuracy of the calculation
ITMAX	: maximum allowed number of iterations.

7.20 Subroutine GAUSS

GAUSS solves the system of linear equations generated by the thermal conduction within the wrapper tube by means of the Gauss elimination method with pivotal condensation. It is called by the subroutine CØNSHR and it is written to solve this particular problem (it cannot be used for general purpose).

Arguments:

A	: array containing the coefficients for the left hand side of the equations
B	: vector containing the coefficients for the right hand side of the equations
X	: vector containing the computed temperatures.

7.21 Subroutine GAUSS1

GAUSS1 is a general purpose subroutine suitable to solve a system of linear equations by means of the Gauss elimination method with pivotal condensation. It is used to solve the equations for the thermal conduction within the pins and is called by LEIST or by BØTH.

Arguments:

A : array containing the coefficients for the left hand side of the equations

B : vector containing the coefficients for the right hand side of the equations

X : vector containing the computed temperatures

NTØT : number of equations

N1 : first dimension in the DIMENSIØN statement for A in the calling subroutine; i.e.: if in the calling subroutine it is DIMENSIØN A (6,7)
Then N1=6.

7.22 Subroutine HEATBA

HEATBA checks the heat balance for the whole bundle. At each axial section the heat transmitted to the gas from the pins and from the shroud is computed for each subchannel. HEATBA is called by the MAIN program and has following arguments:

IND: =-1 initialization of the arrays.
= 0 the arrays are filled up
= 1 print the results

PBT : average gas pressure

INDQ: cfr. pg. 25 in /1/

TE : inlet gas temperature

MFLØW: mass flow rate.

7.23 Subroutine JELLA

JELLA prints the subchannel heat transfer results at each axial section (cfr. 3.3). JELLA is called by the MAIN program and has the following arguments:

JL : index which establishes what results are to be written:
JL<O the results for all subchannel are printed
JL>O the results of the subchannel adjacent to pin JL are printed; also, if JL>O it must be

$$1 \leq JL \leq NRØDS.$$

JELLA is called only at the heated sections.

7.24 Subroutine JELLB

JELLB prints the results relative to the gas conditions in the different subchannels (cfr. 3.3).
It is called by the MAIN program and has the following arguments:

JL : has the same meaning as in JELLA (cfr. 7.23)
PBT : Average pressure of the gas at the present axial section.

JELLB is called only in case of heated sections.

7.25 Subroutine JELLC

JELLC prints information about the roughness parameters in the different subchannels at each axial section (only in the case of rough heated part and turbulent flow). It is called by the MAIN program and has the following arguments:

JL : cfr. 7.2.3
TE : gas bulk temperature at the bundle inlet

TE1 : gas bulk temperature at the point where the calculation is started

BIØT : Biot number

7.26 Subroutine JZURU

JZURU builds the array JZUR(19,42), used for the conduction calculation. It is called by the MAIN program and has no arguments.

JZURU(J,NS) is the index M of the subchannel of the channel NS adjacent to the pin J.

7.27 Real Function KFUEL

KFUEL computes the thermal conductivity of the heated pin core (cal sec^{-1} cm^{-1} C^{-1}) as function of the temperature.

It is called by CFUEL and CFUEL3 and has as argument the temperature of the actual sector of heated core.

7.28 Real Function KGAP

KGAP determines the heat transfer coefficient at the gap between the heated core and the clad material (cal sec^{-1} cm^{-2} C^{-1}). It is called by CFUEL and CFUEL3 and has no arguments.

7.29 Subroutine LEIST

LEIST organizes the calculation for the thermal conduction within the pins in case of heated clad (one-dimensional model). In fig. 4.1 the flow chart of LEIST is presented. It is called by the subroutine TØRG and has the following arguments:

VDIA : volumetric diameter of the pins
PIG : 3.141593
H : height of the axial section
VDIA1 : For laminar calculation only, tip diameter of the pins.
RINT : inner radius of the clad

7.30 Subroutine LINPØW

LINPØW fills the matrix SHQ(18,2) with the initial values of the power generated in the shroud. It is called by the MAIN program and has no arguments.

7.31 Subroutine MATBUI

MATBUI builds the array of the coefficients for the radiation calculations. It is called by the subroutine RADIA and has the following arguments:

A : matrix of coefficients for the left hand side of the radiation equations

B : vector of coefficients for the right hand side of the radiation equations.

7.32 Subroutine MATBUS

MATBUS builds the array of coefficients for the calculation of the thermal conduction within the wrapper tube. It is called by the subroutine CØNSHR and has the following arguments:

A : array of coefficients for the left hand side of the equations.
B : vector of coefficients for the right hand side of the equations.

7.33 Subroutine MEZZI

The subroutine MEZZI determines the arrays GEØ1(42,3) and GEØ(42,3); they represent the fraction of the whole pin corresponding to the correspondent sector: for istance, in case of calculations referred to the whole bundle, it is:

Type of adjacent channel	GEØ1	GEØ
CENTRAL	6	$\pi/6$
WALL	4	$\pi/4$
CORNER	6	$\pi/6$

MEZZI is called by the MAIN program and has following arguments:

NRØDS : total number of rods
NSEL : index for the considered symmetry section (1 = whole 19-rod bundle, 2 = 1/2 of the whole 19-rod bundle, 3 = 1/12 of the whole 19-rod bundle)
NSTØT : total number of channels.

7.34 Subroutine NØRMA

Performs the normalization of the computed view factors depending on the option ICØMP (cfr. 4.3.2 in /3/ and CØMMØN /SC16R/ in 3.2 of this paper). NØRMA is called by the subroutine VFCTR and has the following arguments:

TØT : sum of the computed view factors for sector L (in input, before the normalization, in output after it)
L : considered sector

7.35 Subroutine PIVØT

PIVØT finds the pivotal element for the solution of the thermal conduction equation within the shroud with the Gauss elimination method. It is called by subroutine GAUSS and has following arguments:

A : array of coefficients for the left hand side of the equations
B : vector of coefficients for the right hand side of the equations
N : index of the row reached by the calculation during the elimination process at the moment when PIVØT is called

7.36 Subroutine PIVØT1

This subroutine is equivalent to the subroutine PIVØT, but it can be used for general purpose. It is called by the subroutine GAUSS1 and has the same arguments as PIVØT plus:

NTØT: number of equations in the system
N1 : first number for the size of the array A(N1,N2) in the calling subroutine

7.37 Subroutine QCØC

QCØC builds the array QJ(19,42) /W/ to define the convective heat transferred at the sector corresponding to J and NS, i.e.

$$\ddot{q}_{CONVECTION(J,NS)} * Area_{(J,NS)} = q_{GENERATED(J,NS)} + QJ_{(J,NS)}$$

Furthermore QCØC controls the heat balance for the pin J after the conduction calculation.
QCØC is called by LEIST and BØTH and has following arguments:

NCA : number of sector in the pin J
J : index of the considered pin
X : vector containing the computed wall temperatures
NSE : dimension of the vector X in the calling subroutine

7.38 Subroutine QDEFI

QDEFI builds the arrays QPR(NS,M), heat transferred by radiation at the correspondent pin sector (cal/sec), and QSR(NSW,M), heat transferred by radiation at the correspondent shroud sector (cal/sec), where

NSW = NS-NSTR,

NSTR = number of central type channels.
QDEFI is called by the subroutine , RADIA and as the following argument:
QJR : vector, of length NSECT, where are stored the values of the heat transferred through radiation (cal/sec)

7.39 Subroutine QDEFIS

QDEFIS builds the array SHQC(18,2) after the calculation of the thermal conduction within the shroud. SHQC is analog to the array QJ for the conduction within the pins (cfr. 7.37). QDEFIS controls also the thermal balance. It is called by CØNSHR and has no arguments.

7.40 Subroutine RADIA

RADIA organizes the calculation for the thermal radiation. It's flow chart is presented in fig. 5.1. RADIA is called by the subroutine TEMCØN and has no arguments.

7.41 Subroutine SPANU

SPANU organizes the calculation of the spacer effect on the Nu number of the rods (cfr. 5.7 in /3/).

SPANU is called by the subroutines CEWA, BALA, RTRI, RTSI, RECANG, TEMLAM.

It has following arguments:

R1 : Reynolds number of the considered subchannel
P1 : Prandtl number of the considered subchannel
NS : channel index
M : subchannel index
Y : factor $\frac{Nu}{Nu_o}$

7.42 Subroutine SPANUG

SPANUG computes the effect of the spacer on the Nu number in case of smooth surfaces

$$y = Nu/Nu_o$$

where Nu_o is the undisturbed Nu number. It is called by SPANU and has the following arguments:

X : dimensionless axial distance between the considered point and the begin of the spacer (distance referred to a reference hydraulic diameter: if IHAS=0, then the hydraulic diameter of the whole bundle is used; IHAS=1, the diameter of the considered channel is used. cfr. IHAS in 3.2)
Y : factor Nu/Nu_o
RE : Reynolds number, also depending on the option IHAS
PR : Prandtl number, also depending on IHAS
AL : dimensionless axial width of the spacer (also depending on IHAS)
E : Blockage area of the considered subchannel

7.43 Subroutine SPANUR

SPANUR is perfectly analogous to the subroutine SPANUG, but it is called in case of rough surfaces.

7.44 Subroutine SUFCØN

SUFCØN tests the sufficient condition for the convergence of the Gauss-Seidel Method /6/ by the solution of the radiation equations (cfr. 5.1). It is called by SYSØL and has the following arguments:

A : array of the coefficients for the left hand side of the equations

INDEX:=0 the array A satisfies the sufficient convergence condition
>0 the condition is not respected.

7.45 Subroutine SUPCEN

In SUPCEN is applied the superposition principle in the determination of the wall temperatures of the sectors adjacent to a central channel or to a central part of wall subchannel. The subroutine SUPCEN is called by RTRI only in case of turbulent flow. It has the following arguments:

RH : height of the roughness
PBT : average pressure of the coolant
TWALL : wall temperature of the sector
LAM1 : friction factor of the adjacent subchannel
U1DÜ : cfr. pg. 129 in /1/
YYI : factor for the Nu number improvement due to the spacers (cfr. SPANU)
ITYP : type of subchannel
1: central part of a wall subchannel
2: wall
3: corner

KI : thermal conductivity of the coolant
DEI : hydraulic diameter of the subchannel
FACHE : correction factor which takes into account the inlet effect on Nu number
QA : heat flux at the sector (cal sec^{-1} cm^{-2})
TW1 : surface pin temperature (work area)
G1A : Parameter $G(h^{+})$, by supposing unheated adjacent pins
NU11: : Nu number, by supposing unheated adjacent pins
I : channel index
M : subchannel index
REI : Reynolds number
ITW1 : iteration index in the calling subroutine

7.46 Subroutine SYSØL

SYSØL organizes the solution of the radiation equations by means of the Gauss-Seidel iterative method. It is called by RADIA and has the following arguments:

A : array of the coefficients for the left hand side of the equations
B : vector of the coefficient for the right hand side of the equations
Y : vector for the computed results.

7.47 Subroutine TARRAY

Tarray prepares the connexion arrays for the radiation calculation:

ISU(L,1)=NS, index of the channel adjacent to the sector L
ISU(L,2)=M, index of the subchannel adjacent to the sector L
IGI(NS,M)=L, index of the sector of pin adjacent to subchannel M of channel NS.
ISS(NSW,M)=L, index of the sector of shroud adjacent to subchannel M of channel NS (NSW=NS-NSTR, for NSTR see 7.38).

TARRAY is called by the MAIN program and has no arguments

7.48 Subroutine TBRTBS

TBRTBS builds the arrays

TBR(NS,M): reference gas temperature for the pin in the subchannel NS,M

TBS(NSW,M): reference gas temperature for the shroud in the subchannel NS,M (for NSW see 7.38).

For the meaning of these temperatures see 5.5 in /3/. TBRTBS is called by the subroutine TEMCØN and has no arguments.

7.49 Subroutine TEMCØN

Subroutine TEMCØN organizes the heat transfer calculations (see flow chart in fig. 2.13). It is called by the MAIN program and has the following arguments.

IRH : =1 smooth section
=2 rough section

K : index of the axial section

RH : height of the roughness ribs

SUR : cfr. pg. 163 in /1/

D : volumetric diameter of the pins

TE1 : bulk temperature at the point where the calculation is startet

PBT : average pressure of the gas in the axial section

H : length of the axial section

V : volumetric or tip diameter of the pins (cfr. 3.2)

R : inner radius of the pins

INDICE: number of the sectors for which no convergence for the wall temperature has been reached

QTØT : total heat generated by the pins in this section

7.50 Subroutine TESTB

TESTB performs the convergence test on the bulk temperatures. It is called from the subroutine TEMCØN and has the following argument:

INDICE : =0 convergence has been reached for all subchannels
>0 no convergence for N=INDICE subchannels.

7.51 Subroutine TESTNU

TESTNU is called by the subroutine TEMCØN in order to average the heat transfer coefficients between two successive iterations in case of convergence problems. Such problems arise due to the spacer effect on the Nu number at low Reynolds numbers: at Re $\simeq$ 5000 the factor $Y=Nu/Nu_o$ (cfr. 7.42) has a maximum and a small variation of the Re number (i.e. of the mass flow rate of the channel) generates a great variation of the Y factor generating convergence problems.

Argument of the subroutine is the switch

IND = 0 the heat transfer coefficients are stored in a work area. IND=0 is the input for TESTNU at each iteration.

IND = 1 only in case of convergence problems: the heat transfer coefficients are averaged.

7.52 Subroutine TESTW

Performs the test of convergence on the wall temperatures of the pin and shroud sectors. It is called by TEMCØN and the argument, INDW, indicates the number of sector for which no convergence has been reached; also INDW=0 when the convergence is reached.

7.53 Subroutine TEST1

Performs the test of convergence on the temperatures within the heated core (fuel). It is called by the subroutine BØTH and has following arguments:

NCA : number of sectors in the pin
X : new value for the temperatures
XF : old value for the temperatures (in output, also new value)
KK : convergence index: number of sectors for which no convergence has been reached; also KK = 0 when the convergence is attained.

7.54 Subroutine TNEW

Because the wall temperatures within the conduction model are stored in the vector X(NSECT) while in the flow model they are stored in the array TW(NS,M), TNEW copies the contents of X into TW.
It is called from subroutines LEIST and BØTH and has the following arguments:

NCA : number of sectors in the pin
X : computed results for the wall temperatures of the sectors
J : pin index
NSE : Size of the array X in the calling subroutine.

7.55 Subroutine TNEWS

Is analogous to the subroutine TNEW for the LINER temperatures.
It is called by CØNSHR and has the following argument:

X : vector containing the computed wall temperature for the shroud.

7.56 Subroutine TØRG

Organizes the calculation of the conduction model (see flow chart in fig. 2.14). It is called by subroutine TEMCØN and has the following arguments:

VDIAM : volumetric diameter of the pin
RINT : inner radius of the clad
PIG : 3.141593
H : length of the axial section

7.57 Subroutine TØTSEC

Computes the total number of sectors of pin and liner. It is called by the MAIN program and has the argument NSEL (type of symmetry section, cfr. 7.33).

7.58 Subroutine TTØT

TTØT fills the array containing the bulk temperatures, TBØLD(NS,M) and the wall temperatures for the pins, TD(NS,M), and for the shroud TLD(NS,M), relative to the precedent iteration, to allow the further convergence test. It is called by the MAIN program and by the subroutine TEMCØN and has the following argument:

INDEX : = 1 the array TBØLD is filled
≠ 1 the array TD and TLD are filled.

7.59 Subroutine VFCAL

Organizes the calculation of the view factors for pin and shroud sectors. It is called by the MAIN program and has no arguments.

7.60 Subroutine VFCTR

Controls the view factors computed by VFCAL and the correspondence arrays (determined by VFDET). It is called by the main program and has no arguments.

7.61 Subroutine VFDET

Organize the determination of the correspondence arrays for the radiation calculations.
It is called by the MAIN program and has following arguments:

NSEL : index for the symmetry section (cfr. 7.38)
NRØDS: total number of rods.

7.62 Subroutine VFDE1

Determines the correspondence arrays:

NAFF(L) : number of sectors exchanging through radiation with sector
KAFF(L,K) : index L1 of the K^{TH} sector exchanging with sector L; $1 \leq K \leq NAFF(L)$
VFAC(L,K) : view factor $f_{L,L1}$, being L1=KAFF(L,K)

in case of whole bundle flow section. It is called by VFDET and has no arguments.

7.63 Subroutine VFD3

Analogous to VFED1 used in case of 1/3 of the whole 12-rod bundle (cfr. 5.1).

8. New CØMMØN Blocks

The new CØMMØNS blocks inserted in the program are listed in Tab. 8.1. For each of it the explanation of the arguments and the subroutine where the arguments are defined are given.

CØMMØN BLØCK	Argument(s)	Explanation	Defined in
SC01C	NCAN(19), LIPS(19,10)	NCAN(J): number of channels adjacent to the pin J LIPS(J,K): index of the K^{TH} channel adjacent to the pin J	DELIP or BLØCK DATA
SC02C	QJ(19,42)	QJ(J,NS): cfr. 7.37	QCØC
SC03C	NRØDS	total number of rods	MAIN
SC04C	RINT, RFUEL	RINT = inner radius of the clad RFUEL = radius of the heated core (fuel)	BLØCK DATA
SC05C	JZUR(19,42)	JZUR(J,NS): index M of the subchannel in channel NS adjacent to the pin J	
SC06C	D3(3)	coefficient for the thermal conductivity of the cladding material $K(T)=D3(1)+D3(2)T+D3(3)T^2$	BLØCK DATA
SC07C	H1	length of the axial section	MAIN
SC08D	TLD(18,2)	Work area for the shroud wall temperature	TTØT
SC09C	IS	Option for the superposition principle between wall pins and shroud (cfr. 3.2)	BLØCK DATA
SC10C	ANU(42,3)	ANU(NS,M): Nu number for the sector adjacent to subchannel M of channel NS	TELIN TEMLAM RTRI RTSI
SC11C	FALFA(42,3)	Modified heat transfer coefficient for the pin by NS,M	ALFAC
SC12C	GEØ(42,3)	geometrical factor (cfr. 7.33)	MEZZI
SC13C	GEØ1(42,3)	geometrical factor (cfr. 7.33)	MEZZI
SC14C	TBØLD(42,3)	work area for the bulk temperatures	TTØT MAIN

Tab. 8.1: New CØMMØN blocks

CØMMØN BLØCK	Argument(s)	Explanation	Defined in
SC15C	ALFA(42,3)	Heat transfer coefficient for the pin sector by NS,M	RTRI RTSI TEMLAM TELIN
SC16C	SNU(18,2)	Nu number for the shroud sector by NW (=NS-NSTR),M	TELIN TEMLAM
SC17C	SALFA(18,2)	Heat transfer coefficient for the shroud sector by NW(=NS-NSTR),M	TELIN TEMLAM
SC18C	RES,RIS,ANGW, ANGA	RES: outer radius for the shroud equivalent ring (cm) RIS: inner radius for the shroud equivalent ring (cm) ANGW: angle subtended by a wall shroud sector (R) ANGA: angle subtended by a corner shroud sector (R) (cfr. 3.2)	BLØCK DATA
SC19C	ICØN	Option for the heat conduction model	BLØCK DATA
SC20C	CGAP	Heat transfer coefficient at the gap between clad and fuel (cal sec^{-1} cm^{-1} C^{-1})	KGAP
SC21C	SHQC(18,2)	cfr. 7.39	QDEFIS
SC22C	NTØT	Total number of shroud sectors	BLØCK DATA
SC23C	ICS	Option for the conduction in the shroud (cfr. 3.2)	BLØCK DATA
SC25C	TBR(42,3) TBS(18,2)	Dummy gas temperatures (cfr.7.48)	TBRTBS

Tab. 8.1: Cont.

CØMMØN BLØCK	Argument(s)	Explanation	Defined in
SC30C	ANO(42,3), SNO(18,2)	work areas for the Nu numbers	TESTNU
SC32C	GHPIU(42,3)	Parameter $G(h_W^+)$	RTRI
SC33C	TWINF(42,3)	Wall temperature at the infinite conductivity of the cladding	MAIN
SC34C	ISUC	Option for the superposition principle in the central channels (cfr. 3.2)	BLØCK DATA
SC99C	TD(42,3)	Work area for the wall temperatures	TTØT
SC01L	STLEN	Distance between the bundle inlet and the point where the calculation is started	MAIN
SC02L	JLAM	Option for laminar calculations (cfr. 3.2)	BLØCK DATA
SC03L	IPHUD	Option to compute the shroud heat losses with the EIR Method (cfr. 3.2)	BLØCK DATA
SC04L	TAMB, HUDFAC, TMS	TAMB: room temperature during the experiment HUDFAC: heat transfer coefficient at the outer shroud surface TMS: average outer shroud surface temperature	BLØCK DATA
SC05L	PERLT	wetted perimeter for the whole bundle cross section	MAIN
SC06L	SHQ(18,2)	Heat generation in the shroud	LINPØW
SC07L	WSPØ,XM,NSPACT	WSPØ: spacers width XM : distance between the middle of the axial section and the bundle inlet NSPACT: total number of spacers	MAIN
SC08L	AGRI(42,3,7)	AGRI(NS,M,KS): blockage factor for subchannel M of channel NS at the KS^{TH} spacer	MAIN

Tab. 8.1: Cont.

CØMMØN BLØCK	Argument(s)	Explanation	Defined in
SCO9L	SDIS(7)	SDIS(KS): distance between the KSTH spacer and the bundle inlet	MAIN
SC1OL	RES,PRS,EPSS, DETØLA	RES: Re number for the whole bundle PRS: Pr number for the whole bundle EPSS: blockage factor of the spacer for the whole cross section DETØLA: hydraulic diameter of the whole bundle cross section	MAIN
SCO1P	S(19)	Factor taking into account systematic non-uniform heating of the rods: by NDVQ=1 it will be QPIN(J)=QPIN(1)*S(J) (cfr. pg.32 in /1/). For uniform heating, or NDVQ≠1, put S(J)= 1.0	BLØCK DATA
SCO2P	IPSUB	cfr. Tab. 3.1	BLØCK DATA
SCO1R	NSECT,NSECP	SECT: total number of sectors NSECP: total number of pin sectors	TØTSEC
SCO2R	P,D,Z,ZWC,H, LENGTH	P = pitch of the rods D = volumetric diameter of the rods Z = Distance between the center of the wall pin and the shroud ZWC = height of the blockage triangles H = length of the axial section LENGTH = length of the axial portion	MAIN
SCO3R	F1,F2,F3, ...,FI,...	View factor of type f_i	VFCAL
SCO4R	VFAC(132,13)	VFAC(L,K): view factor $f_{L,L1}$, being L1=KAFF(L,K)	VFDE1 VFD3
SCO5R	IRAD	Option for the radiation calculation (cfr. 3.2)	BLØCK DATA

Tab. 8.1: Cont.

CØMMØN BLØCK	Argument(s)	Explanation	Defined in
SCO6R	ISU(42,3)	cfr. 7.47	TARRAY
SCO7R	EPSR,EPSS SIGMA	ESPR: constant value for the pin emissivity (cfr. 5.2) EPSS: constant value for the shroud emissivity (cfr. 5.2) SIGMA:Stefan-Boltzmann constant (cfr. Tab. 3.1)	BLOCK DATA
SCO8R	QPR(42,3)	Heat transferred through radiation by the pin sectors (cfr. 7.38)	QDEFI
SCO9R	QSR(18,2)	Heat transferred by radiation at the shroud sectors (cfr.7.38)	QDEFI
SC1OR	QSTØT, QRTØT	QSTØT: Total heat generated in the shroud at the axial section QRTØT: total heat generated in the pins at the axial section	MAIN
SC11R	ISS(18,2)	cfr. 7.47	TARRAY
SC12R	IGI(42,3)	cfr. 7.47	TARRAY
SC13C	NAFF(132,13)	cfr. 7.62, 7.63, 3.2	VFDE1, VFD3, BLØCK DATA
SC14R	KAFF(132,13)	cfr. 7.62, 7.63, 3.2	VFDE1, VFD3, BLØCK DATA
SC15R	IEPS	Option for the emissivity model to be used (cfr. 5.2)	BLØCK DATA
SC16R	ICØMP	Option for the view factors normalization method to be used (cfr. Tab. 3.1)	BLØCK DATA
SCO1Z	YH(42,3)	YH(NS,M) = ratio $\frac{Nu}{Nu_o}$ for the sector of pin corresponding to the subchannel M of channel NS	RTRI, RTSI, TEMLAM

Tab. 8.1: Cont.

CØMMØN BLØCK	Argument(s)	Explanation	Defined in
CEVO1	LAMØP1	cfr. Tab. 3.1	BLØCK DATA
CEVO2	RTIP4	cfr. Tab. 3.1	BLØCK DATA
CEVO3	LAMØP2	cfr. Tab. 3.1	BLØCK DATA
CEVO4	LAMØP3	cfr. Tab. 3.1	BLØCK DATA

Tab. 8.1: Cont.

9. Recommendations to the user

1) If ICØN=0 no radiation calculation is allowed (also put IRAD=0)

2) In case of laminar flow, if you wish to use the superposition principle (also IS=1) and the shroud is unheated, a dummy power for the shroud (QLINMT, pg.33, and 31 in /1/) must be defined. The recommended value is

   QLINMT = 0.0007 W/cm

3) In case of laminar calculation, to avoid convergence problems by the spacers, it is recommended to "smooth" the blockage ratio of the spacers in the corner channels and the adjacents part of wall channels.

4) In case of laminar calculation, it is recommended to put

   JLAM = 1

5) The option IPHUD is concerned with a very particular problem. Put IPHUD = 0.0

6) By the radiation calculation it is recommended to put ICØMP=0.

7) No radiation calculation for symmetry sections (1/3 of the bundle etc.), with the exception of 1/3 of the 12-rod bundle, is allowed.

8) Always IPHUD = 0.

References

/1/ A. Martelli
SAGAPØ, a computer code for the thermo-fluiddynamic analysis of gas cooled fuel element bundles
KfK 2483, EUR 5510e, 1977

/2/ A. Martelli
Thermo- und fluiddynamische Analyse von gasgekühlten Brennelementbündeln
KfK 2436, EUR 5508d, 1977

/3/ S. Cevolani
Thermo- und fluiddynamische Analyse von gasgekühlten Brennelementbündeln bei Berücksichtigung von Wärmeleitung und Wärmestrahlung
KfK 3148, EUR 7051d, 1981

/4/ A. Martelli
SAGAPØ-2, an improved version of the SAGAPØ code for the thermo-fluiddynamic analysis of gas cooled fuel element bundles
KfK 2663, 1978

/5/ S. Cevolani
Procedure for the use of the code SAGAPØ-A and auxiliary programs.
KfK 3186, EUR 7055e, 1981

/6/ D.D. Mc Cracken, W.S. Dorn
Numerical methods and FØRTRAN programming
J. Wiley and Sons, New York, 1964

/7/ Md.A. Hassan
Wärmeübergang im Abstandshalterbereich gasgekühlter Stabbündel
KfK 2954, 1980.

/8/ J.M. McCormick, M.G. Salvadori
Numerical Methods in FØRTRAN, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1965

/9/ S. Cevolani, M. Dalle Donne, A. Martelli
unpublished, 1981.

Appendix 1: Text of the Code

```
C     S. CEVOLANI  A. MARTELLI
C     ***********  ***********
C
C
C
C     ==================================================================
C     S A G A P O - A
C
C     A COMPUTER CODE FOR THE THERMO-FLUIDDYNAMIC ANALYSIS OF GAS COOLED
C     BUNDLES OF PARTLY SMOOTH AND PARTLY ROUGHENED RODS IN STEADY STATE
C     CONDITIONS
C     ==================================================================
C
C
C     THE CODE HAS BEEN WRITTEN IN FORTRAN IV FOR THE COMPUTER IBM 3033
C     OF THE KARLSRUHE NUCLEAR CENTER.
C
C     DESCRIPTION OF THE PHYSICAL MODEL: KFK 2436-EUR 5508D
C                                        KFK 3148-EUR 7051D
C     USER'S GUIDE:                      KFK 2483-EUR 5510E
C                                        KFK 3185-EUR 7054e
C
C
C     MAIN PROGRAM
C     -----------------------------------------------------------------
C     THE MAIN PROGRAM  READS MOST OF THE INPUT DATA, PRINTS AND PUNCHS
C     MOST OF THE COMPUTED RESULTS,PERFORMS SOME SIMPLE CALCULATIONS
C     AND ORGANIZES ALL THE CALCULATIONS OF SAGAPO
C
      REAL LENGTH,LAMBDM,MFLOW,MA,MSCH ,MI,MO,MEC,LAM ,MEA1,LAM1,MM2
     1     ,MSCH1,LAMSCH,MSCHB,MSCHB1,LAMBDA(100),MAV,MAWC,MSCWC1,LAMWC,
     2     NDE1,NDE2,KAPPA
      REAL*8 COOLA(4) /'HELIUM  ','CO2     ','NYTROGEN','        '/
      REAL*8 TITLE(4,7)/'INITIAL ','UNHEATED',' SMOOTH ','PART',
     1       'FIRST HE','ATED SMO','OTH PART',' ','FIRST UN',
     2       'HEATED R','OUGH PAR','T','ROUGH PA','RT (HEAT',
     3       'ED OR UN','HEATED)','LAST  UN','HEATED R','OUGH PAR','T',
     4       'SECOND H','EATED SM','OOTH PAR','T','LAST UNH','EATED SM',
     5       'OOTH PAR','T'/
      DIMENSION PGDPT(4),EPSIT(4),CSPT(4),DPBAR(100),PBAR(100),
     1          T(100),RHOBT(100),ETABT(100),UBT(100),REBT(100),P(100),
     2          GRI( 42,3,7),GRIP( 42,3,7),XDE1(7),XDE2(7),
     3          QPIN( 19),XLAM1(7),NSPAC(7),PLEN(7),VDIAM(7),FAREL(7),
     4          CIPA(7),ZIPA(7),TWTIPA(7),TBTIPA(7),TBPIPA(7),WSP(7),
     5          PLENO(7),RHIPA(3),ACW(46),DECW(46),MEC(46),
     6          AA1(30),DEA1(30),MEA1(30),RMISTW(7),RINT(7),
     7          HPLUS1( 42,3),HPLUS2( 42,3),TWA( 42,3),QPLUSA( 42,3),
     8          PRBA ( 42,3),XSTART(7),XEND(7),AMASST( 42,3)
     9          ,AMASSB( 42,3),TEMPBA( 42,3),YDHA( 42,3),TEMPTA( 42,3)
      DIMENSION INDSP(100),NEXPR(7 ),PEX(10),XEXPR(10),NEXTW(7 ),
     1          XEXTW(3),TWTH( 42,3,3),TWP( 42,3),DELTIO( 18,2,90),
     2          GRI1( 18,2,7),GRI2( 18,2,7),YODHA( 42,3),
     3          X2DPRQ(7),NDPRQ(7),QDCOI(7),QLDCOI(7),QDCO(7,7),
     4          QLDCO(7,7),XPRQ(3),BIOT( 42,3),QSECT(3)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 28.02.1981
C    4          QLDCO(7,7),XPRQ(3),BIOT( 42,3),TWINF( 42,3),QSECT(3)
      DIMENSION TWWP( 42,3,2),TWWH( 42,3,2,10)
C
      COMMON /SC33C/  TWINF( 42,3)
      COMMON /GASD4/  IGAS
      COMMON /CEV04/  LAMOP3
```

```
      COMMON /GRIDWC/ EPSWC( 18,2,2,4),CSPWC( 18,2,2,4)
      COMMON /CORRE/  QHRDAR,QRMDAR,QLAMR
      COMMON /QPAR1/  QDEV
      COMMON /QPAR2/  QLINM,QLDEV
      COMMON /QPAR3/  PERL(3)
      COMMON /CORR/   SIGMA( 42),PHI( 42),SBMNS
      COMMON /IDISPB/ IDISP2
      COMMON /CORR1/  SIGMAI( 42,3),PHII( 42,3)
      COMMON /CORR2/  CHI( 18,2,2),PSI( 18,2,2)
      COMMON /GEN1/   LAM ( 42)
      COMMON /GEN2/   A   ( 42)
      COMMON /GEN3/   MI  ( 42)
      COMMON /GEN4/   TEMP( 42)
      COMMON /GEN5/   DE  ( 42)
      COMMON /GEN6/   MO  ( 42)
      COMMON /HEA1/   Q   ( 19)
      COMMON /HEA5/   QQ  ( 42,3)
      COMMON /GRID/   CSPC( 42,4)
      COMMON /GASD1/  NSTOT
      COMMON /GRID0/  CSPSC( 42,3,4)
      COMMON /GRID1/  EPSISC( 42,3,5),DIST(7 )
      COMMON /GRID3/  X(100)
      COMMON /GRID6/  EPSIC ( 42,4)
      COMMON /GRID7/  PGDPC ( 42,4)
      COMMON /GRID8/  PGDPSC( 42,3,4)
      COMMON /IND3/   NTYP( 42)
      COMMON /GEO2/   ATOT,DETOT,ASEC
      COMMON /GEO0/   ACH(3)
      COMMON /GAGR/   DPSI
      COMMON /GEO5/   ATC,DETC,ATW,DETW,ATA,DETA,AAC,AAW,AAA
      COMMON /HB3/    TEMP2( 42)
      COMMON /MOB1/   MM2( 42)
      COMMON /MOB4/   WCF( 42)
      COMMON /MOB2/   UAV( 42)
      COMMON /MOB5/   TAV( 42)
      COMMON /MOB6/   MAV( 42)
      COMMON /MOB8/   DPNS( 42)
      COMMON /MOB24/  WT( 42,3)
      COMMON /HEA6/   NPIN( 42),JPIN( 42,3)
      COMMON /SUB1/   ASCH( 42,3)
      COMMON /SUB2/   TSCH( 42,3),MSCH( 42,3)
      COMMON /SUB6/   TSCH1( 42,3)
      COMMON /SUB5/   LAMSCH( 42,3)
      COMMON /SUB8/   MSCH1( 42,3)
      COMMON /SUB20/  PROVI( 18,2)
      COMMON /SUB22/  TW( 42,3)
      COMMON /SUB23/  HPLUSB( 42,3),HPLUSW( 42,3),QPLUS( 42,3),
     >                PRB( 42,3),YODH( 42,3)
      COMMON /COLAM1/ COLAMB
      COMMON /IJ1/    NER( 42),NIS( 42,3)
      COMMON /MART/   ITCORR
      COMMON /DAT/    PIG
      COMMON /COLAM2/ COLAMA
      COMMON /SUB21/  TTSCHA( 18,2),TTSCHB( 18,2)
      COMMON /PRSPA/  DIST00
      COMMON /DAT1/   O0,O1,O2,O3,O4,O5,O16,O17,O18,O19
      COMMON /DAT2/   O6,O7,O8,O9,O10,O11,O12,O13,O14,O15
      COMMON /DAT4/   NDEST,NDEEND
      COMMON /DAT6/   IRHPL
      COMMON /DAT7/   CNUSS(2)
      COMMON /WACO1/  XMSCHB( 18,2),XMSCHA( 18,2)
      COMMON /WCSE2/  MSCWC1( 18,2,2)
      COMMON /WCSE3/  LAMWC( 18,2,2)
      COMMON /WCSE5/  TSCWC1( 18,2,2)
      COMMON /WCSE6/  ASCWC1( 18,2,2)
```

```
      COMMON /WCSE7/  MAWC( 18,2,2)
      COMMON /WCSE8/  ASCHWC( 18,2,2)
      COMMON /WCSE9/  TAVWC( 18,2,2)
      COMMON /WCSE1/  DEWC( 18,2,2),PPWWCC( 18,2,2)
      COMMON /WCSE12/ TWWC( 18,2,2)
      COMMON /GRAV/   IGRAV
      COMMON /PARTB/  TEMPB( 42,3),XMASSB( 42,3),YDH( 42,3)
      COMMON /INITL/  XMHE
      COMMON /WSSCH/  T1SSC1( 18,2),T2SSC1( 18,2),
     >                T1SSC2( 18,2),T2SSC2( 18,2)
      COMMON /WSSCH0/ TBSSC1( 42,3),TWSSC1( 42,3),
     >                TBSSC2( 42,3),TWSSC2( 42,3)
      COMMON /WSSCH1/ DELTIE( 18,2,90),DTIEAV( 18,2)
      COMMON /WSSCH2/ TIO( 18,2,90)
      COMMON /IROSMO/ IRH
      COMMON /SUBLA/  CLASUB
      COMMON /SHROUD/ TLINER( 18,2)
      COMMON /QSHR/   QALIN
      COMMON /LAMIN3/ F1ATIP( 42),F1DTIP( 42)
      COMMON /LAMIN4/ F2ATIP( 42,3),F2DTIP( 42,3)
      COMMON /LAMIN5/ RTIP(7)
      COMMON /LAMIN0/ I2TIP( 42,3)
      COMMON /MART5/  NSTR
      COMMON /GAAGT/  FCOPWT
      COMMON /INPAR/  IPA
      COMMON /LIRGR/  IRGRI( 42,3)
      COMMON /SC01L/  STLEN
      COMMON /SC02C/  QJ( 19, 42)
      COMMON /SC03C/  NRODS
      COMMON /SC07C/  H1
      COMMON /SC13C/  GEO1( 42,3)
      COMMON /SC15C/  ALFA( 42,3)
      COMMON /SC19C/  ICON
      COMMON /SC32C/  GHPIU( 42,3)
      COMMON /CEV02/  RTIP4
      COMMON /CEV03/  LAMOP2
      COMMON /SC99C/  TD( 42,3)
      COMMON /SC03L/  IPHUD
      COMMON /SC04L/  TAMB,HUDFAC,TMS
      COMMON /SC05L/  PERLTT
      COMMON /SC02R/  PIT,DIT,ZIT,ZWCIT,H,LENGTH
      COMMON /SC10R/  QSTOT,QRTOT
      COMMON /SC06L/  SHQ( 18,2)
      COMMON /SC07L/  WSP0,XM,NSS
      COMMON /SC08L/  AGRI( 42,3,7)
      COMMON /SC09L/  SDIS(7)
      COMMON /SC10L/  RES,PRS,EPSS,DETOLA
      COMMON /SC05R/  IRAD
      COMMON /SC01P/  S( 19)
      COMMON /SC02P/  IPSUB
C
      EXTERNAL RTRI,RTSI
C
C     ...........................................................
C
C   1-READ AND WRITE INPUT DATA
C
      WRITE(6,4721)
 4721 FORMAT( ' BEGINN OF SAGAPO, FIRST EXECUTABLE STATEMENT')
      TIME0=0.
      TIME1=ZEIT(TIME0)
      DIST00=-1.E07
      PEXOUT=0.
      COLAMB=1.
      SQ3=SQRT(3.)
      PIG=3.141593
```

```
    1 FORMAT(8I10)
    2 FORMAT(8F10.5)
  357 FORMAT(20A4)
      READ(5,357) ADUMMY
      READ(5,1)NEXCON,NRODS,NSPACT,NSPAC
      CALL HEATBA(-1,PBT,INDQ,TE,MFLOW)
      NSS=NSPACT
      READ(5,2)C,Z,ZWC,RH,PLEN,VDIAM
      READ(5,2)AREFB,RMISTW,RINT,RTIP
      READ(5,1)NDVQ,NSEL,NSC30C,NSC30W,NSC30A
      READ(5,2)PE,PE1,TE,TE1,MFLOW,XLAM1
      READ(5,2)COLAMA
      READ(5,1)IPAST,IPAEND,IREAD1
      READ(5,2)STLEN
      READ(5,1)INDPR,INDQ
      READ(5,1) NEXPRT,NEXPR
      READ(5,1) NEXTWT,NEXTW
      READ(5,1)ITCM,ITC1,ITC2,MSUBDH
      READ(5,2)XDE1,XDE2
      READ(5,2)FT,PCORR,CTU1,CTU2,CTU3
      READ(5,2)TWPRCF,TCPRCF
      READ(5,2)CINL,COUT
      READ(5,2)FAREL
      READ(5,2)TWTIPA,TBTIPA,TBPIPA
      IF(NEXPRT.GT.0)READ(5,2)(XEXPR(I),PEX(I),I=1,NEXPRT),PEXOUT
      IF(NEXTWT.GT.0)READ(5,2)(XEXTW(I),I=1,NEXTWT)
      IF(NDVQ.EQ.1)GOTO 3
      READ(5,2)(QPIN(I),I=1,NRODS)
      GOTO 5
    3 READ(5,2)Q1
      DO 4 I=1,NRODS
C
    4 QPIN(I)=Q1*S(I)
C
    5 CONTINUE
      READ(5,2)QLINMT
      IF(INDQ.EQ.1)GOTO 3800
      DO 3799 I=1,NRODS
 3799 QPIN(I)=QPIN(I)/4.186
      QLINMT=QLINMT/4.186
 3800  CONTINUE
      READ(5,1)NDPRQT,NDPRQ
      IF(NDPRQT.EQ.0)GOTO 3716
      READ(5,2)(X2DPRQ(I),I=1,NDPRQT)
      READ(5,1)NQDCO
        READ(5,4001)((QDCO(I,J),J=1,NQDCO),I=1,NDPRQT)
        READ(5,4001)((QLDCO(I,J),J=1,NQDCO),I=1,NDPRQT)
 4001 FORMAT(6E12.5)
 3716 CONTINUE
      HEALEN=PLEN(2)+PLEN(4)+PLEN(6)
      TOTLEN=PLEN(1)+PLEN(3)+PLEN(5)+PLEN(7)+HEALEN
      IF(ABS(PE/PE1-1.).GT.1.E-05)CINL=0.
      PE0=PE1
      IF(INDPR.EQ.1)GOTO 6529
      PE=PE/0.980665
      PE0=PE1/0.980665
 6529 CONTINUE
      PEBAR=PE*0.980665
      PE0BAR=PE0*0.980665
      WRITE(6,6)NRODS,PE0,PE0BAR,TE1,MFLOW,COOLA(IGAS),C,Z,ZWC,TOTLEN,
     >          HEALEN
    6 FORMAT(1H1,5X,I4,' RODS BUNDLE :'//5X,'INLET PRESSURE=',F10.7,' KG
     >/SQCM   =',F10.7,' BARS'
     1      /5X,'INLET TEMPERATURE=',F10.2,' C'/5X,'TOTAL MASS FLOW RATE
     2=',F12.6,' G/SEC'/5X,'COOLANT : ',A8,
```

```
     2                ///5X, GEOMETRY AT 20 DEGREES :  //
     3                                        5X,'ROD PITCH=',F10.6,' CM'/
     45X,'DISTANCE CENTER OF ROD - EXAGONAL WALL=',F10.6,' CM'/5X,
     5'ZWC=',F10.7,' CM'//5X,
     6  'TOTAL LENGTH=',F10.3,' CM'/5X,'HEATED LENGTH=',F10.3,' CM'/5X,
     7'LENGTH AND VOL. DIAMETERS FOR THE EXISTING PARTS :')
      DO 972 IPA=1,7
      IF(PLEN(IPA).LE.1.E-06)GOTO 972
      WRITE(6,971)IPA,PLEN(IPA),IPA,VDIAM(IPA)
  971 FORMAT(5X,'LENGTH(',I1,')=',F10.6,' CM',5X,'VOL. DIAM.(',I1,')=',
     1    F10.6,' CM')
  972 CONTINUE
      IF(PLEN(4).GT.1.E-06)
     *WRITE(6,980)RH,
     1            016,017,04,05,00,01,02,03,                 06,07,08,09
     2,010,011,013,014,015
  980 FORMAT(//////5X,'HEIGHT OF ROUGHNESS (RH) =',F8.5,' CM'/5X,
     1'G(H+) * ((R2-R1)/RH*',F6.3,')**',F6.3,' / (PR**',F6.3,'*((TW+273.
     215)/(TB+273.15))**',F6.3,')='/5X,'=',F6.3,'*(HW+)**',F6.3,'+',F9.3
     3,'/(HW+)**' ,F6.3///5X,
     5 'R(H+)=(', F6.3,'+', F7.1,'/(HW+)**', F6.3,')**', F6.3,'+',F6.3,
     6'*LN(RH/(',F6.3,'*(R0-R1))'/12X,'+',F6.3,'/(HW+)**',F6.3,'*((TW+27
     73.16) /(TB1+273.16)-1)**',F6.3//)
      WRITE(6,3727)QLINMT,(I,QPIN(I),I=1,NRODS)
 3727 FORMAT(//5X,'MAXIMUM POWER FROM THE LINER:'//5X,'Q MAX=',E15.5,
     1   ' CAL/SEC*CM'
     2        ///5X,'MAXIMUM POWER OF RODS:'//(5X,'Q MAX(',I4,')=',E15.5,
     3   ' CAL/SEC*CM'))
      IF(NDPRQT.EQ.0)GOTO 3730
      WRITE(6,3731)
 3731 FORMAT(//5X,'COEFFICIENTS FOR THE POLYNOMIAL PROFILES OF THE ROD P
     1OWER ( 0 TAKEN AT THE BEGINNING OF THE ACTUAL PART ):'/)
      DO 3729 I=1,NDPRQT
 3729 WRITE(6,3728)X2DPRQ(I),(QDCO(I,J),J=1,NQDCO)
 3728 FORMAT(/5X,'AS FAR AS X =',F10.6,' CM :'/(5X,8E15.5))
      WRITE(6,3733)
 3733 FORMAT(//5X,'COEFFICIENTS FOR THE POLYNOMIAL PROFILE OF THE LINER
     1POWER ( 0 TAKEN AT THE BEGINNING OF THE ACTUAL PART ):'/)
      DO 3732 I=1,NDPRQT
 3732 WRITE(6,3728)X2DPRQ(I),(QLDCO(I,J),J=1,NQDCO)
 3730 CONTINUE
C     ..............................................................
C
C   2-INDEXING AND CONNECTIONS FOR THE CHANNELS
C
      CALL INDEX(NSEL,NEXCON,NSTR,NSTOT,NROM1)
      NSPER=NSTOT-NSTR
C     ..............................................................
C
C   3-READ AND WRITE INPUT DATA
C
      IF(NSPACT.EQ.0)GOTO 7
      READ(5,2)WSP0,(DIST(I),I=1,NSPACT)
      WRITE(6,970)WSP0,(I,DIST(I),I=1,NSPACT)
  970 FORMAT(//5X,'SPACERS (AT 20 DEGREES):'//5X,'WIDTH=',F10.6,' CM'//
     1(5X,'DIST(',I2,')=',F10.3,' CM'))
      WRITE(6,83)
      DO 11 I=1,NSPACT
       READ(5,981)((GRI(NS,J,I  ),J=1,3),NS=1,NSTOT)
C     READ(5,981)((GRIP(NS,J,I  ),J=1,3),NS=1,NSTOT)
C     READ(5,981)((GRI1(K,J,I  ),J=1,2),K=1,NSPER)
C     READ(5,981)((GRI2(K,J,I  ),J=1,2),K=1,NSPER)
   11 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 1119 I=1,NSPACT
              SDIS(I)=DIST(I)
```

```
               DO 1115 NS=1,NSTOT
                    DO 1113 J=1,3
                    GRIP(NS,J,I)=1.0
                    AGRI(NS,J,I)=GRI(NS,J,I)
 1113               CONTINUE
 1115          CONTINUE
           DO 1118 NS=1,NSPER
             DO 1116 J=1,2
               GRI1(NS,J,I)=1.0
               GRI2(NS,J,I)=1.0
 1116          CONTINUE
 1118        CONTINUE
 1119      CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      READ(5,982)((IRGRI(NS,J),J=1,3),NS=1,NSTOT)
  981 FORMAT(6F10.5)
  982 FORMAT(6I10)
    7 CONTINUE
      READ(5,2)TIMEPU
      READ(5,1)IPUNCH
C     ..............................................................
C
C   4-CORRECTION OF THE INPUT DIMENSIONS TO TAKE INTO ACCOUNT THE
C     THERMAL EXPANSION OF THE BUNDLE STRUCTURE
C
      SPLENO=0.
      NEXPRP=0
      NEXTWP=0
      NSPACP=0
      NDPRQP=0
      DO 882 IPA=1,7
      EXFTBP=1.+EXPCL(TBPIPA(IPA))*(TBPIPA(IPA)-20.)
      EXFTWT =1.+EXPCO(TWTIPA(IPA))*(TWTIPA(IPA)-20.)
      IF(NEXPR(IPA).EQ.0)GOTO 1010
      IEXPR1=NEXPRP+1
      IEXPR2=NEXPRP+NEXPR(IPA)
      NEXPRP=IEXPR2
      DO 1009 IEXPR=IEXPR1,IEXPR2
 1009 XEXPR(IEXPR)=(XEXPR(IEXPR)-SPLENO)*EXFTBP
 1010 CONTINUE
      IF(NEXTW(IPA).EQ.0)GOTO 1012
      IEXTW1=NEXTWP+1
       IEXTW2=NEXTWP+NEXTW(IPA)
      NEXTWP=IEXTW2
      DO 1011 IEXTW=IEXTW1,IEXTW2
 1011 XEXTW(IEXTW)=(XEXTW(IEXTW)-SPLENO)*EXFTWT
 1012 CONTINUE
      IF(NDPRQ(IPA).EQ.0)GOTO 1015
      IDPRQ1=NDPRQP+1
      IDPRQ2=NDPRQP+NDPRQ(IPA)
      NDPRQP=IDPRQ2
      DO 1014 IDPRQ=IDPRQ1,IDPRQ2
      CALL MODFQD(IDPRQ,NDPRQT,NQDCO,QDCO,EXFTWT)
      CALL MODFQD(IDPRQ,NDPRQT,NQDCO,QLDCO,EXFTWT)
 1014 X2DPRQ(IDPRQ)=(X2DPRQ(IDPRQ)-SPLENO)*EXFTWT
 1015 CONTINUE
      IF(NSPAC(IPA).EQ.0)GOTO 882
      ISPAC1=1+NSPACP
      ISPAC2=NSPACP+NSPAC(IPA)
      NSPACP=ISPAC2
      WSP(IPA)=WSP0*(1.+EXPCO(TBTIPA(IPA))*(TBTIPA(IPA)-20.))
      DO 881 ISPAC=ISPAC1,ISPAC2
  881 DIST(ISPAC)=(DIST(ISPAC)-SPLENO)*EXFTBP
  882 SPLENO=SPLENO+PLEN(IPA)
      EXCON=NEXCON
```

```
      DO 983 IPA=1,7
      EXFTBP=1.+EXPCL(TBPIPA(IPA))*(TBPIPA(IPA)-20.)
      EXFTWT =1.+EXPCO(TWTIPA(IPA))*(TWTIPA(IPA)-20.)
      PLEN0(IPA)=PLEN(IPA)
      PLEN(IPA)=PLEN(IPA)*EXFTBP
      RMISTW(IPA)=RMISTW(IPA)*EXFTWT
      RINT  (IPA)=RINT  (IPA)*EXFTWT
      RTIP  (IPA)=RTIP  (IPA)*EXFTWT
      VDIAM(IPA)=VDIAM(IPA)*EXFTWT
      CIPA(IPA)=C*(1.+EXPCO(TBTIPA(IPA))*(TBTIPA(IPA)-20.))
      ZIPA(IPA)=(Z+EXCON*C*SQ3*0.5)*EXFTBP-EXCON*CIPA(IPA)*SQ3*0.5
  983 CONTINUE
      SPLEN=0.
      NEXPRP=0
      NSPACP=0
      NDPRQP=0
      DO 885 IPA=1,7
      IF(NEXPR(IPA).EQ.0)GOTO 1020
      IEXPR1=NEXPRP+1
      IEXPR2=NEXPRP+NEXPR(IPA)
      NEXPRP=IEXPR2
      DO 1019 IEXPR=IEXPR1,IEXPR2
 1019 XEXPR(IEXPR)=XEXPR(IEXPR)+SPLEN
 1020 CONTINUE
      IF(NSPAC(IPA).EQ.0)GOTO 885
      ISPAC1=1+NSPACP
      ISPAC2=NSPACP+NSPAC(IPA)
      NSPACP=ISPAC2
      DO 884 ISPAC=ISPAC1,ISPAC2
  884 DIST(ISPAC)=DIST(ISPAC)+SPLEN
  885 SPLEN=SPLEN+PLEN(IPA)
      DO 8 IPA=1,7
      IF(PLEN(8-IPA).LE.1.E-06)GOTO 8
      IPAM=8-IPA
      GOTO 9
    8 CONTINUE
    9 CONTINUE
      PLEN(IPAM)=SPLEN
      IPA1=IPAM-1
      DO 10 IPA=1,IPA1
      PLEN(IPA)=PLEN0(IPA)*(1.+EXPCO(TWTIPA(IPA))*(TWTIPA(IPA)-20.))
      PLEN(IPAM)=PLEN(IPAM)-PLEN(IPA)
   10 CONTINUE
      DO 886 IPA=3,5
      IF(PLEN(IPA).LE.1.E-06)GOTO 886
      RHIPA(IPA-2)=RH*(1.+EXPCO(TWTIPA(IPA))*(TWTIPA(IPA)-20.))
  886 CONTINUE
      SPLEN=0.
      NEXTWP=0
      DO 1013 IPA=1,7
      IF(NDPRQ(IPA).EQ.0)GOTO 1017
      IDPRQ1=NDPRQP+1
      IDPRQ2=NDPRQP+NDPRQ(IPA)
      NDPRQP=IDPRQ2
      DO 1016 IDPRQ=IDPRQ1,IDPRQ2
 1016 X2DPRQ(IDPRQ)=X2DPRQ(IDPRQ)+SPLEN
      X2DPRQ(IDPRQ2)=X2DPRQ(IDPRQ2)*1.1
 1017 CONTINUE
      IF(NEXTW(IPA).EQ.0)GOTO 1022
      IEXTW1=NEXTWP+1
       IEXTW2=NEXTWP+NEXTW(IPA)
      NEXTWP=IEXTW2
      DO 1021 IEXTW=IEXTW1,IEXTW2
 1021 XEXTW(IEXTW)=XEXTW(IEXTW)+SPLEN
 1022 CONTINUE
```

```
 1013 SPLEN=SPLEN+PLEN(IPA)
      UNHLE=PLEN(1)+PLEN(3)
C     .....................................................................
C
C   5-REARRANGEMENT OF THE GEOMETRIC AXIAL DATA IF THE CALCULATION DOES
C     NOT START AT THE BUNDLE INLET
C
      ISTAIN=1
      SPLEN=0.
      IPAST1=IPAST-1
      NEXPRS=0
      NEXTWS=0
      NSPACS=0
      NDPRQS=0
      IF(IPAST1.EQ.0)GOTO 2222
      DO 6532 IPA=1,IPAST1
      SPLEN=SPLEN+PLEN(IPA)
      PLEN(IPA)=0.
      NEXPRT=NEXPRT-NEXPR(IPA)
      NEXPRS=NEXPRS+NEXPR(IPA)
      NEXPR(IPA)=0
       NEXTWT=NEXTWT-NEXTW(IPA)
      NEXTWS=NEXTWS+NEXTW(IPA)
      NEXTW(IPA)=0
      NDPRQT=NDPRQT-NDPRQ(IPA)
      NDPRQS=NDPRQS+NDPRQ(IPA)
      NDPRQ(IPA)=0
      NSPACT=NSPACT-NSPAC(IPA)
      NSPACS=NSPACS+NSPAC(IPA)
       NSPAC(IPA)=0
 6532 CONTINUE
      IF(IPAST.EQ.4)AREFB=AREFB*(PLEN(4)+SPLEN-STLEN)/PLEN(4)
 2222 CONTINUE
      PLEN(IPAST)=PLEN(IPAST)+SPLEN-STLEN
      IF(ABS(STLEN-SPLEN).GT.1.E-04)ISTAIN=2
      IF(NEXPRT.EQ.0)GOTO 6534
      IEXPR1=0
      DO 6533 I=1,NEXPRT
      XEXPR(I)=XEXPR(I+NEXPRS)-STLEN
      IF(XEXPR(I).LE.0.)IEXPR1=IEXPR1+1
 6533 PEX(I)=PEX(I+NEXPRS)
      NEXPR(IPAST)=NEXPR(IPAST)-IEXPR1
       NEXPRT=NEXPRT-IEXPR1
      IF(NEXPRT.EQ.0)GOTO 6534
      DO 973 I=1,NEXPRT
      XEXPR(I)=XEXPR(I+IEXPR1)
  973 PEX(I)=PEX(I+IEXPR1)
 6534 CONTINUE
      IF(NEXTWT.EQ.0)GOTO  6536
      IEXTW1=0
      DO 6535 I=1,NEXTWT
      XEXTW(I)=XEXTW(I+NEXTWS)-STLEN
      IF(XEXTW(I).LE.0.)IEXTW1=IEXTW1+1
 6535 CONTINUE
      NEXTW(IPAST)=NEXTW(IPAST)-IEXTW1
      NEXTWT=NEXTWT-IEXTW1
      IF(NEXTWT.EQ.0)GOTO 6536
      DO 974 I=1,NEXTWT
  974 XEXTW(I)=XEXTW(I+IEXTW1)
 6536 CONTINUE
      IF(NDPRQT.EQ.0)GOTO 6539
      IDPRQ1=0
      DO 6540 I=1,NDPRQT
      X2DPRQ(I)=X2DPRQ(I+NDPRQS)-STLEN
      IF(X2DPRQ(I).LE.0.)IDPRQ1=IDPRQ1+1
      DO 6540 IQDCO=1,NQDCO
```

```
      QDCO(I,IQDCO)=QDCO(I+NDPRQS,IQDCO)
      QLDCO(I,IQDCO)=QLDCO(I+ NDPRQS,IQDCO)
 6540 CONTINUE
      NDPRQ(IPAST)=NDPRQ(IPAST)-IDPRQ1
      NDPRQT=NDPRQT-IDPRQ1
      IF( NDPRQT.EQ.0)GOTO  6539
      DO 976 I=1,NDPRQT
      X2DPRQ(I)=X2DPRQ(I+IDPRQ1)
      DO 976 IQDCO=1,NQDCO
      QDCO(I,IQDCO)=QDCO(I+IDPRQ1,IQDCO)
      QLDCO(I,IQDCO)=QLDCO(I+IDPRQ1,IQDCO)
  976 CONTINUE
 6539 CONTINUE
      IF(NSPACS.NE.0)DIST00=DIST(NSPACS)-STLEN
      NSPA00=NSPACT+1
      IF(NSPACS.GT.NSPA00)NSPA00=NSPACS
      IF(NSPACT.EQ.0 .AND. NSPACS.EQ.0)GOTO 6538
      ISPAC1=0
      DO 6537 I=1,NSPA00
      IF(I.GT.NSPACT)GOTO 2006
      DIST(I)=DIST(I+NSPACS)-STLEN
      IF(DIST(I).LE.0.)ISPAC1=ISPAC1+1
      IF(DIST(I).LE.0.)DIST00=DIST(I)
 2006 CONTINUE
      DO 2000 NS=1,NSTOT
      DO 2001 J=1,3
      IF(NSPACS.GT.0)GRI00=GRI(NS,J,I)
      IF(I.LE.NSPACT)GRI(NS,J,I)=GRI(NS,J ,I+NSPACS)
      IF(I.LE.NSPACT .AND. I.GE.NSPACS .AND. NSPACS.GT.0)GRI(NS,J,I+1)=
     =GRI00
      IF(I.GT.NSPACT .AND. I.GE.NSPACS .AND. NSPACS.GT.0)GRI(NS,J,NSPACT
     ++1)=GRI00
      IF(I.LE.NSPACT)GRIP(NS,J,I)=GRIP(NS,J,I+NSPACS)
 2001 CONTINUE
      IF(NS.LE.NSTR .OR. I.GT.NSPACT)GOTO 2000
      NSW=NS-NSTR
      DO 2002 J=1,2
      GRI1(NSW,J,I)=GRI1(NSW,J,I+NSPACS)
 2002 GRI2(NSW,J,I)=GRI2(NSW,J,I+NSPACS)
 2000 CONTINUE
 6537 CONTINUE
      NSPAC(IPAST)=NSPAC(IPAST)-ISPAC1
      NSPACT=NSPACT-ISPAC1
      NSPA00=NSPACT+1
      IF(ISPAC1.GT.NSPA00)NSPA00=ISPAC1
      IF(NSPACT.EQ.0 .AND. ISPAC1.EQ.0)GOTO 6538
       DO 977 I=1,NSPA00
      DO 2003 NS=1,NSTOT
      DO 2004 J=1,3
      IF(ISPAC1.GT.0)GRI00=GRI(NS,J,I)
      IF(I.LE.NSPACT)GRI(NS,J,I)=GRI(NS,J,I+ISPAC1)
      IF(I.GE.ISPAC1 .AND. ISPAC1.GT.0 .AND. I.LE.NSPACT)GRI(NS,J,I+1)=
     =GRI00
      IF(I.GE.ISPAC1 .AND. ISPAC1.GT.0 .AND. I.GT.NSPACT)GRI(NS,J,
     *NSPACT+1)=GRI00
      IF(I.GT.NSPACT)GOTO 2004
      GRIP(NS,J,I)=GRIP(NS,J,I+ISPAC1)
 2004 CONTINUE
      IF(NS.LE.NSTR .OR. I.GT.NSPACT)GOTO 2003
      NSW=NS-NSTR
      DO 2005 J=1,2
      GRI1(NSW,J,I)=GRI1(NSW,J,I+ISPAC1)
 2005 GRI2(NSW,J,I)=GRI2(NSW,J,I+ISPAC1)
 2003 CONTINUE
      IF(I.LE.NSPACT)DIST(I)=DIST(I+ISPAC1)
  977 CONTINUE
```

```
 6538 CONTINUE
      HEALEN=PLEN(2)+PLEN(4)+PLEN(6)
      UNHLE1=PLEN(1)+PLEN(3)
      TOTLEN=UNHLE1+HEALEN+ PLEN(5)+PLEN(7)
      HRDAR=RH/AREFB
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(PLEN(4).GT.1.E-06)FCORLA=ALOG(RTIP(4)/(RTIP4-RHIPA(2)))/
     /                        (2.*PIG*PLEN(4))
C     IF(PLEN(4).GT.1.E-06)FCORLA=ALOG(RTIP(4)/(RTIP(4)-RHIPA(2)))/
C    /                        (2.*PIG*PLEN(4))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     .............................................................
C
C   6-INITIALIZATION OF VARIABLES
C
      QLINMT=QLINMT*HEALEN
      DO 3734 I=1,NRODS
 3734 QPIN(I)=QPIN(I)*HEALEN
      ANCE=NSC30C
      ANWA=NSC30W
      ANCO=NSC30A
      ALFACE=PIG/(6.*ANCE)
      ALFAWA=PIG/(6.*ANWA)
      ALFACO=PIG/(ANCO*6.)
      NSC90=3*NSC30W
      NSC45=NSC30C/2+1+NSC30C
      L=1
      T(1)=TE
      P(1)=PE
      PBAR(1)=PEBAR
      X(1)=0.
      XDEST=NDEST
      XDEEND=NDEEND
      TO=TE
      PO=PE
      INLET=1
      ISPAC=1
      II=1
      HH=0.
      IEXPR1=1
      IEXTW1=1
      IEXTWC=1
      IDPRQ=1
      SPRLEN=0.
      IRHPL=1
C     .............................................................
C
C     7- LOOP IPA :  A SUBDIVISION OF RODS INTO SEVEN POSSIBLE DIFFERENT
C        PARTS IS MADE ( BUT ONLY FIVE TOGETHER ARE SUPPOSED TO EXIST :
C        1) SMOOTH UNHEATED+SMOOTH HEATED+ROUGH+SMOOTH HEATED+SMOOTH
C           UNHEATED
C        2) SMOOTH UNHEATED+ROUGH UNHEATED+ROUGH HEATED+ROUGH UNHEATED+
C           SMOOTH UNHEATED)
C
C     IPA=1 : INITIAL UNHEATED SMOOTH PART
C     IPA=2 : FIRST HEATED SMOOTH PART
C     IPA=3 : FIRST UNHEATED ROUGH PART
C     IPA=4 : ROUGH PART (HEATED OR UNHEATED)
C     IPA=5 : LAST UNHEATED ROUGH PART
C     IPA=6 : SECOND HEATED SMOOTH PART
C     IPA=7 : LAST UNHEATED SMOOTH PART
C
      DO 8888 IPA=1,7
      IF(IPA.EQ.IPAEND+1)CALL TMPUN(NSTOT,NSTR,T(L),P(L),PBAR(L),
     *TE1,PEO,PEOBAR,INDPR,MFLOW,IPA,IPAEND,2,XLAM1,X(L)+STLEN,&742)
      IF(PLEN(IPA)  .LE.1.E-06)GOTO 8888
```

```
        SPLENG=SPRLEN
      IF(IPA.EQ.IPAST)SPLENG=SPLEN-STLEN
      C=CIPA(IPA)
      Z=ZIPA(IPA)
      D=VDIAM(IPA)
      LENGTH=PLEN(IPA)
      LAM1=XLAM1(IPA)
      MSPAC=NSPAC(IPA)
      NDE1=XDE1(IPA)
      NDE2=XDE2(IPA)
      FREL=FAREL(IPA)
      ZWCIPA=ZWC     *(1.+EXPCO(TBPIPA(IPA))*(TBPIPA(IPA)-20.))
CCCCCCCCCCCCCCCCCCC
      DIT=D
      PIT=C
      ZIT=Z
      ZWCIT=ZWCIPA
CCCCCCCCCCCCCCCCCCCCCC
      P0BAR=P0*0.980665
      WRITE(6,991)(TITLE(I,IPA),I=1,4),C,Z,ZWCIPA,D,LENGTH,MSPAC,T0,P0
     1,P0BAR
  991 FORMAT(1H1,5X,4A8///5X,'C=',F10.6,' CM'/5X,'Z=',F10.6,' CM'/5X,
     1'ZWC=',F10.6,' CM'/ 5X,'VOL. DIAMETER=',F10.6,' CM'/5X,'PART LENGT
     2H='  ,F10.5,' CM'/5X,'NUMBER OF SPACERS=',I3//5X,'INLET CONDITIONS
     3 :'/5X,'INLET AVERAGE TEMPERATURE=',F7.2,' C'/5X,'INLET PRESSURE='
     4,F10.7,' KG/SQCM    =',F10.7,' BARS'/////)
      IF(MSPAC.EQ.0)GOTO 968
      ISPAC2=ISPAC+MSPAC-1
      WRITE(6,967)WSP(IPA),(I,DIST(I),I=ISPAC,ISPAC2)
  967 FORMAT(5X,'SPACERS (DISTANCES ARE EVALUATED FROM THE BUNDLE ENTRAN
     1CE) :'/5X,'WIDTH=',F10.6,' CM'/(5X,'DIST(',I2,')=',F10.3,' CM'))
      WRITE(6,83)
  968 CONTINUE
      C2=C*0.5
      EM1=C2-ZWCIPA*SQ3
      X(1)=HH
      DDD=HH+LENGTH
      SUR=PIG*D*LENGTH
      IF(IPA.EQ.5)IRHPL=2
      GOTO(993,994,993,994,993,994,993),IPA
  993 PLDHL=0.
      INDICE=0
      GOTO 995
  994 PLDHL= PLEN(IPA)/HEALEN
      INDICE=0
  995 CONTINUE
      GOTO(996,996,997,997,997,996,996),IPA
  996 IRH=1
      CLASUB=1.0576
      XMAXNU=1.6
      CHSLNU=2./3.
      GOTO 998
  997 IRH=2
      CLASUB=1.
      XMAXNU=1.
      CHSLNU=.5
      RH=RHIPA(IPA-2)
      WRITE(6,990)RH
  990 FORMAT(///5X,'HEIGHT OF ROUGHNESS=',F10.7,' CM'////)
  998 CONTINUE
      CONST=CNUSS(IRH)
      QTOT=0.
      DO 992 I=1,NRODS
      Q(I)=QPIN(I)*PLDHL
  992 QTOT=QTOT+Q(I)
C
```

```
C     .............................................................
C
C   8-SUBROUTINES HEATI,TOTGEO,INQUA, KAPCOR
C
      CALL HEATI(NSTOT,NSTR,NSEL,NEXCON,IPA)
CCCCCCCCCCCCCCCCCCCCCC
      CALL JZURU
      CALL MEZZI(NRODS,NSEL,NSTOT)
      CALL DELIP
      IF(IRAD.GT.0) CALL TOTSEC(NSEL)
      IF(IRAD.GT.0) CALL TARRAY
      IF(IRAD.GT.0) CALL VFCAL
      IF(IRAD.GT.0) CALL VFDET(NSEL,NRODS)
      IF(IRAD.GT.0) CALL VFCTR
CCCCCCCCCCCCCCCCCCCCCCC
      CALL TOTGEO(NSEL,D,C,Z,PIG,NEXCON,NRODS,WW,WA,ZA,EM1,PERLT,
     *RTIP(IPA))
CCCCCCCCCCCCCCCCCCCCCC 02.11.1979
      PERLTT=PERLT
CCCCCCCCCCCCCCCCCCCCCC 02.11.1979
      QLINM=QLINMT*PLDHL/PERLT
      CALL INQUA(NSEL,NSTOT,NEXCON,ATC,ATW,ATA,DETC,DETW,DETA)
      CALL KAPCOR(NSTOT,NSTR)
C     .............................................................
C
C   9-DEFINITION OF THE REGIONS WHERE INDISTURBED FLOW IS ASSUMED AND
C     EVALUATION OF THE SPACER PARAMETERS
C
      DXST =XDEST*DETC
      DXEND=XDEEND*DETC
      XSTART(1)=X(1)+DXST
      XEND(MSPAC+1)=X(1)+LENGTH-DXEND
      IMSPAC=ISPAC1+NSPACS
      IF(IMSPAC.EQ.0  .OR. IPA.NE.IPAST)GOTO 7007
      DO 7002 NS=1,NSTOT
      NP=NPIN(NS)
      DO 7002 J=1,NP
      DO 7002 M=1,NP
      IF(IRGRI(NS,J).EQ.JPIN(NS,M))EPSISC(NS,M,MSPAC+1)=GRI(NS,J,NSPACT+
     +1)
 7002 CONTINUE
 7007 CONTINUE
      IF(IPA.EQ.IPAST .OR. II.EQ.1)GOTO 7009
      DO 7008 NS=1,NSTOT
      NP=NPIN(NS)
      DO 7008 J=1,NP
      DO 7008 M=1,NP
      IF(IRGRI(NS,J).EQ.JPIN(NS,M))EPSISC(NS,M,MSPAC+1)=GRI(NS,J,II-1)
 7008 CONTINUE
 7009 CONTINUE
      IF(MSPAC.EQ.0)GOTO 12
      JSP=MSPAC+ISPAC-1
      IPAFD=1
      DO 4430 I=ISPAC,JSP
      I1SPAC=I-ISPAC+1
      IPAFD=IPAFD+1
      XSTART(IPAFD)=DIST(I)+DXST+WSP(IPA)*0.5
      XEND(IPAFD-1)=DIST(I)-WSP(IPA)*0.5-DXEND
      PGDPT(I1SPAC)=0.
      EPSIT(I1SPAC)=0.
      DO 5601 NS=1,NSTOT
      PGDPC(NS,I1SPAC)=0.
      EPSIC(NS,I1SPAC)=0.
      NP=NPIN(NS)
      DO 5600 J=1,NP
      DO 5599 M=1,NP
```

```
      IF(IRGRI(NS,J).NE.JPIN(NS,M))GOTO 5599
      EPSISC(NS,M,I1SPAC)=GRI(NS,J,I)
      PGDPSC(NS,M,I1SPAC)=GRIP(NS,J,I)
      EPSIC(NS,I1SPAC)=EPSIC(NS,I1SPAC)+EPSISC(NS,M,I1SPAC)*ASCH(NS,M)
      PGDPC(NS,I1SPAC)=PGDPC(NS,I1SPAC)+PGDPSC(NS,M,I1SPAC)*4.*ASCH(NS,
     *M)/DE(NS)
      IF(NTYP(NS).NE.2)GOTO 5600
      NSW=NS-NSTR
      EPSWC(NSW,M,1,I1SPAC)=GRI1(NSW,M,I)*EPSISC(NS,M,I1SPAC)
      EPSWC(NSW,M,2,I1SPAC)=GRI2(NSW,M,I)*EPSISC(NS,M,I1SPAC)
      GOTO 5600
 5599 CONTINUE
 5600 CONTINUE
      EPSIT(I1SPAC)=EPSIT(I1SPAC)+EPSIC(NS,I1SPAC)
      PGDPT(I1SPAC)=PGDPT(I1SPAC)+PGDPC(NS,I1SPAC)
      EPSIC(NS,I1SPAC)=EPSIC(NS,I1SPAC)/A(NS)
      PGDPC(NS,I1SPAC)=PGDPC(NS,I1SPAC)*DE(NS)*0.25/A(NS)
 5601 CONTINUE
      EPSIT(I1SPAC)=EPSIT(I1SPAC)/ASEC
      PGDPT(I1SPAC)=PGDPT(I1SPAC)*DETOT*0.25/ASEC
      CSPT(I1SPAC)=GRIFUN(EPSIT(I1SPAC))
      DO 5602 NS=1,NSTOT
      CSPC(NS,I1SPAC)=GRIFUN(EPSIC(NS,I1SPAC))
      NP=NPIN(NS)
      DO 5602 M=1,NP
      CSPSC(NS,M,I1SPAC)=GRIFUN(EPSISC(NS,M,I1SPAC))
      IF(NTYP(NS).NE.2)GOTO 975
      NSW=NS-NSTR
      CSPWC(NSW,M,1,I1SPAC)=GRIFUN(EPSWC(NSW,M,1,I1SPAC))
      CSPWC(NSW,M,2,I1SPAC)=GRIFUN(EPSWC(NSW,M,2,I1SPAC))
  975 CONTINUE
 5602 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  08.02.1980
      EPSS=EPSIT(I1SPAC)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WRITE(6,960)I,EPSIT(I1SPAC)
  960 FORMAT(//  5X,'SPACER NR.',I5,5X,'EPSILON TOT.=',F10.7)
      DO 964 NS=1,NSTOT
      NP=NPIN(NS)
      WRITE(6,961)NS,EPSIC(NS,I1SPAC),(JPIN(NS,M),NS,M,
     *EPSISC(NS,M,I1SPAC),M=1,NP)
  961 FORMAT(/5X,'CHANNEL NR.',I5,5X,'EPSILON=',F10.7/
     1  5X,'SUBCHANNELS:'/(5X,'ROD NR.',I5,') EPSILON(',I5,',',I2,')=',
     2  F10.7))
  964 CONTINUE
      WRITE(6,83)
 4430 CONTINUE
   12 CONTINUE
C
C     .................................................................
C  10-SUBROUTINES INGE AND CEWACO
C
      CALL INGE(NEXCON,NSEL,NSTR,NSTOT,C,Z,D,ATC,ATW,ATA,PIG,PCORR,CTU1,
     *CTU2,DETC,DETW,EM1)
      CALL CEWACO(NSC30C,NSC45,12,ALFACE,D,C2,AAC,DETC,MFLOW,ATOT,ACW,
     *           DECW,MEC)
      CALL CEWACO(NSC30A,NSC30A,3,ALFACO,D,ZA,AAA,DETA,MFLOW,ATOT,AA1,
     *           DEA1,MEA1)
      MA=MFLOW/ATOT
      PROV1=MA*DETOT
      PROV2=-1.E-03*MA**2/980.665
C     .................................................................
C
C  11-INLET MASS FLOW RATES AND TEMPERATURES ;EVALUATION OF PRESSURE
C     LOSS AT THE BUNDLE INLET
C
```

```
      IF(INLET.NE.1)GOTO 4435
      CALL INLCON(NSTOT,MFLOW,ATOT,TE,IREAD1,NSTR)
      PI=PE
      DO 4432 I=1,10
CCCCCCCCCCCCCCCCCCCCCCCC
      CIN1= DPIN(MFLOW,PI,TE,CINL)
      P0=PE+CIN1*PROV2/RHO(PI,TE)*0.5
CCCCCCCCCCCCCCCCCCCCCCCC
      IF(ABS(P0/PI-1.).LE.1.E-06)GOTO 4434
 4432 PI=P0
      WRITE(6,4433)P0,PI
 4433 FORMAT(1H1,5X,'MAIN',/5X,
     >               'CALCULATION STOPS : P0=',F10.7,'  ; PI=',F10.7)
      STOP
 4434 CONTINUE
      DPE=P0-PE
      DPEBAR=DPE*0.980665
      WRITE(6,1333)DPE,DPEBAR,CIN1
 1333 FORMAT(///130('*')//5X,
     *               'PRESSURE LOSS DUE TO ENTRANCE=',F10.7,' KG/SQCM  =',
     *F10.7,' BARS ( CINL=',F4.2,')'//)
      INLET=2
      DPBAR(1)=PE0BAR-PEBAR-DPEBAR
      IF(STLEN.GT.1.E-06 .OR. IPUNCH.EQ.2)GOTO 4435
      WRITE(1,1)NSPACT
      IF(NSPACT.GT.0)WRITE(1,6069)(DIST(I),I=1,NSPACT)
      XLTOT=0.
      WRITE(1,1)IPA
      WRITE(1,6069)XLTOT,DPBAR(1)
 4435 CONTINUE
C     ...........................................................
C
C  12-EVALUATION OF SECTION LENGTHS AND CORRECTION FACTORS FOR NUSSELT
C
      CALL AXSEC(NDE1,NDE2,DETC,WSP(IPA),CONST,DDD,II,HH,MSPAC,LENGTH,N,
     *IPA,QTOT,NSTOT,XMAXNU,CHSLNU)
      WRITE(6,14)LENGTH,N,FREL
   14 FORMAT(///130('*')///
     *       5X,'TOTAL LENGTH=',F7.2,1X,'CM',5X,'NUMBER OF SECTIONS=',I3
     *,5X,'FREL=' ,F10.4///)
C     ...........................................................
C
C  13-INITIALIZATION OF VARIABLES
C
      T(1)=T0
      P(1)=P0
      PBAR(1)=P0*0.980665
      NSEFD=0
      IPAFD=1
      TM=0.
      PM=0.
      LAMBDM=0.
      REM=0.
      UM=0.
      DELTAX=0.
      TWTC=0.
      TBTC=0.
      TBPC=0.
       NSTR1=NSTR+1
      DO 5636 NS=NSTR1,NSTOT
      NP=NPIN(NS)
      DO 5636 M=1,NP
      TLINER(NS-NSTR,M)=TSCH1(NS,M)
      IF(ISTAIN.EQ.2)GOTO 978
      DO 5634 JWC=1,2
```

```
 5634 TSCWC1(NS-NSTR,M,JWC)=TSCH1(NS,M)
  978 CONTINUE
      DO 5635 I=1,NSC90
      DELTIO(NS-NSTR,M,I)=0.
 5635 TIO(NS-NSTR,M,I)=0.
 5636 CONTINUE
      IF(IRH.NE.2)GOTO 4439
      DO 4438 NS=1,NSTOT
      NP=NPIN(NS)
      DO 4438 M=1,NP
      HPLUS1(NS,M)=0.
      HPLUS2(NS,M)=0.
      TWA(NS,M)=0.
      QPLUSA(NS,M)=0.
      PRBA (NS,M)=0.
      YDH(NS,M)=0.
      YODH(NS,M)=0.
      TEMPB(NS,M)=TE
      XMASSB(NS,M)=1.
      YODHA(NS,M)=0.
      YDHA(NS,M)=0.
      TEMPBA(NS,M)=0.
      AMASSB(NS,M)=0.
      TEMPTA(NS,M)=0.
      AMASST(NS,M)=0.
 4438 CONTINUE
 4439 CONTINUE
C
C     ..................................................................
C  14-THE AXIAL LOOP STARTS (K=INDEX OF THE AXIAL SECTION )
C
      K1=1
      NSUBDH=0
 8503 CONTINUE
      DO 9999 K=K1,N
C
      TIME2=ZEIT(TIME1)
      IF(TIME2.GT.TIMEPU)CALL TMPUN(NSTOT,NSTR,T(K),P(K),PBAR(K),
     *TE1,PEO,PEOBAR,INDPR,MFLOW,IPA,IPAEND,2,XLAM1,X(K)+STLEN,&742)
      ASECLA=ASEC
      DETOLA=DETOT
      L=K+1
      H=X(L)-X(K)
CCCCCCCCCCCCCCCCCCCCCCC
      CALL DECP(IPRINT,X(K),X(L),STLEN)
CCCCCCCCCCCCCCCCCCCCCCC
      QDEV=0.
      QLDEV=0.
      INDPRQ=1
      IF(NDPRQ(IPA).EQ.0) GO TO 3702
      XPRQ(1)=X(K)-SPLENG
      IF(X(L).LT.X2DPRQ(IDPRQ))GOTO 1018
      XPRQ(2)=X2DPRQ(IDPRQ)-SPLENG
      INDPRQ=2
 1018 XPRQ(INDPRQ+1)=X(L)-SPLENG
      DO 3402 IQDEV=1,INDPRQ
      IQDEV1=IQDEV+1
      IIQDEV=IDPRQ+IQDEV-1
      DO 3401 IQDCO=1,NQDCO
      QDCOI(IQDCO)=QDCO(IIQDEV,IQDCO)
 3401 QLDCOI(IQDCO)=QLDCO(IIQDEV,IQDCO)
      QDEV=FQDEV(QDCOI,NQDCO,XPRQ(IQDEV),XPRQ(IQDEV1))+QDEV
 3402 QLDEV=FQDEV(QLDCOI,NQDCO,XPRQ(IQDEV),XPRQ(IQDEV1))+QLDEV
      QDEV=QDEV/H
      QLDEV=QLDEV/H
 3702 CONTINUE
```

```
      QALIN=QLINM*QLDEV/LENGTH
      DO 6670 NS=NSTR1,NSTOT
      NP=NPIN(NS)
      DO 6670 M=1,NP
 6670 DTIEAV(NS-NSTR,M)=0.
      XM=(X(L)+ X(K))*0.5+STLEN
      XMHE=XM-UNHLE
      IF(NSUBDH.EQ.0)WRITE(6,8504)
 8504 FORMAT(1H1)
      WRITE(6,15)K,H,XM
   15 FORMAT(    5X,'AXIAL SECTION NR.',I4,5X,'( SECTION LENGTH=',F10.5,
     *'; HEIGHT=',F10.5,' )')
      H1=H/LENGTH
      TMS=T(K)
      QRTOT=QTOT*QDEV*H1
      QSTOT=QLINM*QLDEV*H1*PERLT
      IF(IPHUD .EQ. 1) QSTOT=HUDFAC*(TMS-TAMB)*H
      DELTAH=(QRTOT+QSTOT)/MFLOW
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979
      RHO1=RHO(P(K),T(K))
      IF(NSPACT.EQ.0)GOTO 16
      IF(X(K).LT.DIST(ISPAC))GOTO 4437
      IF(IPAFD.LE.MSPAC)IPAFD=IPAFD+1
      IF(ISPAC.EQ.NSPACT)GOTO 16
      ISPAC=ISPAC+1
 4437 CONTINUE
      IF(X(L).LT.DIST(ISPAC))GOTO 16
      INDSP(K)=2
      I1SPAC= ISPAC-II+1
      WRITE(6,4440)ISPAC
 4440 FORMAT(1H+,80X,'SPACER NR.',I3,' IS PRESENT',24('.')/5X,21('-'))
      IF(K.EQ.1)GOTO 8500
      GOTO 17
   16 INDSP(K)=1
      SBMNS=MFLOW/ATOT*ASEC
      WRITE(6,4441)
 4441 FORMAT(1H+,78X,50('.')/5X,21('-'))
   17 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WRITE(6,820) TIME2
  820 FORMAT(/5X,'TIME = ',F10.5,' SEC.')
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 4444 NS=1,NSTOT
      SIGMA(NS)=0.
      PHI(NS)=0.
      NP=NPIN(NS)
      DO 4443 M=1,NP
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(NS.GT.NSTR) SHQ(NS-NSTR,M)  = 0.0
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      MSCH(NS,M)=MSCH1(NS,M)
      TSCH(NS,M)=TSCH1(NS,M)
      IF(K.GT.1 .AND. NSUBDH.EQ.0)TWP(NS,M)=TW(NS,M)
      SIGMAI(NS,M)=0.
      PHII(NS,M)=0.
      IF(NS.LE.NSTR)GOTO 4443
      DO  4442 JWC=1,2
CCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(K.GT.1 .AND. NSUBDH .EQ.0)
     >          TWWP(NS,M,JWC)=TWWC(NS-NSTR,M,JWC)
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      CHI(NS-NSTR,M,JWC)=1.
      PSI(NS-NSTR,M,JWC)=1.
 4442 CONTINUE
 4443 CONTINUE
 4444 CONTINUE
```

```
      ITGLT=0
      DELTAP=0.
      T(L)=DELTAH/CP(P(K),T(K))+T(K)
      PBT=P(K)
      CALL LINPOW
C     ..................................................................
C
C  15-THE LOOP ITCORR STARTS
C
      ITCON =0
      DO 49 ITCOR1=1,ITCM
      ITCORR=ITCOR1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979
      IF(IPHUD.EQ.1) QSTOT=HUDFAC*(TMS-TAMB)*H
      DELTAH=(QTOT*QDEV*H1+QSTOT)/MFLOW
      IF(INDSP(K).EQ.2 .AND. ITCORR.GT.2)GOTO 45
C
      LAMBDA(K)=LAM1
      DDDDT=0.
C*****CALCULATION OF DELTAP AND DELTAT FOR THE WHOLE BUNDLE FLOW SECT.*
      DO 4448 ITTE1=1,10
      TL=T(L)
      TBT=(T(L)+T(K))*0.5
      DO 4445 ITTE2=1,10
      TBT1=TBT
      TBT=DELTAH/CP(PBT,TBT)*0.5+T(K)
      IF(ABS(TBT/TBT1-1.).LE.1.E-04)GOTO 4447
 4445 CONTINUE
      WRITE(6,4446)TBT,TBT1
 4446 FORMAT(1H1,'MAIN',/5X,
     >           'CALCULATION STOPS: ITTE2=10 ; TBT=',E15.7,5X,'TBT1=',
     *    E15.7)
      STOP
 4447 CONTINUE
       T(L)=DELTAH/CP(PBT,TBT)+T(K)
      DO 18 ITPR=1,10
      DP=DELTAP
      PBT=P(K)+0.5*DP
      P(L)=P(K)+DP
      RHOBT(K)=RHO(PBT,TBT)
      RHO2=RHO(P(L),T(L))
      DEL1RT=(RHO1-RHO2)/RHOBT(K)**2
      DECORR=DETOLA/FCOPWT
      DELTAP=PROV2*(LAMBDA(K)*H/(2.*RHOBT(K)*DECORR)+DEL1RT)*(ASEC/
     /ASECLA)**2+IGRAV*RHOBT(K)*H*0.001
      ETABT(K)=ETA(PBT,TBT)
      REBT(K)=PROV1/ETABT(K)*DETOLA/DETOT*ASEC/ASECLA
CCCCCCCCCCCCCCCCCCCCCCCCCCCC 08.02.1980
      RES=REBT(K)
      PRS=ETA(PBT,TBT)*CP(PBT,TBT)/KAPPA(PBT,TBT)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(INDSP(K).EQ.2)DELTAP=DELTAP+PROV2*(CSPT(I1SPAC)+DSPDPF(EPSIT(
     1 I1SPAC),DETOLA,LAMBDA(K),WSP(IPA),PGDPT(I1SPAC),REBT(K),4))/
     2  RHOBT(K)
      PLL=P(K)+DELTAP
      IF(ABS(PLL/P(L)-1.).LE.1.E-05)GOTO 20
   18 CONTINUE
      WRITE(6,19)K,ITCORR,DP,DELTAP
   19 FORMAT(1H1,5X,'MAIN',/5X,
     >              'CALCULATION STOPS: ITPR=10 FOR SECTION',I4,2X,'(ITC
     *ORR=',I2,') DP=',E20.5,5X,'AND DELTAP=',E20.5)
      STOP
   20 CONTINUE
       T(L)=DELTAH/CP(PBT,TBT)+T(K)
      IF(ABS(T(L)/TL-1.).LE.1.E-04)GOTO 4450
 4448 CONTINUE
```

```
      WRITE(6,4449)T(L),TL
 4449 FORMAT(1H1,5X,'MAIN',/5X,
     >              'CALCULATION STOPS: ITTE1=10 ; T(L)=',E15.7,5X,
     *   'TL=',E15.7)
      STOP
 4450 CONTINUE
      UBT(K)=MA/RHOBT(K)/100. *ASEC/ASECLA
      PROV=REBT(K)*ETABT(K)*SQRT(ABS(LAMBDA(K))*0.125/RHOBT(K))
      SQDPG=SQRT(ABS(DELTAP)*980665.)
      SQDPGF=SQRT(ABS(DELTAP-IGRAV*RHOBT(K)*H*0.001)*980665.)
      IF(ITCORR.EQ.1)DPSI=DELTAP/ABS(DELTAP)
      SIGMST=(SQRT(ABS(DELTAP-IGRAV*RHOBT(K)*H*0.001)*980665.)-SQDPG)/
     /SQRT(LAMBDA(K)*H/(2.*DETOLA*RHOBT(K)))
      IF(INDSP(K).EQ.2)GOTO 45
C     ..................................................................
C
C     FOR SECTIONS WITHOUT SPACERS: SUB-SUBCHANNEL CALCULATION
C
      DO 6671 NS=NSTR1,NSTOT
      NP=NPIN(NS)
      DO 6671 M=1,NP
      DO 6671 I=1,NSC90
 6671  DELTIE(NS-NSTR,M,I)=DELTIO(NS-NSTR,M,I)-DTIEAV(NS-NSTR,M)
      ASECLA=0.
      DETOLA=0.
      DO 29 NS=1,NSTOT
      IF(ITCORR.EQ.1)SIGMA(NS)=SIGMST
      IF(NTYP(NS).EQ.3)GOTO 25
      NP=NPIN(NS)
      DDDDNS=0.
      TNS=0.
      AMNS=0.
      DO 24 M=1,NP
      IF(ITCORR.EQ.1)SIGMAI(NS,M)=SIGMST
      IF(NTYP(NS).EQ.2)GOTO 22
C
C*****SUB-SUBCHANNEL CALCULATION FOR THE CENTRAL CHANNELS**************
C
      CALL TRICA1(K,NS,NSC30C,IRH,PROV,PBT,RH,ACW,DECW,MEC,AAC,DETC,DETO
     *T,H1,ALFACE,H,M,P(K),P(L),SQDPG,TE1,SUR,D,AMT,DDDD,ATSCH,&8500,C)
      AMSCH=AMT*ASCH(NS,M)/AAC
      GOTO 23
C
   22 CONTINUE
C*****SUB-SUBCHANNEL CALCULATION FOR THE WALL CHANNELS*****************
      CALL RECCA1(K,NS,NSC90,NSC45,IRH,PROV,PBT, RH,H1,ALFAWA,ACW,DECW,M
     *EC,AAW,DETW,ATOT,DETOT,MFLOW,WW,D,C,M,NSTR,H,P(K),P(L),SQDPG,TE1,
     *SUR,AMT,DDDD,ATSCH,CTU3,EM1,&8500,ALFACE)
      NSW=NS-NSTR
      IF(K.GT.1)GOTO 4455
      DO 4451 JWC=1,2
      IF(IREAD1.EQ.1 .OR. ISTAIN.EQ.1)MSCWC1(NSW,M,JWC)=MSCH1(NS,M)/ASCH
     *(NS,M)*ASCHWC(NSW,M,JWC)/F2ATIP(NS,M)
 4451 ASCWC1(NSW,M,JWC)=ASCHWC(NSW,M,JWC)
 4455 CONTINUE
      AMSCH=AMT
   23 CONTINUE
      DDDDNS=DDDDNS+AMSCH/AMT*DDDD+ASCH(NS,M)*(SIGMAI(NS, M)-SIGMA(NS))/
     *SQDPGF
      AMNS=AMNS+AMSCH
   24 TNS=TNS+ATSCH*AMSCH
      TNS=TNS/AMNS
      RHONS=RHO(PBT,TNS)
      LAM(NS)=((A(NS)/DDDDNS)**2)*2.*DE(NS)*RHONS/H *F1ATIP(NS)**2*
     *F1DTIP(NS)
```

```
      GOTO 28
C
C*****SUB-SUBCHANNEL CALCULATION FOR THE CORNER CHANNELS***************
   25 CONTINUE
      IF(ITCORR.EQ.1)SIGMAI(NS,1)=SIGMST
      CALL ANGCA1(K,NS,NSC30A,IRH,PROV,PBT,RH,H1,ALFACO,AA1,AAA,DETA,DET
     *OT,D,WA,NSTR,H,P(K),P(L),SQDPG,TE1,SUR,    AMT,DDDDNS,&8500,AMAT,
     2AMBT)
      AMNS=AMT*ASCH(NS,1)/AAA
      XMSCHA(NS-NSTR,1)=AMAT*ASCH(NS,1)/AAA
      XMSCHB(NS-NSTR,1)=AMBT*ASCH(NS,1)/AAA
      DDDDNS=DDDDNS*ASCH(NS,1)/AAA
      LAM(NS)=LAMSCH(NS,1)
   28 CONTINUE
      ASECLA=ASECLA+A(NS)*F1ATIP(NS)
       DETOLA=DETOLA+A(NS)/DE(NS)*F1ATIP(NS)/F1DTIP(NS)
      MO(NS)=2.*AMNS-MI(NS)
      DDDDT=DDDDT+DDDDNS
   29 CONTINUE
      DETOLA=ASECLA/DETOLA
C     ................................................................
C
C  16-NEW VALUE FOR THE WHOLE BUNDLE FRICTION FACTOR
C
      IF(             ITCORR.EQ.1)GOTO 48
      DDDDT=DDDDT*(MFLOW*ASEC)/(SBMNS*ATOT)
      LAM1=((ASECLA/DDDDT)**2-DEL1RT)*2.*DETOLA*RHOBT(K)/H
      DPSI= DPAV/ABS(DPAV)
C     ................................................................
C
C  17-CONVERGENCE TEST FOR THE LOOP ITCORR
C
   45 CONTINUE
      IF(ITCORR.LE.ITC2)GOTO 48
      DELAM=ABS(LAMBDA(K)/LAM1-1.)
      CALL CONTRO(LAMBDA(K),LAM1,ITCORR,INDICE)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.11.1980
C     IF(.NOT.(DELAM.LE.1.E-04 .OR. (DELAM.LE.1.E-03 .AND. ITCORR.GT.
C    *   ITC1) .OR. (DELAM.LE.1.E-02 .AND. ITCORR.GT.(ITC1+5))))GOTO 48
      IF(.NOT.(DELAM.LE.1.E-01 .OR. (DELAM.LE.1.E-01 .AND. ITCORR.GT.
     *   ITC1) .OR. (DELAM.LE.1.E-01 .AND. ITCORR.GT.(ITC1+5))))GOTO 48
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.11.1980
CCCCCCCCCCCCC
      ITCON=ITCON+1
      IF(INDICE.GE. 1  )GOTO 48
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     ................................................................
C
C  18-CONVERGENCE HAS BEEN REACHED: PRINT AND PUNCH RESULTS FOR SECT. K
C
C
 9003 WRITE(6,46)
     *  T(L),    P(L),PBT,DELTAP,LAMBDA(K),ITCORR,ITGL,ITGLT,ITERM,FREL,
     >  ITCON,QDEV
   46 FORMAT(/5X,'T 2=',F10.4,5X,'P 2=',F10.6,5X,'P AV=',F10.6,5X,
     * 'DELTAP=',F11.8,5X,'LAMBDA=',F7.5/5X,'( ITCORR=',I2,
     *        5X,' ITGL=',I5,5X,'ITGL1=',I5,5X,'ITERM=',I5,5X,
     *  'FREL=',F5.2,5X,'ITCON = ',I3,5X,'QDEV=',E12.6,
     *              ' )'///     5X,'CHANNEL',9X,'OUTLET MASS',8X,'AVERA
     *GE MASS',7X,'OUTLET TEMP.',8X,'AVERAGE TEMP.',7X,'PRESSURE LOSS'/)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(IPRINT.LE.0) GO TO 2021
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WRITE(6,81)(NS,MM2(NS),MAV(NS),TEMP2(NS),TAV(NS),DPNS(NS),
     *  NS=1,NSTOT)
   81 FORMAT(I12,5E20.8)
```

```
      WRITE(6,83)
      DO 80 NS=1,NSTOT
      WRITE(6,78)NS,UAV(NS),NS,WCF(NS)
      IF(INDSP(K).EQ.1)WRITE(6,79)NS,MO(NS),NS,LAM(NS)
   78 FORMAT(5X,'UAV(',I3,')=',E13.5,10X,'WCF(',I3,')=',E12.3)
   79 FORMAT(1H+,T70,'MO(',I3,')=',F10.2,10X,'LAM(',I3,')=',F10.5)
   80 CONTINUE
      WRITE(6,83)
   83 FORMAT(/ )
      DO 85 NS=1,NSTOT
      NI=NER(NS)
      WRITE(6,84)(M,NS,NIS(NS,M),WT(NS,M),M=1,NI)
   84 FORMAT(3(5X,I2, ') WT(',I4,',',I4,')=',E12.3))
   85 CONTINUE
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 2021 CONTINUE
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      XLTOT=X(L)+STLEN
       PBAR(L)=P(L)*0.980665
      DPBAR(L)=PEOBAR-PBAR(L)
      IF(IPUNCH.EQ.1)WRITE(1,1)IPA
      IF(IPUNCH.EQ.1)WRITE(1,6069)XLTOT,DPBAR(L)
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      WRITE (2) XLTOT,DPBAR(L)
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
C     .................................................................
C
C  19-CORRECTION OF THE COMPUTED SURFACE PIN TEMPERATURES FOR THE BIOT
C     EFFECT AND FOR THE RADIAL POSITION OF THE THERMOCOUPLES
C
      IF( QTOT.LE.1.E-06)GOTO 50
      DO 53 NS=1,NSTOT
      NSW=NS-NSTR
      NP=NPIN(NS)
      DO 51 M=1,NP
      TWINF(NS,M)=TW(NS,M)
      IF(QQ(NS,M).LT.1.E-06)GOTO 51
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     QHRDAR=QQ(NS,M)*QDEV*HRDAR                   OLD
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                            26.03.1979
      JP=JPIN(NS,M)
      QAD=QJ(JP,NS)/H1*GEO1(NS,M)
      QF=QQ(NS,M)*QDEV+QAD
      QHRDAR=QF*HRDAR
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      RVOL=D*0.5
CCCCCCCCCCCCCCCCCCCCCCCCCC
C     IF(I2TIP(NS,M).EQ.1.AND.IPA.EQ.4                )RVOL=RTIP(4)-RH
      IF(I2TIP(NS,M).EQ.1.AND.IPA.EQ.4.AND.LAMOP2.EQ.1)RVOL=RTIP(4)-RH
CCCCCCCCCCCCCCCCCCCCCCCCCC
      FCORTW=((RVOL**2-RMISTW(IPA)**2)*0.5+RINT(IPA)**2*ALOG(RMISTW(IPA)
     */RVOL))/((RVOL**2-RINT(IPA)**2)*SUR)*RVOL
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     QRMDAR=QQ(NS,M)*QDEV*FCORTW                OLD
C     IF(IPA.EQ.4)QLAMR=QQ(NS,M)*QDEV*FCORLA
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                          07.03.1979
      QRMDAR=QF*FCORTW
      IF(IPA.EQ.4)QLAMR=QF*FCORLA
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      CALL CORRTE(TW(NS,M),TSCH(NS,M),PBT,    NS,M,0,BIOT(NS,M),TWINF(NS,
     *M))
      CALL CORRTE(TWSSC1(NS,M),TBSSC1(NS,M),PBT,    NS,M,1,BIOTI1,TWINF1)
      CALL CORRTE(TWSSC2(NS,M),TBSSC2(NS,M),PBT,    NS,M,2,BIOTI2,TWINF2)
```

```
      IF(NTYP(NS).NE.2)GOTO 51
      DO 3721 JWC=1,2
       CALL CORRTE(TWWC(NSW,M,JWC),TAVWC(NSW,M,JWC),PBT,   NS,M,JWC,BIOT
     *WC,TWINWC)
 3721 CONTINUE
   51 CONTINUE
   53 CONTINUE
CCCCCCCCCCCC 28.02.1981
      IF(IPRINT.LE.0) GO TO 1949
      IF(IPA/2*2.EQ.IPA) CALL JELLA(-1)
      IF(IPA/2*2.EQ.IPA) CALL JELLB(-1,PBT)
      IF(IPA.EQ.4 .AND. LAMOP3.NE.1) CALL JELLC(-1,TE,TE1,BIOT)
 1949 CONTINUE
CCCCCCCCCCCC 28.02.1981
      IF(IPUNCH.EQ.1)WRITE(1,6069)XM
      DO 88 NS=1,NSTOT
            NP=NPIN(NS)
            DO 3723 M=1,NP
                    JP=JPIN(NS,M)
                    WRITE (3) NS,M,XM,TW(NS,M)
                    IF (NS.LE.NSTR) GO TO 3723
                    NSW=NS-NSTR
                    WRITE (4) NS,M,XM,TLINER(NSW,M)
                    DO 62 JWC=1,2
                          WRITE (9) NS,M,JWC,XM,TWWC(NSW,M,JWC)
   62                     CONTINUE
 3723               CONTINUE
C
            IF(IPUNCH.EQ.1) WRITE(1,6069) (TW(NS,M),M=1,NP)
            IF(NS.LE.NSTR)GOTO 88
            NSW=NS-NSTR
            IF(IPUNCH.EQ.1)WRITE(1,6069)(TLINER(NSW,M),M=1,NP)
 6069       FORMAT( 3E15.5)
   88       CONTINUE
      GOTO 50
C     ..............................................................
C
C  20-CALCULATION IN THE CHANNELS, IN THE SUBCHANNELS AND IN THE TWO
C     PORTIONS OF THE WALL SUBCHANNELS
C
   48 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL TTOT(1)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL BALA(K,NSTOT,INDSP(K),ASECLA,H,LENGTH,P(K),P(L),PBT,FREL,FT,
     *ITCORR,ITCM,DPAV,ITERM,ITGL,&8500,WSP(IPA),I1SPAC)
      ITGLT=ITGLT+ITGL
      CALL SUBBAL(NSTOT,NSTR,INDSP(K),H,LENGTH,D,PIG,P(K),P(L),PBT,FREL,
     *FT,ITCORR,DPAV,&8500,WSP(IPA),I1SPAC)
      CALL NORMT(NSTOT,NSTR,TBT,ATOT,ASEC,MFLOW)
      CALL TEMCON(IRH,K,RH,SUR,D,PIG,TE1,PBT,&8500,H,VDIAM(IPA),
     >              RINT(IPA),INDICE,QTOT)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
  879 NS1=NSTR+1
      DO 64 NS=NS1,NSTOT
      NP=NPIN(NS)
      NSTYP=NTYP(NS)
      NSW=NS-NSTR
      DO 7034 M=1,NP
      GOTO(7033,7033,7032),NSTYP
 7032 PROVI(NSW,M)=MSCH(NS,M)*DE(NS )/ASCH(NS,M)*SQRT(LAMSCH(NS,M)*0.125
     */RHO(PBT,TSCH(NS,M)))
      GOTO 7034
 7033 PROVI(NSW,M)=MAWC(NSW,M,1)*DEWC(NSW,M,1)/ASCHWC(NSW,M,1)*
     *SQRT(LAMWC(NSW,M,1)*0.125/RHO(PBT,TAVWC(NSW,M,1)))
 7034 CONTINUE
```

```
   64 CONTINUE
   49 CONTINUE
C     ..............................................................
C
C  21-END OF THE LOOP ITCORR
C
      WRITE(6,56)ITCORR,LAMBDA(K),LAM1,DELTAP,DPAV,DELAM
   56 FORMAT(1H1,'CALCULATION STOPS: ITCORR=',I5/5X,'( LAMBDA=',E15.7,5X
     *,'LAM1=',E15.7,5X,'DELTAP=',E15.7,5X,'DPAV=',E15.7,' )',5X,
     > 'DELAM = ',E12.5)
      WRITE(6,59) ITCON,INDICE
   59 FORMAT( 5X,'ITCON = ',I2,'    INDICE = ',I3,
     > //5X,'PIN TEMPERATURES : LAST ITERATION, WITHOUT CONDUCTION, WITH
     > CONDUCTION.')
      CALL JELLA(-1)
      STOP
C     ..............................................................
C
C  22-DEFINITION OF THE INLET VALUES OF CHANNEL AND SUBCHANNEL
C     VARIABLES FOR THE NEXT AXIAL SECTION; SUBSEQUENT ADDITIONS FOR
C     AVERAGE VALUES OF VARIABLES
C
   50 CONTINUE
      NSUBDH=0
      INDTW=1
      PBAR(L)=P(L)*0.980665
      DO 100 NS=1,NSTOT
      TEMP(NS)=(2.*MAV(NS)*TAV(NS)-MI(NS)*TEMP (NS))/MM2(NS)
      MI(NS)=MM2(NS)
      NP=NPIN(NS)
      DO 97 M=1,NP
      PMSCH1=MSCH1(NS,M)
      MSCH1(NS,M)=2.*MSCH(NS,M)-MSCH1(NS,M)
      TSCH1(NS,M)=(2.*MSCH(NS,M)*TSCH(NS,M)-PMSCH1*TSCH1(NS,M))/MSCH1(NS
     *,M)
      IF(NTYP(NS).NE.3 .OR. INDSP(K).EQ.2)GOTO 6647
      DO 6646 I=1,NSC90
 6646 DELTIO(NS-NSTR,M,I)=(TIO(NS-NSTR,M,I)-TSCH1(NS,M))*TCPRCF
 6647 CONTINUE
      IF(QQ(NS,M).GT.1.E-06)GOTO 5633
      TW(NS,M)=TSCH(NS,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      NCEV=NS-NSTR
      IF(NS.GT.NSTR) TLINER(NCEV,M)=TSCH(NS,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      TEMPB(NS,M)=TSCH(NS,M)
      BIOT(NS,M)=0.
      TWINF(NS,M)=TSCH(NS,M)
 5633 CONTINUE
      IF(K.EQ.1 .OR. NEXTW(IPA).EQ.0)GOTO 1180
      IF(X(K).GT.XEXTW(IEXTWC).AND.X(K-1).LE.XEXTW(IEXTWC))GOTO 1181
      IF(K.EQ.N .AND. X(N+1).GT.XEXTW(IEXTWC) .AND. X(N).LE.XEXTW(IEXTWC
     *))GOTO 1181
      GO TO 1180
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C1181 TWTH(NS,M,IEXTWC)=(TW(NS,M)-TWP(NS,M))/(X(K+1)-X(K-1))*2.*(XEXTW(
C    *IEXTWC)-0.5*(X(K)+X(K-1)))+TWP(NS,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
 1181 CONTINUE
      TWTH(NS,M,IEXTWC)=(TW(NS,M)-TWP(NS,M))/(X(K+1)-X(K-1))*2.*(XEXTW(
     *IEXTWC)-0.5*(X(K)+X(K-1)))+TWP(NS,M)
      IF(NTYP(NS) .NE. 2) GO TO 1183
      DO 1182 JWC=1,2
      TWWH(NS,M,JWC,IEXTWC)=(TWWC(NS-NSTR,M,JWC)-TWWP(NS,M,JWC))/
     >(X(K+1)-X(K-1))  *2.*(XEXTW(IEXTWC)-0.5*(X(K)+X(K-1)))+
```

```
     >TWWP(NS,M,JWC)
 1182 CONTINUE
 1183 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      INDTW=2
 1180 CONTINUE
      IF(NTYP(NS).NE.2)GOTO 97
      NSW=NS-NSTR
      DO 95 JWC=1,2
      PMSCWC=MSCWC1(NSW,M,JWC)
      MSCWC1(NSW,M,JWC)=2.*MAWC(NSW,M,JWC)-PMSCWC
      TSCWC1(NSW,M,JWC)=(2.*MAWC(NSW,M,JWC)*TAVWC(NSW,M,JWC)-PMSCWC*
     *   TSCWC1(NSW,M,JWC))/MSCWC1(NSW,M,JWC)
      ASCWC1(NSW,M,JWC)=ASCHWC(NSW,M,JWC)
   95 CONTINUE
      IF(INDSP(K).EQ.2)GOTO 97
      DO 6648 I=1,NSC90
 6648 DELTIO(NSW,M,I)=(TIO(NSW,M,I)-TSCWC1(NSW,M,1))*TWPRCF
   97 CONTINUE
  100 CONTINUE
      IF(INDTW.EQ.2 .AND. IEXTWC.LT.NEXTWT)IEXTWC=IEXTWC+1
      IF(INDPRQ.EQ.2 .AND. IDPRQ.LT.NDPRQT)IDPRQ=IDPRQ+1
      TWTC=TWFUN(NRODS,NSTOT,PIG,AAC,AAA)*H+TWTC
      TBPC=TBFUN(NSTR,NSTOT)*H+TBPC
      TBTC=TBT*H+TBTC
      IF(X(K).LT.XSTART(IPAFD) .OR. X(L).GT.XEND(IPAFD))GOTO 103
      NSEFD=NSEFD+1
      TM=TM+TBT*H
      PM=PM+PBT*H
      LAMBDM=LAMBDM+LAMBDA(K)*H
      REM=REM+REBT(K)*H
      UM=UM+UBT(K)*H
      DELTAX=DELTAX+H
      IF(IRH.EQ.1)GOTO 103
      DO 9899 NS=1,NSTOT
      NP=NPIN(NS)
      DO 9899 M=1,NP
      HPLUS1(NS,M)=HPLUS1(NS,M)+HPLUSB(NS,M)*H
      HPLUS2(NS,M)=HPLUS2(NS,M)+HPLUSW(NS,M)*H
      QPLUSA(NS,M)=QPLUSA(NS,M)+QPLUS(NS,M)*H
      PRBA (NS,M)=PRBA (NS,M)+PRB (NS,M)*H
      TWA(NS,M)=TWA(NS,M)+TW(NS,M)*H
      YDHA(NS,M)=YDHA(NS,M)+YDH(NS,M)*H
      YODHA(NS,M)=YODHA(NS,M)+YODH(NS,M)*H
      AMASSB(NS,M)=AMASSB(NS,M)+XMASSB(NS,M)*H
      TEMPBA(NS,M)=TEMPBA(NS,M)+TEMPB(NS,M)*XMASSB(NS,M)*H
      AMASST(NS,M)=AMASST(NS,M)+MSCH(NS,M)*H
      TEMPTA(NS,M)=TEMPTA(NS,M)+TSCH(NS,M)*MSCH(NS,M)*H
 9899 CONTINUE
  103 CONTINUE
C     .................................................................
C
C  23-PRINT SUBCHANNEL VARIABLES
C
      IF(IPRINT.LE.0) GO TO 2023
      IF(IPSUB.LE.0)  GO TO 2023
C
      WRITE(6,83)
      DO 8887 NS=1,NSTOT
      NTYPNS=NTYP(NS)
      NP=NPIN(NS)
      NSW=NS-NSTR
      WRITE(6,8885)NS
 8885 FORMAT(5X,'CHANNEL NR.',I5)
      DO 8887 M=1,NP
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  OLD
```

```
C     IF(QQ(NS,M).GT.1.E-06)SCNUSS=QQ(NS,M)*QDEV*DE(NS)*F2DTIP(NS,M)/
C    /(SUR*(TWINF(NS,M)-TSCH(NS,M))*KAPPA(PBT,TSCH(NS,M)))*D*0.5/
C    /RTIP(IPA)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                   07.03.1979
      IF(QQ(NS,M).LE.1.E-06)  GO TO 70
      NT=NTYP(NS)
      JP=JPIN(NS,M)
      QF=QQ(NS,M)*QDEV/SUR + QJ(JP,NS)/(PIG*D*H/GEO1(NS,M))
      SCNUSS=QF*DE(NS)*F2DTIP(NS,M)/
     /((TWINF(NS,M)-TSCH(NS,M))*KAPPA(PBT,TSCH(NS,M)))*D*0.5/
     /RTIP(IPA)
      ALFAD = SCNUSS * KAPPA(PBT,TSCH(NS,M)) / DE(NS) * 41860.
   70 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SCREB=MSCH(NS,M)*DE(NS)*F2DTIP(NS,M)/(ASCH(NS,M)*F2ATIP(NS,M)*
     *ETA(PBT,TSCH(NS,M)))
      SCREW=SCREB*ETA(PBT,TSCH(NS,M))*RHO(PBT,TWINF(NS,M))/(ETA(PBT,
     *TWINF(NS,M))*RHO(PBT,TSCH(NS,M)))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979  (WRITE 3725)
C     QLINSC=QLINM*QLDEV*PERL(NTYPNS)*ASCH(NS,M)/ACH(NTYPNS)*H1
      QLINSC=0.0
      IF(NS.GT.NSTR) QLINSC=SHQ(NSW,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979
      WRITE(6,8886)M,JPIN(NS,M),MSCH1(NS,M),TSCH1(NS,M),LAMSCH(NS,M)
     *,SCREB,SCREW
 8886 FORMAT(5X,I2,'-(ROD NR.',I4,')',T27,'OUT. MASS',F10.6,T52,'OUT. TE
     1MP.=',F7.2,T75,'LAMBDA=',F10.5,T99,'REB=',F7.0,T112,'REW=',F7.0)
      WRITE(6,3725) QLINSC
 3725 FORMAT(T27,'Q LINER=',E15.5)
      IF(IRH.EQ.2 .AND. QTOT.GT.1.E-06 .AND. I2TIP(NS,M).NE.1)
     *                                 WRITE(6,3724)BIOT(NS,M)
 3724 FORMAT(1H+,T52,'BIOT=',F10.5)
      IF(QTOT.GT.1.E-06)WRITE(6,3722) TWINF(NS,M)
 3722 FORMAT(1H+,                     T75,'TW INF.=',F10.2)
      IF(QTOT.GT.1.E-06 .AND. NTYP(NS).NE.1)WRITE(6,3740)TLINER(NSW,M)
 3740 FORMAT(1H+,T99,'T  AT LINER=',F10.2)
        IF(INDSP(K).EQ.2)GOTO 91
      IF(IRH.EQ.1 .OR. I2TIP(NS,M).EQ.1)GOTO 3726
      RHPLM=RHPLUS(HPLUSB(NS,M),TW(NS,M),TE1 ,QPLUS(NS,M),HPLUSW(NS,M),
     *TEMPB(NS,M),YODH(NS,M))
      WRITE(6,8883)HPLUSB(NS,M),HPLUSW(NS,M),RHPLM
 8883 FORMAT(T27,'HB+ =',E12.5,T52,'HW+ =',E12.5,T75,'R( H+ )=',E12.5)
      IF(QQ(NS,M).LE. 1.E-06)GOTO 91
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 09.10.1980
C     GHPLM=GHPLUS(HPLUSW(NS,M),TW(NS,M),TSCH(NS,M),PRB (NS,M),YDH(NS,M)
C    1,10000.,0.)
      GHPLM=GHPIU(NS,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 09.10.1980
      TWDTEM=(TW(NS,M)+273.16)/(TE+273.16)
      TWDTBM=(TW(NS,M)+273.16)/(TSCH(NS,M)+273.16)
      PHIM=GHPLM/(PRB (NS,M)**O4 * TWDTBM**O5)*(O16*YDH(NS,M))**O17
      WRITE(6,94)GHPLM,SCNUSS      ,TWDTBM,TWDTEM,YDH(NS,M),O4,O5,O16,
     1     O17,PHIM
   94 FORMAT(1H+,T99,'G( HW+ )=',E12.5/T27,'NU =',E13.6,T52,'TW/TB=',E13
     1.5,T75,'TW/TE=',E13.5,T99,'Y/RH=',E13.5/T27,'G( HW+ )/( PR**',F4.2
     2,' * (TW/TB)**',F4.2,' ) * (',F6.3,'*Y/RH)**',F6.3,' =',E13.6)
 3726 IF(QQ(NS,M).LE.1.E-06)GOTO 91
      IF(IRH.EQ.1 .OR. I2TIP(NS,M).EQ.1)WRITE(6,4242)SCNUSS
 4242 FORMAT(1H+,T52,'NU =',E13.6)
      WRITE(6,6685)TBSSC1(NS,M),TWSSC1(NS,M),TBSSC2(NS,M),TWSSC2(NS,M)
 6685 FORMAT(T27,'TBSSCH(1)=',F7.2,T52,'TWSSCH(1)=',F7.2,T75,'TBSSCH(N)=
     1',F7.2,T99,'TWSSCH(N)=',F7.2)
      IF(NTYP(NS).EQ.1)GOTO 91
      WRITE(6,6640)TTSCHA(NSW,M),TTSCHB(NSW,M),TEMPB(NS,M)
```

```
 6640 FORMAT(T27,'TA=',F7.2,T52,'TB=',F7.2,T75,'TBC=',F7.2)
      IF(NTYP(NS).EQ.2)WRITE(6,6644)TWWC(NSW,M,1),TWWC(NSW,M,2)
 6644 FORMAT(T27,'TW(1)=',F7.2,T52,'TW(2)=',F7.2)
      WRITE(6,6645)T1SSC1(NSW,M),T2SSC1(NSW,M),T1SSC2(NSW,M),
     *             T2SSC2(NSW,M)
 6645 FORMAT(T27,'T1SSCH(1)=',F7.2,T52,'T2SSCH(1)=',F7.2,T75,'T1SSCH(N)=
     1',F7.2,T99,'T2SSCH(N)=',F7.2)
   91 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ALFAW=ALFA(NS,M)*41860
      IF(QQ(NS,M).GT.1.E-06)  WRITE(6,8712) ALFAD,ALFAW
 8712 FORMAT(T27,'ALFAD = ',E12.6,T52,'ALFA = ',E12.6)
      IF(QQ(NS,M).LE.1.E-06 .AND. QTOT.GT.1.E-06)  WRITE(6,8773) ALFAW
 8773 FORMAT(T27,' ALFA = ',E12.6)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(NTYP(NS).NE.2)GOTO 8887
      WRITE(6,90)(JWC,MSCWC1(NSW,M,JWC),JWC,TSCWC1(NSW,M,JWC),JWC,ASCWC1
     *(NSW,M,JWC),JWC,LAMWC(NSW,M,JWC),JWC=1,2)
   90 FORMAT(T27,'MOUT(',I1,')=',E13.6,T52,'TOUT(',I1,')=',E13.6,T75,
     1       'AREA(',I1,')=',E13.6,T99,'LAMBDA(',I1,')=',E13.6)
 8887 CONTINUE
 2023 CONTINUE
      IF( IPA/2*2 .EQ. IPA) CALL HEATBA(0,PBT,INDQ,TE,MFLOW)
 9999 CONTINUE
      GOTO 8499
C     ...............................................................
C
C  24-POINT REACHED IN THE CASE OF CONVERGENCE PROBLEMS (LOOP K ENDS
C     AT LABEL 9999)
 8500 CONTINUE
      NSUBDH=NSUBDH+1
      IF(NSUBDH.LE.MSUBDH)GOTO 8502
      WRITE(6,8501)MSUBDH
 8501 FORMAT(//' STOP DUE TO REACHED MAXIMUM NUMBER OF SUBDIVISIONS FOR
     *AXIAL PITCH: NSUBDH=',I2)
      STOP
C
 8502 CALL SUBDH(N,K,K1,NSTOT)
      GOTO 8503
C     ...............................................................
C
C  25-VALUES OF VARIABLES FOR THE WHOLE BUNDLE FLOW SECTION
C
 8499 CONTINUE
      DEPTOT=P(L)-P(1)
      WRITE(6,8889)
 8889 FORMAT(1H1,4X,'VARIABLES FOR THE WHOLE BUNDLE'/5X,30('-')////
     *  5X,'A) INLET VALUES OF TEMPERATURE AND PRESSURE'//5X,'SECTION NR
     *.' ,T26,'HEIGHT (CM)',T43,'TEMPERATURE ( C)',T63,'PRESSURE (KG/SQC
     *M)',T86,'PRESSURE (BARS)'/)
      WRITE(6,8890)(I,X(I),T(I),P(I),PBAR(I),I=1,L)
 8890 FORMAT(7X,I3,15X,F9.4,13X,F7.2,11X,F9.5,12X,F9.5)
      WRITE(6,8891)
 8891 FORMAT(/////5X,'B) VALUES AVERAGED OVER AXIAL SECTIONS'//5X,'SECTI
     *ON NR.',T23,'DENSITY (G/CCM)',T41,'VISCOSITY(G/CM*SEC)',T64,'VELOC
     *ITY (M/SEC)',T85,'REYNOLDS NR.',T99,'FRICTION FACTOR'/)
      WRITE(6,8892)(I,RHOBT(I),ETABT(I),UBT(I),REBT(I),LAMBDA(I),I=1,N)
 8892 FORMAT(7X,I3,17X,F7.5,12X,F9.7,12X,F7.3,11X,F9.2,6X,F7.5)
      WRITE(6,8878)DEPTOT
 8878 FORMAT(///5X,'TOTAL PRESSURE DROP=',F9.6,' KG/SQCM')
C     ...............................................................
C
C  26-EVALUATION AND PRINTING OF AVERAGE VALUES OF VARIABLES FOR THE
C     REGIONS WHERE INDISTURBED FLOW IS ASSUMED
C
      IF(NSEFD.EQ.0)GOTO 8897
```

```
      TM=TM/DELTAX
      PM=PM/DELTAX
      PMBAR=PM*0.980665
      LAMBDM=LAMBDM/DELTAX
      RHOM=RHO(PM,TM)
      ETAM=ETA(PM,TM)
      REM=REM/DELTAX
      UM=UM/DELTAX
      WRITE(6,8893) TM,PM,PMBAR,RHOM,ETAM,UM,REM,LAMBDM
 8893 FORMAT(////5X,'C) TOTAL MEAN VALUES AVERAGED IN PARTS WHERE UNDIST
     *URBED FLOW IS SUPPOSED'//              5X,'TEMPERATURE',T22,'=',F9.
     *2,' C'/5X,'PRESSURE',T22,'=',F9.4,' KG/SQCM   =',F9.4,' BARS'
     *                                        /5X,'DENSITY',T22,'=',
     *  F9.5,' G/CCM'/5X,'VISCOSITY',T22,'=',F9.7,' G/CM*SEC'/5X,
     *'VELOCITY',T22,'=',F9.3,' M/SEC'/5X,'REYNOLDS NR.',T22,'=',F9.2/5X
     *     ,'FRICTION FACTOR',T22,'=',F9.5//)
      IF(IRH.EQ.1)GOTO 8897
      WRITE(6,83)
      DO 8876 NS=1,NSTOT
      NP=NPIN(NS)
      DO 8874 M=1,NP
      IF(I2TIP(NS,M).EQ.1)GOTO 8897
      WRITE(6,8885)NS
      HPLUS1(NS,M)=HPLUS1(NS,M)/DELTAX
      HPLUS2(NS,M)=HPLUS2(NS,M)/DELTAX
      QPLUSA(NS,M)=QPLUSA(NS,M)/DELTAX
      PRBA (NS,M)=PRBA (NS,M)/DELTAX
      TWA(NS,M)=TWA(NS,M)/DELTAX
      YDHA(NS,M)=YDHA(NS,M)/DELTAX
      YODHA(NS,M)=YODHA(NS,M)/DELTAX
      TEMPTA(NS,M)=TEMPTA(NS,M)/AMASST(NS,M)
      TEMPBA(NS,M)=TEMPBA(NS,M)/AMASSB(NS,M)
      RHPLA=RHPLUS(HPLUS1(NS,M),TWA(NS,M),TE1,QPLUSA(NS,M),HPLUS2(NS,M),
     1TEMPBA(NS,M),YODHA(NS,M))
      WRITE(6,8875)M,JPIN(NS,M),HPLUS1(NS,M),HPLUS2(NS,M),RHPLA
 8875 FORMAT(5X,I2,'-(ROD NR.',I4,')',T27,'HB+ =',E12.5,T52,'HW+ =',E12.
     15,T75,'R( H+ )=',E12.5)
      IF(QQ(NS,M).LE.1.E-06)GOTO 8874
      TWDTEA=(TWA(NS,M)+273.16)/(TE+273.16)
      TWDTBA=(TWA(NS,M)+273.16)/(TEMPTA(NS,M)+273.16)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 09.10.980
C     GHPLA=GHPLUS(HPLUS2(NS,M),TWA(NS,M),TEMPTA(NS,M),PRBA (NS,M),
C    1YDHA(NS,M),10000.,0.)
C     PHIA=GHPLA/(PRBA (NS,M)**O4 * TWDTBA**O5)*(O16*YDHA(NS,M))**O17
C     WRITE(6,94)GHPLA,QPLUSA(NS,M),TWDTBA,TWDTEA,YDHA(NS,M),O4,O5,O16,
C    1O17,PHIA
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 09.10.980
 8874 CONTINUE
 8876 CONTINUE
 8897 CONTINUE
C     ..........................................................
C
C  27-COMPARISON BETWEEN THE INPUT AND THE COMPUTED AVERAGE
C     TEMPERATURES OF THE GAS, OF THE SHROUD AND OF THE PINS IN THE
C     AXIAL PORTION
C
      TWTC=TWTC/LENGTH
      TBTC=TBTC/LENGTH
      TBPC=TBPC/LENGTH
      WRITE(6,69) TWTIPA(IPA),TWTC,TBTIPA(IPA),TBTC,TBPIPA(IPA),TBPC
   69 FORMAT(///5X,'COMPARISON OF INPUT TEMPERATURES WITH COMPUTED VALUE
     1S'//T17,'INPUT',T26,'COMPUTED'/5X,'TWTIPA',2F11.2/5X,'TBTIPA',2F11
     2.2/5X,'TBPIPA',2F11.2)
C     ..........................................................
C
```

```
C  28-COMPARISON BETWEEN THE EXPERIMENTAL AND THE COMPUTED PRESSURE
C      LOSSES
C
       IF(NEXPR(IPA).GT.0 .OR. NEXTW(IPA).GT.0)WRITE(6,1023)
  1023 FORMAT(////5X,'COMPARISON WITH EXPERIMENTAL RESULTS'/5X,36('-')//)
       IF(NEXPR(IPA).EQ.0)GOTO 1040
       GOTO (1069,1070),INDPR
  1069 WRITE(6,1072)
       GOTO 1071
  1070 WRITE(6,1073)
  1071 CONTINUE
  1072 FORMAT(5X, '1) PRESSURES (KG/SQCM)'//)
  1073 FORMAT(5X, '1) PRESSURES (BARS)'//)
       IEXPR2=IEXPR1+NEXPR(IPA)-1
       K1=1
       DO 1037 IEXPR=IEXPR1,IEXPR2
       DO 1024 K=K1,N
       K2=K
       IF(XEXPR(IEXPR).GE.X(K) .AND. XEXPR(IEXPR).LT.X(K+1))GOTO 1025
  1024 CONTINUE
       GOTO 1040
  1025 K1=K2
       IF(INDSP(K).EQ.2)GOTO 1026
       KK=K2
       GOTO 1027
  1026 KK=K2-1
       IF(KK.EQ.0)KK=2
  1027 CONTINUE
       GOTO (1028,1029),INDPR
  1028 PR1=P(KK)
       PR2= P(KK+1)
       GOTO 1030
  1029  PR1=PBAR(KK)
       PR2=PBAR(KK+1)
  1030 PTH=(PR2-PR1)/(X(KK+1)-X(KK))*( XEXPR(IEXPR)-X(KK))+PR1
       DPEX=PEX(IEXPR)-PE1
       DPTH=PTH-PE1
       PTMPEX=PTH-PEX(IEXPR)
       DPERR=(DPTH-DPEX)/DPEX*100.
       WRITE(6,1031)IEXPR,XEXPR(IEXPR),PEX(IEXPR),DPEX,PTH,DPTH,PTMPEX,
      *DPERR
  1031 FORMAT(5X,I2,')HEIGHT=',F10.5,' CM',5X,'P EX.=',F10.5,5X,'P EX.-PE
      *1=',F10.7,5X,'P TH.=',F10.5,5X,'P TH.-PE1=',F10.7/33X,'P TH.-P EX.
      *=',F10.7,5X,'(DP TH.-DP EX.)/DP EX. *100 =',F7.3/)
  1037 CONTINUE
  1040 CONTINUE
C      ................................................................
C
C  29-PRINT OF THE PIN TEMPERATURES AT SPECIAL AXIAL POSITIONS
C
       IF(NEXTW(IPA).EQ.0)GOTO 1060
       WRITE(6,1041)
  1041 FORMAT(///5X,'2) COMPUTED ROD TEMPERATURES ( C)'//)
       IEXTW2=IEXTW1+NEXTW(IPA)-1
       DO 1050 IEXTW=IEXTW1,IEXTW2
       WRITE(6,1045)IEXTW,XEXTW(IEXTW)
       DO 1044 NS=1,NSTOT
       NP=NPIN(NS)
       WRITE(6,1046)(M,NS, JPIN(NS,M),TWTH(NS,M,IEXTW),M=1,NP)
  1044 CONTINUE
  1045 FORMAT(/5X,I2,')HEIGHT=',F10.5,' CM'/)
  1046 FORMAT(3(5X,I2,') TW TH.(',I5,',',I5,')=',F10.3,' C'))
CFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
       WRITE(6,1047)
  1047 FORMAT(//5X,'TEMPERATURES OF THE TWO PARTS OF WALL SUBCHANNEL',//)
       DO 1048 NS=1,NSTOT
```

```
          IF(NTYP(NS) .NE. 2) GO TO 1048
          NP=NPIN(NS)
          DO 1043 M=1,NP
             DO 1042 JWC=1,2
                WRITE(6,1049) NS,JPIN(NS,M),JWC,TWWH(NS,M,JWC,IEXTW)
 1042           CONTINUE
 1043     CONTINUE
 1048 CONTINUE
 1049 FORMAT(5X,'TW(',I2,',',I2,',',I1,') = ',F10.3)
CFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
 1050 CONTINUE
 1060 CONTINUE
C     ...............................................................
C
C  30-STARTING VALUES OF VARIABLES FOR THE NEXT AXIAL PORTION
C
      IF(X(L).GT.DIST(ISPAC) .AND. ISPAC.NE.NSPACT)ISPAC=ISPAC+1
      TO=T(L)
      PO=P(L)
      DPBAR(1)=DPBAR(L)
      II=II+NSPAC(IPA)
      IEXPR1=IEXPR1+NEXPR(IPA)
      IEXTW1=IEXTW1+NEXTW(IPA)
      SPRLEN=SPRLEN+PLEN(IPA)
      HH=SPRLEN
      IF(II.GT.1 .AND. IDISP2.EQ.1)DISTOO=DIST(II-1)-SPRLEN
      IF(NDPRQ(IPA).GT.0)IDPRQ=IDPRQ+1
      ISTAIN=1
 8888 CONTINUE
C     ...............................................................
C
C  31-END OF THE LOOP IPA; CALCULATION OF THE PRESSURE RECOVERY AT THE
C     OUTLET OF THE BUNDLE
C
      DEPOUT=-COUT*PROV2/RHO2*0.5
      POUT=PO+DEPOUT
      POBAR=POUT*0.980665
      WRITE(6,8896)DEPOUT,POUT,POBAR,COUT
 8896 FORMAT(////5X,'PRESSURE RECAPTURE DUE TO EXIT=',F7.5,' KG/SQCM',5X
     *   ,'PRESSURE OUTSIDE=',F10.5,' KG/SQCM   =',F10.5,' BARS  ( COUT=
     *',F4.2,' )')
      DPOBAR=PEOBAR-POBAR
      IF(IPUNCH.EQ.1)WRITE(1,6069)DPOBAR
      IF(PEXOUT.LE.1.E-06) GO TO 741
      IF(INDPR.EQ.2)POUT=POBAR
      DPEX=PEXOUT-PE1
      DPTH=POUT-PE1
      DPERR=( DPTH-DPEX)/DPEX*100.
      WRITE(6,1008)PEXOUT,DPTH,DPEX,DPERR
 1008 FORMAT(/5X,'EXP. PRESSURE OUTSIDE=',F10.5/5X,'P TH.-PE1=',F10.7/5X
     *,'P EX.-PE1=',F10.7/5X,'(DP TH.- DP EX.)/DP EX.*100=',F6.3)
  741 CONTINUE
C
      CALL HEATBA(1,PBT,INDQ,TE,MFLOW)
      WRITE(6,740)
  740 FORMAT(/5X,' REGULAR CALCULATIONS END')
  742 STOP
      END
```

```
      FUNCTION AKA(R1DR2,PHI)
C     ------------------------------------------------------------------
C     AKA COMPUTES THE ADDITIONAL FRICTION IN THE LAMINAR HYDRODYNAMIC
C     ENTRANCE LENGTH
C
      IF(PHI.GT.0.002)GOTO 1
      AKA=132.53*PHI/R1DR2**0.013
      RETURN
    1 IF(PHI.GE.0.01755)GOTO 2
      A=0.7982+0.3421*ALOG(PHI)
      GOTO 4
    2 IF(PHI.GE.0.05)GOTO 3
      A=-0.05033+0.1322*ALOG(PHI)
      GOTO 4
    3 IF(PHI.GT.0.1)GOTO 5
      A=-0.4463
    4 B=-0.205*PHI**0.44362
      AKA=EXP(A)*R1DR2**B
      RETURN
    5 AKA=0.64/R1DR2**0.0738
      RETURN
      END
```

```
      SUBROUTINE ALFAC (VDIAM,H)
C ----------------------------------------------------------------------
C     TRANFORMS THE CONVECTION FACTORS ALFA  TO THE COEFFICIENT FALFA.
C
      COMMON /SC11C/ FALFA( 42,3)
      COMMON /SC15C/ ALFA ( 42,3)
      COMMON /SC13C/ GEO1( 42,3)
      COMMON /GASD1/ NSTOT
      COMMON /HEA6 / NPIN( 42),JPIN( 42,3)
      COMMON /DAT  / PIG
C
      DO 200 NS=1,NSTOT
             NP=NPIN(NS)
             DO 100 M=1,NP
                    FALFA(NS,M)=ALFA(NS,M)*PIG*VDIAM*H/GEO1(NS,M)
  100               CONTINUE
  200        CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE ANGCA1(K,NS,N,IRH,PROV,PB,    RH,H1,ALFA,A,AT,DET,DETOT,
     *D,W,NSTR,H,PR1,PR2,SQDPG,TE,SUR,          AMT,DDDD,*, AMA,AMB)
C-----------------------------------------------------------------------
C     SUBROUTINE ANGCA1 CALCULATES FRICTION FACTORS AND APPROXIMATE
C     OUTLET MASS FLOW RATES AND TEMPERATURES FOR CORNER CHANNELS
C
      REAL LAMSCH,LAMB,MSCH1,KAPPA,MSCH,LAMLAM
C
      DIMENSION A(30)
C
      COMMON /WACO1/  XMSCHB( 18,2),XMSCHA( 18,2)
      COMMON /DAT/    PIG
      COMMON /ANG2/   PA(30)
      COMMON /SUB1/   ASCH( 42,3)
      COMMON /SUB2/   TSCH( 42,3),MSCH( 42,3)
      COMMON /SUB3/   ADAB( 18,2),DETB( 18,2)
      COMMON /SUB4/   LAMB( 18,2)
      COMMON /SUB5/   LAMSCH( 42,3)
      COMMON /SUB6/   TSCH1( 42,3)
      COMMON /SUB8/   MSCH1( 42,3)
      COMMON /INPAR/  IPA
      COMMON /SUB22/  TW( 42,3)
      COMMON /SUB23/  HPLUSB( 42,3),HPLUSW( 42,3),
     *                QPLUS( 42,3),PRB( 42,3),YODH( 42,3)
      COMMON /MART/   ITCORR
      COMMON /HEA5/   QQ( 42,3)
      COMMON /LAMIN0/ I2TIP( 42,3)
      COMMON /LAMIN1/ AKAPPA( 42)
      COMMON /LAMIN2/ FATIP(3),FDTIP(3)
      COMMON /LAMIN3/ F1ATIP( 42),F1DTIP( 42)
      COMMON /LAMIN4/ F2ATIP( 42,3),F2DTIP( 42,3)
      COMMON /LAMIN5/ RTIP(7)
      COMMON /LAMIN9/ I3TIP( 42,3)
      COMMON /WSSCH1/ DELTIE(  18,2,90),DTIEAV( 18,2)
      COMMON /REC1/   PVERT(90),PRAD(90)
      COMMON /REC2/   E(90)
      COMMON /REC3/   P(90)
      COMMON /WSSCH/  T1SSC1( 18,2),T2SSC1( 18,2),
     >                T1SSC2( 18,2),T2SSC2( 18,2)
      COMMON /WSSCH0/ TBSSC1( 42,3),TWSSC1( 42,3),
     >                TBSSC2( 42,3),TWSSC2( 42,3)
      COMMON /SHROUD/ TLINER( 18,2)
      COMMON /GAAG1/  FCOPW1(3)
      COMMON /GAAG2/  FCOPW2( 18,2)
      COMMON /HEA6 /  NPIN( 42),JPIN( 42,3)
      COMMON /SC02C/  QJ( 19, 42)
      COMMON /SC13C/  GEO1( 42,3)
      COMMON /QPAR1/  QDEV
C
C
C
      III=NS-NSTR
      FCOPW2(III,1)=FCOPW1(3)
      DTIEAV(III,1)=0.
      I2TIP(NS,1)=I3TIP(NS,1)
      IF(                I2TIP(NS,1).EQ.1)GOTO 2999
C     ---------------------------------------------------------------
C
C     I3TIP#1: THE TURBULENT CALCULATION MUST BE PERFORMED
C
      TWIAV=0.
```

```
      CS=1.
      AMA1=MSCH1(NS,1)/AT
      ANGT=0.
      AMT=0.
      TT=0.
      AMB=0.
      TTB=0.
      SRAMIB=0.
      DDDDA=0.
      DDDDB=0.
      ATB=0.
      HPLUSB(NS,1)=0.
      HPLUSW(NS,1)=0.
      TI=TSCH1(NS,1)
      DEPA=DETOT
C
      DO 3 I=1,N
      AI=I
      ANGT=ANGT+ALFA
C*****FIRST STEP: EVALUATION OF THE TAU=0 LINE AS FOR WALL CHANNELS****
      CALL RECANG(I,AI,NS,K,1,IRH,ALFA,AMA1,TI,PB,D,W,RH,DETOT,PROV,DAI
     *,DBI,AAI,ABI,GG,SSSA,SSSB,AMTI,3,H1,H,PR1,PR2,SQDPG,1,TE,SUR,TWI,
     *AMAI,TAI,AMBI,TBI,III,TSCH1(NS,1),TSCH(NS,1),HPLUS1,HPLUS2,ANGT,0.
     *,0.,1.,&777,DEPA,CS)
C*****SECOND STEP: SUBCHANNELS DEFINED WITH RADII FROM ROD CENTER******
      AAI=A(I)-ABI
      DAI=4.*AAI/PA(I)
      TI1=TI
      TAI1=TAI
      TBI1=TBI
      TW1=TWI
      CS1=CS
      CALL RECANG(I,AI,NS,K,2,IRH,ALFA,AMA1,TI1,PB,D,W,RH,DETOT,PROV,DA
     *I,DBI,AAI,ABI,GG,SSSA,SSSB,AMTI,3,H1,H,PR1,PR2,SQDPG,1,TE,SUR,TW1,
     *AMAI,TAI1,AMBI,TBI1,III,TSCH1(NS,1),TSCH(NS,1),HPLUS1,HPLUS2,ANGT,
     *0.,0.,1.,&777,DEPA,CS1)
      TWIAV=TWIAV+TW1*ALFA
      DTIEAV(III,1)=DTIEAV(III,1)+DELTIE(III,1,I)*AMTI
      AMT=AMT+AMTI
      TT=TT+AMTI*TI1
      AMB=AMB+AMBI
      RAMIB=AMTI*ABI/(AAI+ABI)
      SRAMIB=SRAMIB+RAMIB
      TTB=TTB+RAMIB*TBI1
      DDDDA=DDDDA+SSSA
      DDDDB=DDDDB+SSSB
      DDDD=DDDDA+DDDDB
      ATB=ATB+ABI
      IF(IRH.EQ.1)GOTO 3
      HPLUSB(NS,1)=HPLUSB(NS,1)+HPLUS1*ABI
      HPLUSW(NS,1)=HPLUSW(NS,1)+HPLUS2*ABI
    3 CONTINUE
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 21.09.1979
C        TWIAV=TWIAV*12./PIG
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
         TWIAV=TWIAV*6./PIG
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DTIEAV(III,1)=DTIEAV(III,1)/AMT
       ATSCH=TT/AMT
      RHOT=RHO(PB,ATSCH)
      LAMSCH(NS,1)=((AT/DDDD)**2          )*2.*DET*RHOT/H
      ADAB(III,1)=AT/ATB
      DETB(III,1)=48.*ATB/(PIG*D)
      AMA=AMT-AMB
```

```
      TSCHB=TTB/SRAMIB
      RHOBT=RHO(PB,TSCHB)
      LAMB(III,1)=((ATB/DDDDB)**2        )*2.*DETB(III,1)*RHOBT/H
      I2TIP(NS,1)=0
      F1ATIP(NS)=1.
      F1DTIP(NS)=1.
      F2ATIP(NS,1)=1.
      F2DTIP(NS,1)=1.
      IF(I3TIP(NS,1).EQ.2)GOTO 3000
C     -----------------------------------------------------------------
C
C     I3TIP=3: THE LAMINAR CALCULATION MUST BE ALSO PERFORMED
C
      IF(ITCORR.GT.1)GOTO 2999
      MSCH(NS,1)=AMT*ASCH(NS,1)/AT
      TSCH(NS,1)=ATSCH
      TW(NS,1)=TWIAV
C     -----------------------------------------------------------------
C
C     FOR I3TIP=1 OR I3TIP=3
C
 2999 CONTINUE
      R1DR2L=1./SQRT(1.+12.*AT*FATIP(3)/(PIG*RTIP(IPA)**2))
      RELA=RELAM(ASCH(NS,1)*FATIP(3),DET*FDTIP(3),PB,TSCH(NS,1),TW(NS,1)
     &             ,MSCH(NS,1),TLINER(III,1),3,R1DR2L,1.)
      LAMLAM=AKAPPA(NS)/RELA
      R2COR=RTIP(IPA)/R1DR2L
      CALL ENTRFR(K,1,3,RTIP(IPA),R2COR,R2COR,NS,III,1,DET*FDTIP(3),
     *             ASCH(NS,1)*FATIP(3),MSCH(NS,1),PB,TSCH(NS,1),LAMLAM)
      IF(                    I2TIP(NS,1).EQ.1)GOTO 2997
C     -----------------------------------------------------------------
C
C     I3TIP=3: SAGAPO DECIDES WHETHER THE FLOW IS LAMINAR OR TURBULENT
C
      IF(LAMSCH(NS,1).GT.LAMLAM)GOTO 3000
C     -----------------------------------------------------------------
C
C     THE FLOW IS LAMINAR
C
 2997 CONTINUE
      LAMSCH(NS,1)=LAMLAM
      DDDD=AT*FATIP(3)/SQRT(LAMLAM*H/(2.*DET*FDTIP(3)*
     *RHO(PB,TSCH(NS,1))))
      AMT=MSCH(NS,1)*AT/ASCH(NS,1)
      ATSCH=TSCH(NS,1)
      I2TIP(NS,1)=1
      F1ATIP(NS)=FATIP(3)
      F1DTIP(NS)=FDTIP(3)
      F2ATIP(NS,1)=FATIP(3)
      F2DTIP(NS,1)=FDTIP(3)
      HPLUSB(NS,1)=1.
      HPLUSW(NS,1)=1.
      QPLUS(NS,1)=1.
      PRB(NS,1)  =1.
      YODH(NS,1)=1.
      TBSSC1( NS,1  )=TSCH(NS,1  )
      T1SSC1(III,1  )=TSCH(NS,1  )
      T2SSC1(III,1  )=TSCH(NS,1  )
      TWSSC1( NS,1  )=TW(NS,1  )
      TBSSC2(NS ,1  )=TSCH(NS,1  )
      T1SSC2(III,1  )=TSCH(NS,1  )
      T2SSC2(III,1  )=TSCH(NS,1  )
      TWSSC2( NS,1  )=TW(NS,1  )
      ADAB(III,1)=2.
      AMA=AMT*0.5
       AMB=AMA
```

```
 3000 CONTINUE
C     -----------------------------------------------------------------
C
C     THE FLOW IS TURBULENT
C
      IF(IRH.EQ.1)RETURN
      IF(                  I2TIP(NS,1  ).EQ.1)RETURN
C
      HPLUSB(NS,1)=HPLUSB(NS,1)/ATB
      HPLUSW(NS,1)=HPLUSW(NS,1)/ATB
      CPTB=CP(PB,TSCHB)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  OLD CCCCCCCCCC
C     QPLUS(NS,1)=QQ(NS,1)*ATB/(SUR*AMB*CPTB*(TE+273.16))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                             07.03.1979
      QAD=0.0
      JP=JPIN(NS,1)
      IF (QDEV .GT. 1.E-06)  QAD=QJ(JP,NS)*GEO1(NS,1)/(QDEV*H1)
      QAD1=QQ(NS,1)+QAD
      QPLUS(NS,1)=QAD1*ATB/(SUR*AMB*CPTB*(TE+273.16))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      PRB(NS,1)  =ETA(PB,ATSCH)*CP(PB,ATSCH)/KAPPA(PB,ATSCH)
      YODH(NS,1)=0.5*(SQRT(D**2+D*DETB(III,1))-D)/RH
      RETURN
  777 RETURN 1
      END
```

```
      SUBROUTINE ASSE(NCA,XF)
C ---------------------------------------------------------------------
C     ASSIGNES A START VALUE TO THE FUEL TEMPERATURES.
C
      DIMENSION XF(13)
C
      NC1=NCA+1
      NC2=NCA*2+1
      DO 100 K=NC1,NC2
  100 XF(K)=250.
      RETURN
      END
```

```
      SUBROUTINE AXSEC(NDE1,NDE2,DETC,WSP,CONST,DDD,II,HH,MSPAC,LENGTH,N
     *,IPA,QTOT,NSTOT,XMAXNU,CHSLNU)
C-----------------------------------------------------------------------
C     AXSEC EVALUATES SECTION LENGTHS AND CORRECTION FACTORS FOR NU.
C
      REAL LENGTH,NDE1,NDE2
C
      COMMON /HEA6/   NPIN( 42),JPIN( 42,3)
      COMMON /GRID1/  EPSISC( 42,3,5),DIST(7)
      COMMON /GRID2/  YY(100, 42,3)
      COMMON /GRID3/  X(100)
      COMMON /PRSPA/  DISTO
      COMMON /IDISPA/ IDISP1
C
      DIMENSION       B( 42,3),AA( 42,3),SLOPE( 42,3),YYM(3, 42,3)
C
C
C
      X1=NDE1*DETC
      IF(IDISP1.EQ.2)GOTO 1040
      DE11W=11.*DETC-WSP*0.5
      ISPA0=0
      IF(-DISTO.LT.DE11W*0.999)ISPA0=1
      MOSPAC=MSPAC+ISPA0
      IF(MOSPAC.GT.0 .AND. IPA.EQ.IPA/2*2 .AND. QTOT.GT.1.E-06)GOTO 2
C***********************************************************************
C*****UNHEATED SMOOTH PART OR PART WITHOUT SPACERS OR IDISP1=2*********
C***********************************************************************
 1040 SEC=LENGTH/X1+1.
      N=SEC
      SEC=N
      H=LENGTH/SEC
      DO 1 K=1,N
      DO 100 NS=1,NSTOT
      NP=NPIN(NS)
      DO 100 M=1,NP
  100 YY(K,NS,M)=1.
    1 X(K+1)=X(K)+H
      IF(MSPAC.EQ.0)RETURN
      N1=N
 1044 III=II
      K0=1
 1045 CONTINUE
      DO 1046 K=K0,N
      IF(X(K).GT.DIST(III) .AND. III.LT.MSPAC)III=III+1
      IF(III+1.GT.MSPAC)GOTO 1046
      KK=K
      IF(X(K).LT.DIST(III) .AND. X(K+1).GT.DIST(III+1))GOTO 1047
 1046 CONTINUE
      GOTO 1049
 1047 N=N+1
      NK1=N-KK
      DO 1048 K=1,NK1
       K1=N-K+2
 1048 X(K1)=X(K1-1)
      X(KK+1)=X(KK)+(DIST(III)+DIST(III+1))*0.5
      K0=KK+1
      GOTO 1045
 1049 CONTINUE
      IF(N1.EQ.N)RETURN
      K1=N1+1
```

```
      DO 1050 K=K1,N
      DO 1050 NS=1,NSTOT
      NP=NPIN(NS)
      DO 1050 M=1,NP
 1050 YY(K,NS,M)=1.
      RETURN
C*********************************************************************
C*****HEATED PART WITH SPACERS: AXIAL STEPS FIT CORR. PROF. FOR NU*****
C*********************************************************************
    2 CONTINUE
      ZETA1=2./(1.+XMAXNU)
      JSPAC=MSPAC+II-1
C
      AMM=8./NDE2+0.5
      MM=AMM
      BMM=MM
      NDE2=8./BMM
      X2=NDE2*DETC
C
C     (NOTE THAT 8/NDE2 MUST BE AN INTEGER TO FIT CORR. PROF. FOR NU)
      NPSEC=0
      K=0
      M1=NPSEC+1
      M2=M1+MM+3
      M3=M2+1
      JSPAC0=JSPAC+1
      XXX1=1.E07
       XXX11=1.E07
      KLK=0
C
C
      DO 16 ISPAC=II,JSPAC0
      DEL0=HH
      IF(ISPAC.LE.JSPAC)XXX1=DIST(ISPAC)-WSP*0.5-DETC
      IF(XXX1.LE.HH)DIST0=-1.E07
 1020 I1SPAC=ISPAC-II+1
      IF(-DIST0.LE.DE11W*0.999)I1SPAC=MSPAC+1
      IF(-DIST0.GT.DE11W*0.999 .AND. ISPAC.EQ.JSPAC+1)GOTO 1030
      DO 3 NS=1,NSTOT
      NP=NPIN(NS)
      DO 3 NN=1,NP
      B(NS,NN)=CONST*EPSISC(NS,NN,I1SPAC)**2
      YYM(1,NS,NN)=1.+0.75*ZETA1*B(NS,NN)
      YYM(2,NS,NN)=(1.+B(NS,NN)*0.5*(1.+ZETA1))*(XMAXNU-1.)+(2.-XMAXNU)*
     *(1.+0.5*B(NS,NN)*(1.+CHSLNU+(1.-CHSLNU)/(3.-XMAXNU)))
      YYM(3,NS,NN)=1.+0.5*B(NS,NN)*(2.*CHSLNU+(1.-CHSLNU)/(3.-XMAXNU))
      AA(NS,NN)=1.+CHSLNU*B(NS,NN)
    3 SLOPE(NS,NN)=CHSLNU*B(NS,NN)*0.125/DETC
      IF(ISPAC.EQ.JSPAC+1)GOTO 4
      IF(ISPAC.EQ.II .AND. HH.GE.XXX1)GOTO 11
    4 K=K+1
      L=K+1
      IF(K.NE.NPSEC+1)GOTO 10
      IF(-DIST0.GT.DE11W*0.999)GOTO 1010
C
C*****EFFECT OF THE LAST SPACER PRECEEDING THE POINT AT WHICH THE
C     CALCULATION HAS BEEN STARTED (ADDED AT GA)                *******
      X10=X(1)
      DEL0=DE11W+DIST0+X10
      DIST0=-1.E07
      XLL=DEL0-8.*DETC
      X00=X10
      DO 1000 KI=1,MM
      X00=X00+X2
      KI2=KI
       IF(X00.GE.DEL0*0.999)GOTO 1003
```

```
 1000 CONTINUE
      MM1=MM+1
      MM2=MM+3
      DO 1001 KI=MM1,MM2
      X00=X00+DETC
      KI2=KI
      IF(X00.GE.DELO*0.999)GOTO 1003
 1001 CONTINUE
      WRITE(6,1002)
 1002 FORMAT(1H1,5X,'ERROR IN AXSEC (DELO)')
      STOP
C
 1003 L=KI2+1
      J=4
      X(L)=DELO
      DO 1008 KI1=1,KI2
      K=KI2+1-KI1
      IF(KI1-MM)1004,1004,1006
 1004 X(K)=X(K+1)-X2
      DO 1005 NS=1,NSTOT
      NP=NPIN(NS)
      DO 1005 NN=1,NP
 1005 YY(K,NS,NN)=AA(NS,NN)-(X(K)+X2*0.5-XLL)*SLOPE(NS,NN)
      GOTO 1008
 1006 X(K)=X(K+1)-DETC
      J=J-1
      DO 1007 NS=1,NSTOT
      NP=NPIN(NS)
      DO 1007 NN=1,NP
 1007 YY(K,NS,NN)=YYM(J,NS,NN)
 1008 CONTINUE
      X(1)=X10
      DO 1013 KI1=1,KI2
      L=KI1+1
      K=L-1
      NPSEC=K
      IF(X(L).GT.XXX1)GOTO 1015
      IF(X(L).GT.DDD)GOTO 1014
 1013 CONTINUE
      GOTO 1020
C
C     THE END OF THE AXIAL PORTION HAS BEEN OVERTAKEN
 1014 X(L)=DDD
      N=KI1
      RETURN
C
C
C     THE BEGINNING OF THE INFLUENCE REGION OF THE SUCCEEDING SPACER HAS
C     BEEN OVERTAKEN
 1015 X(L)=XXX1
      DELO=XXX1
      GOTO 1020
C
C
C*****AXIAL STEPS WHERE NO EFFECT OF SPACERS ON NU IS PRESENT**********
 1010 CONTINUE
      DX=XXX1-DELO
      SEC=DX/X1+1.
      NSEC=SEC
      SEC=NSEC
      H=DX/SEC
      IF(ABS(DX).LE.1.E-05)NSEC=0
      M1=NSEC+NPSEC+1
      M2=M1+MM+3
      M3=M2+1
      KLK=0
```

```
      XXX11=1.E07
      IF(ISPAC.LT.JSPAC)XXX11=DIST(ISPAC+1)-WSP*0.5-DETC
      IF(NSEC.EQ.0)GOTO 10
    7 CONTINUE
      DO 8 NS=1,NSTOT
      NP=NPIN(NS)
      DO 8 NN=1,NP
    8 YY(K,NS,NN)=1.
      X(L)=X(K)+H
      GOTO 4
   10 IF(K-M1)7,11,13
   11 CONTINUE
C
C****AXIAL STEPS (DIST(ISPAC)-WSP/2-DETC)-(DIST(ISPAC)-WSP/2+3*DETC) **
      IF(ISPAC.EQ.II .AND. HH.GE.XXX1)K=1
      XXX2=X(K)-XXX1
C     XXX2#0 IF DETC > DISTANCE BETWEEN THE FIRST SPACER AND THE INLET
C     OF THE PART
      XXX3=DETC-XXX2
      X(K+1)=X(K)+XXX3
      K=K-1
      M1=M1-1
      M2=M2-1
      M3=M3-1
      IF(XXX3.LE.1.E-03)GOTO 101
      K=K+1
      M1=M1+1
      M2=M2+1
      M3=M3+1
      XXX3=0.
      DO 12 NS=1,NSTOT
      NP=NPIN(NS)
      DO 12 NN=1,NP
   12 YY(K,NS,NN)=1.+0.25*B(NS,NN)*(1.+XXX2/DETC)*ZETA1
  101 CONTINUE
      DO 60 J=1,3
      K=K+1
      L=K+1
      X(L)=X(K)+DETC+XXX3
      IF(X(L).GT.X(K))GOTO 77
      K=K-1
      XXX3=XXX3+DETC
      GOTO 60
   77 XXX3=0.
      DO 59 NS=1,NSTOT
      NP=NPIN(NS)
      DO 59 NN=1,NP
   59 YY(K,NS,NN)=YYM(J,NS,NN)
      IF(X(L).GT.DDD)GOTO 61
   60 CONTINUE
      LL=L
      GOTO 4
C
C*****PART ENDS BEFORE (DIST(ISPAC)-WSP/2+3*DETC ) IS REACHED *********
   61 CONTINUE
      X(L)=DDD
      N=K
      RETURN
C
   13 IF(K.EQ.M3)GOTO 15
C
C*****AXIAL STEPS WHERE INFLUENCE OF SPACERS IS DECREASING************
      X(L)=X(K)+X2
      DO 14 NS=1,NSTOT
      NP=NPIN(NS)
       DO 14 NN=1,NP
```

```
   14 YY(K,NS,NN)=AA(NS,NN)-(X(K)+X2*0.5-X(LL))*SLOPE(NS,NN)
      IF(X(L).GT.XXX11 .AND. KLK.EQ.0)KLK=K
      GOTO 4
   15 CONTINUE
C
C     END OF INFLUENCE OF THE SPACER.
      IF(KLK.NE.0)M2=KLK
      IF(KLK.NE.0) X(KLK+1)=XXX11
      K=M2
      NPSEC=M2
      HH=DIST(ISPAC)+DE11W
      IF(KLK.NE.0)HH=XXX11
      DELO=HH
   16 CONTINUE
C
C     ALL SPACERS HAVE BEEN CONSIDERED.
 1030 HH=DELO
      IF(HH.GT.DDD)GOTO 21
C*****END OF SMOOTH OR ROUGH PART NOT YET REACHED*********************
      DX=DDD-HH
      SEC=DX/X1+1.
      NSEC=SEC
      SEC=NSEC
      H=DX/SEC
      K1=K+1
      N=K+NSEC
      DO 20 K=K1,N
      L=K+1
      X(L)=X(K)+H
      DO 19 NS=1,NSTOT
      NP=NPIN(NS)
      DO 19 NN=1,NP
   19 YY(K,NS,NN)=1.
   20 CONTINUE
      RETURN
C
C*****END OF SMOOTH OR ROUGH PART OVERTAKEN: CORRECTION TO FIT END POINT
   21 CONTINUE
      DX=DDD-X(LL)
      SEC=DX/X2+1.
      NSEC=SEC
      SEC=NSEC
      H=DX/SEC
      N=LL+NSEC-1
      DO 25 K=LL,N
      L=K+1
      X(L)=X(K)+H
      DO 24 NS=1,NSTOT
      NP=NPIN(NS)
      DO 24 NN=1,NP
   24 YY(K,NS,NN)=AA(NS,NN)-(X(K)+H*0.5-X(LL))*SLOPE(NS,NN)
   25 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BALA(K,NSTOT,INDSP,ASEC,H,LENGTH,PR1,PR2,PBT,    FREL,FT
     *,ITCORR,ITCM,DPAV,ITERM,ITGL,*,WSP,I1SPAC)
C-----------------------------------------------------------------------
C     SUBROUTINE BALA EVALUATES OUTLET MASS FLOW RATES AND TEMPERATURES
C
      REAL LAM,MI,M2,MAV,LENGTH,MAVCF,MAV1,MAV2,KAPPA
C
      DIMENSION WCF1( 42),EP1( 42),A( 42),DE( 42),
     1          TA( 42), RHOAV( 42),RHO1( 42),XMEM( 42),I1TIP( 42)
C
      COMMON/GEO0/ACH(3)/HEA6/NPIN( 42),JJROD( 42,3)/GRID/CSPAC( 42,4)
     1      /CORR/SIGMA( 42),PHI( 42),SBMNS/LAMIN0/I2TIP( 42,3)
     2      /IJ1/NER( 42),NIS( 42,3)
     3      /GEN4/TEMP( 42)/GEN5/DEZ( 42)/LAMIN3/F1ATIP( 42),F1DTIP( 42)
     4      /IND3/NTYP( 42)/MOB1/M2( 42)/MOB2/UAV( 42)/MOB8/DP( 42)
     5      /MOB4/WCF( 42)/MOB5/TAV( 42)/MOB6/MAV( 42)/MOB24/WT( 42,3)
     6      /MOB26/RUAS( 42)
     7      /QPAR1/QDEV/QPAR2/QLINM,QLDEV/QPAR3/PERL(3)/GRID6/EPS( 42,4)
     8      /GRID7/PGDP( 42,4)/COND1/CCOND( 42,3)/MART2/NNSS1,NNSS2
     9      /GRAV/IGRAV/GAAG1/FCOPW1(3)
      COMMON/ENEOP/IENE/MIXS1/CY/MIXS2/CCY/SECIN/KK/GRID2/YY(100, 42,3)
      COMMON  /GEN1/LAM( 42)/GEN2/AZ( 42)/GEN3/MI( 42)
      COMMON  /TUR1/CTURB(  42,3)/HB3/TEMP2( 42)/HEA3/QT( 42)
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC07C/ H1
      COMMON /SC02L/ JLAM
      COMMON /MART5/ NSTR
      COMMON /SC06L/ SHQ( 18,2)
      COMMON /SC21C/ SHQC( 18,2)
      COMMON /SC09R/ QSR( 18,2)
C     ..................................................................
C     APPROXIMATE METHOD FOR THE LAMINAR CALCULATIONS
C
      KK=K
      CCY=CY
      IENFR=1
      DO 1001 NS=1,NSTOT
      NP=NPIN(NS)
      DO 1000 JJJ=1,NP
      IF(I2TIP(NS,JJJ).EQ.0 .OR. NTYP(NS).EQ.1)GOTO 1000
      IENFR=2
 1000 CONTINUE
 1001 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF (JLAM .EQ. 1  .AND. IENFR .EQ. 2 )          CALL ENFRC1
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(NNSS1.NE.0 .AND. NNSS2.NE.0 .AND. IENFR.EQ.2)CALL ENFRCO
C
      DO 400 NS=1,NSTOT
      RHO1(NS)=RHO(PR1,TEMP (NS))
      NP=NPIN(NS)
C
C     THE FLOW AREAS AND THE EQUIVALENT DIAMETERS ARE BASED ON THE TIP
C     DIAMETER OF THE RODS IN THE CASE OF LAMINAR CALCULATIONS
C     I1TIP(NS)=0 FOR TURBULENT FLOW; I1TIP(NS)=1 FOR LAMINAR FLOW
C
      I1TIP(NS)=0
      A(NS)=AZ(NS)*F1ATIP(NS)
      DE(NS)=DEZ(NS)*F1DTIP(NS)
      DO 399 JJJ=1,NP
  399 I1TIP(NS)=I1TIP(NS)+I2TIP(NS,JJJ)
```

```
C
      DO 400 M=1,3
      WT(NS,M)=0.
  400 CONTINUE
      XX=1./980665.
C     ...............................................................
C     ITERATION ON THE RELAXATION FACTOR (LOOP ITFREL)
C
      DO 999 ITFREL=1,98
      IVIA=1
C     ...............................................................
C     CALCULATION OF THE PRESSURE LOSSES (LOOP ITGL)
C
      DO 15 ITGL=1,70
C*****EVALUATION OF CROSS-FLOW SOLUTIONS*******************************
      CALL CRFL1(ITGL,DPAV,FREL,ASEC,NSTOT,A,MI,DP,WCF,WCF1,EP1)
      DO 1  NS=1,NSTOT
      M2(NS)=MI(NS)-H*WCF(NS)
      MAV(NS)=(M2(NS)+MI(NS))*0.5
      TA(NS)=TEMP(NS)
    1 CONTINUE
      IF(ITGL.GT.1  .AND. IVIA.EQ.1)GOTO 9
C     ...............................................................
C     CALCULATION OF THE AVERAGE GAS TEMPERATURES (LOOP ITERM)
C
      XPREC=1.E-04
      DO 7 ITERM=1,20
      DO 3 NS=1,NSTOT
      NP=NPIN(NS)
      YYNS=0.
      DO 1002 JJJ=1,NP
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 10.9.980
      CALL SPANU(1.,2.,3,4,YY(K,NS,JJJ))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
 1002 YYNS=YYNS+YY(K,NS,JJJ)
      YYNS=YYNS/FLOAT(NP)-1.
      THEX=0.
      CONHE=0.
      NI=NER(NS)
      NTYPNS=NTYP(NS)
      ACH1=ACH(NTYPNS)
      MAV1=MAV(NS)*ACH1/AZ(NS)
      DO 2 M=1,NI
      J=NIS(NS,M)
      NP=NPIN(J)
      YYJ=0.
      DO 1003 JJJ=1,NP
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 10.9.980
      CALL SPANU(1.,2.,3,4,YY(K,J,JJJ))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
 1003 YYJ=YYJ+ YY(K,J,JJJ)
      YYJ=YYJ/FLOAT(NP)-1.
      YYNSJ=(YYNS+YYJ)*CCY*0.5+1.
      NTYPJ=NTYP(J)
      ACH2=ACH(NTYPJ)
      MAV2=MAV(J)*ACH2/AZ(J)
      IF(TA(NS).LE.0. .OR. TA(NS).GT.3000. .OR. TA(J).LE.0. .OR. TA(J)
     *.GT.3000.)GOTO 302
      WT(NS,M)=TME(PBT,MAV1,MAV2,TA(NS),TA(J),LAM(NS),LAM(J),ACH1,ACH2,
     *CTURB(NS,M))*YYNSJ
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     IF(I1TIP(NS).NE.0 .OR. I1TIP(J).NE.0)WT(NS,M)=0.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 10.9.980
      IF(ABS(YY(K,J,JJJ)*YY(K,NS,JJJ)-1.0) .GT. 1.E-03) GO TO 401
      IF(I1TIP(NS).NE.0 .OR. I1TIP(J).NE.0)WT(NS,M)=0.
  401 CONTINUE
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      TANSJ=(TA(NS)*MAV1+TA(J)*MAV2)/(MAV1+MAV2)
      CONHE=CONHE-(TA(NS)-TA(J))*CCOND(NS,M)*(KAPPA(PBT,TA(NS))+KAPPA
     *(PBT,TA(J)))
    2 THEX=THEX-(TA(NS)-TA(J))*WT(NS,M)*CP(PBT,TANSJ)
      IF(ITGL.GT.1)GOTO 101
      CFHEX=0.
      GOTO 102
  101 CONTINUE
      DO 303 LS=1,NSTOT
      IF(M2(LS).LE.0.)GOTO 302
  303 CONTINUE
      CALL TMCF(NS,NI,TACF,MAVCF,MAV1)
      TANSCF=(TA(NS)*MAV1+TACF*MAVCF)/(MAV1+MAVCF)
      CFHEX=WCF(NS)*(TA(NS)-TACF)*CP(PBT,TANSCF)
  102 XXMAV=MAV(NS)
      XXM2=M2(NS)
      IF(IENE.EQ.2)XXMAV=MI(NS)
      IF(IENE.EQ.2)XXM2=XXMAV
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C 13.11.1979
      QTR=QT(NS)*QDEV*H1
      QTL=0.0
      NS1=NS-NSTR
      NIP=NPIN(NS)
      DO 450 L=1,NIP
       JP=JJROD(NS,L)
       QTR=QTR+QJ(JP,NS)
       IF (NS.GT.NSTR)  QTL=QTL+SHQ(NS1,L)+SHQC(NS1,L)
  450  CONTINUE
      Q=QTR+QTL
      TEMP2(NS)=TEMP(NS)+(Q+(THEX+CFHEX+CONHE)*H)/(XXMAV*CP(PBT,TA(NS)))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      PHI(NS)=(THEX+CFHEX+CONHE)*H/AZ(NS)
      TAV(NS)=(XXM2*TEMP2(NS)+MI(NS)*TEMP(NS))*0.5/XXMAV
    3 CONTINUE
      IF(ITGL.EQ.1)GOTO 9
      IF(ITERM.GT.10)XPREC=1.E-03
      IF(ITERM.GT.15)XPREC=1.E-02
      DO 4 NS=1,NSTOT
      IF(ABS(TAV (NS)/TA(NS)-1.).GT.XPREC)GOTO 5
    4 CONTINUE
      GOTO 9
    5 CONTINUE
      DO 6 NS=1,NSTOT
    6 TA(NS)=TAV(NS)
    7 CONTINUE
C     ..............................................................
C     END OF LOOP ITERM
C
       WRITE(6,8)ITGL,ITCORR,(TAV(NS),NS=1,NSTOT)
    8 FORMAT(    5X,'   SUB. BALA',
     >          /5X,'   CHANNEL CALCULATION STOPS IN LOOP ITERM AT ITGL=
     $',I6,/5X ,'ITCORR=',I5/5X,'TEMPERATURES='/(8E15.5))
      RETURN 1
C     ..............................................................
C     CALCULATION OF PRESSURE LOSSES FOR CHANNELS
C
    9 CONTINUE
      DO 10 NS=1,NSTOT
      RHOAV(NS)=RHO(PBT,TAV(NS))
   10 UAV(NS)=MAV(NS)/(A(NS)*RHOAV(NS))
      DPAV=0.
      SMA=0.
      DO 13 NS=1,NSTOT
       TMOEX=0.
```

```
      NI=NER(NS)
      NTYPNS=NTYP(NS)
      ACH1=ACH(NTYPNS)
      DO 11 M=1,NI
      J=NIS(NS,M)
      TMOEX=TMOEX-(UAV(NS)-UAV(J))*WT(NS,M)
   11 CONTINUE
      TMOEX=FT*TMOEX/A(NS)*H
      IF(ITGL.GT.1)GOTO 103
      CFMOEX=0.
      GOTO 104
  103 UCFAV=UA(NS,NI,ACH1,1)
      CFMOEX=(2.*UAV(NS)-UCFAV)*WCF(NS)/A(NS) *H
  104 CONTINUE
      XMEM(NS)=LAM(NS)*H/(2.*DE(NS)*RHOAV(NS))*FCOPW1(NTYPNS)
      RE=MAV(NS)*DE(NS)/(A(NS)*ETA(PBT,TAV(NS)))
      IF(INDSP.EQ.2)XMEM(NS)=XMEM(NS)+(CSPAC(NS,I1SPAC)+DSPDPF(EPS(NS,I1
     *SPAC), DE(NS),LAM(NS),WSP,PGDP(NS,I1SPAC),RE,NTYP(NS)))/RHOAV(NS)
   12 DP(NS)=XX*(-(MAV(NS)/A(NS))**2*(XMEM(NS)-(RHO(PR2,TEMP2(NS))-RHO1(
     *NS))/RHOAV(NS)**2)+TMOEX+CFMOEX+IGRAV*RHOAV(NS)*980.665*H)
      DPAV=DPAV+DP(NS)*MI(NS)
      SMA=SMA+MI(NS)
   13 CONTINUE
      DPAV=DPAV/SMA
C     ..................................................................
C     TEST OF CONVERGENCE FOR THE CHANNEL PRESSURE LOSSES
C
      DO 14 NS=1,NSTOT
      IF(ABS(DP(NS)/DPAV-1.).GT.1.E-02)GOTO 15
      IF(ABS(DP(NS)/DPAV-1.).GT.1.E-03 .AND. ITGL.LT.40)GOTO 15
   14 CONTINUE
      IF(IVIA.EQ.2)GOTO 17
      DO 301 NS=1,NSTOT
      IF(M2(NS).LE.0.)GOTO 302
  301 CONTINUE
      IVIA=2
   15 CONTINUE
C     ..................................................................
C     END OF LOOP ITGL
C
  302 CONTINUE
      AIT=ITFREL
      FREL=1.-AIT*0.01
  999 CONTINUE
C     ..................................................................
C     END OF LOOP ITFREL
C
      WRITE(6,16)ITCORR,(DP(NS),NS=1,NSTOT),(MAV(NS),NS=1,NSTOT),(TAV(NS
     *),NS=1,NSTOT)
   16 FORMAT(//  5X,'SUB. BALA',
     >        /  5X,'CHANNEL CALCULATION STOPS IN LOOP ITGL AT ITCORR=',
     *I5/5X,'PRESSURE LOSSES, AVERAGE MASSES, AVERAGE TEMPERATURES:'/
     *(8E15.5))
      RETURN 1
C     ..................................................................
C     CONTRIBUTIONS OF CROSS-FLOW,TURBULENT MIXING AND DENSITY
C     TO THE PRESSURE DROPS OF THE CHANNELS (SIGMA)
C
   17 CONTINUE
      SBMNS=0.
      DO 21 NS=1,NSTOT
      NTYPNS=NTYP(NS)
      RUAS(NS)=MAV(NS)*SQRT(LAM(NS)*0.125)/AZ(NS)*ACH(NTYPNS)
      DPAVF=DPAV-IGRAV*RHOAV(NS)*0.001*H
      BMNS=SQRT(ABS(DPAVF)/(XX*XMEM(NS)))*A(NS)
      SIGMA(NS)=(MAV(NS)-BMNS)/AZ(NS)
```

```
      SBMNS=SBMNS+BMNS
   21 CONTINUE
      RETURN
      END
```

```
      FUNCTION BETAF(P,W,ZWC)
C     ------------------------------------------------------------------
C     BETAF EVALUATES THE PARAMETER BETA FOR THE DETERMINATION OF THE
C     SEPARATION LINE DEFINING THE TWO PORTIONS OF THE WALL SUBCHANNELS
C     IN THE LAMINAR CALCULATIONS
C     THE FOLLOWING EQUATION IS EXACTLY VALID AT ZWC=0, 1.2<P/D,W/D<1.5)
C
      BETAF=( 3.77165-2.0795*P)+(-1.71985+1.2139*P)*W
      RETURN
      END
```

```
      SUBROUTINE BOTH (VDIAM,PIG,H,VDIA1,RINT)
C -------------------------------------------------------------------
C     ORGANIZES THE CALCULATION OF THE THERMAL CONDUCTION WITHIN THE PIN
C     POWER GENERATED WITHIN THE FUEL.
C
      DIMENSION A(13,13),B(13),X(13),XF(13)
C
      COMMON /SC01C/ NCAN( 19),LIPS( 19,10)
      COMMON /SC03C/ NRODS
C
C     1. -----------------------------------------------------------
C     PROVIDES THE HEAT TRANSFER COEFFICIENTS.
      CALL ALFAC(VDIA1,H)
      CALL KGAP
C
C     2. -----------------------------------------------------------
C     COMPUTES SOME GEOMETRICAL FACTORS.
      CALL FGEO(VDIAM,RSTAR,S,R,RINT)
C
C     3. -----------------------------------------------------------
C     LOOP ON THE RODS.
C
      DO 1000 J=1,NRODS
            NCA=NCAN(J)
            IF ( NCA - 1 ) 1000,990,100
C           3.0 ---------------------------------------------------
C           CONVERGENCE LOOP ON THE FUEL TEMPERATURE.
  100       CALL ASSE(NCA,XF)
            DO 900 LIN=1,10
C           3.1 ---------------------------------------------------
C           THE ARRAYS ARE SET TO ZERO AT THE BEGINN OF EACH ITERATION
            DO 300 I=1,13
                  B(I)=0.0
                  DO 200 L=1,13
                        A(I,L)=0.0
  200                   CONTINUE
  300             CONTINUE
C
            IF ( NCA .EQ. 3 ) GO TO  500
C           3.2 ---------------------------------------------------
C           COMPUTES THE COEFFICIENTS OF THE EQUATIONS FOR THE CLAD.
C           CASE OF WHOLE PIN.
C
            CALL CCLAD(NCA,A,B,RSTAR,R,H,J,S)
C           3.3 ---------------------------------------------------
C           COMPUTES THE COEFFICIENTS OF THE EQUATIONS FOR THE FUEL.
C           CASE OF WHOLE PIN.
C
            CALL CFUEL(NCA,A,B,RSTAR,H,J,XF)
            GO TO 800
C           3.4 ---------------------------------------------------
C           COMPUTES THE COEFFICIENTS OF THE EQUATIONS FOR THE CLAD.
C           CASE OF HALF PIN.
C
  500       CALL CCLAD3(A,B,RSTAR,R,H,J,S)
C           3.5 ---------------------------------------------------
C           COMPUTES THE COEFFICIENTS OF THE EQUATIONS FOR THE FUEL.
C           CASE OF HALF PIN.
C
            CALL CFUEL3(A,B,RSTAR,H,J,XF)
  800       CONTINUE
```

```
C           3.6 ---------------------------------------------------
C           SOLVES THE SYSTEM OF EQUATIONS.
C
            NC2 = NCA*2+1
C           CALL LINAL (A,NC2,13,A,B,1,13,X)
            CALL GAUSS1(A,B,X,NC2,13)
C           3.7 ---------------------------------------------------
C           CONVERGENCE TEST ON THE THERMAL CONDUCTIVITY OF THE FUEL.
            CALL TEST1(NCA,X,XF,KK)
            IF (KK.LE.0) GO TO 950
  900       CONTINUE
            WRITE(6,920) LIN,J
  920       FORMAT ( //////5X,'SUBR. LEIT',/5X,
     >              'LIN = ',I2,'  ==>  CONVERGENCE PROBLEMS ',5X,
     >              'FOR PIN N. ',I3,/5X,
     >              'IN THE LOOP FOR FUEL THERMAL CONDUCTIVITY',/5X,
     >              'CALCULATION STOPS')
            STOP
C
C           3.8 ---------------------------------------------------
C           ASSIGNES THE NEW TEMPERATURE VALUES (X) TO THE ARRAY TW.
C
  950       CALL TNEW (NCA,X,J,13)
C           3.9 ---------------------------------------------------
C           DETERMINES THE HEAT EXCHANGED BY CONDUCTION.
C
  990       CALL QCOC (NCA,J,X,13)
 1000 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE CCLAD (NCA,A,B,RSTAR,R,H,J,S)
C ----------------------------------------------------------------------
C     COMPUTES THE COEFFICIENTS FOR THE EQUATIONS OF THERMAL CONDUCTION
C     WITHIN THE CLAD IN CASE OF POWER GENERATED IN THE FUEL.
C
C
      DIMENSION A(13,13),B(13)
C
      COMMON /SC01C/ NCAN( 19),LIPS( 19,10)
      COMMON /SC05C/ JZUR( 19, 42)
      COMMON /SC11C/ FALFA( 42,3)
      COMMON /SC12C/ GEO( 42,3)
      COMMON /SC20C/ CGAP
      COMMON /SC08R/ QPR( 42,3)
      COMMON /SUB22/ TW( 42,3)
      COMMON /SC26C/ TBR( 42,3),TBS( 18,2)
C
      REAL KMET
C
      DO 1000 K=1,NCA
              KP=K-1
              IF(K .EQ. 1) KP=NCA
              KD=K+1
              IF(K .EQ.NCA) KD=1
              KF=K+NCA
C
              NS1 = LIPS(J,K)
              NSP = LIPS(J,KP)
              NSD = LIPS(J,KD)
              MZ1 = JZUR(J,NS1)
              MZP = JZUR(J,NSP)
              MZD = JZUR(J,NSD)
C
              TMP = (TW(NS1,MZ1)+TW(NSP,MZP))*0.5
              TMD = (TW(NS1,MZ1)+TW(NSD,MZD))*0.5
              FKP = KMET(TMP)*H*S/(R*(GEO(NS1,MZ1)+GEO(NSP,MZP)))
              FKD = KMET(TMD)*H*S/(R*(GEO(NS1,MZ1)+GEO(NSD,MZD)))
              FKF = CGAP*RSTAR*GEO(NS1,MZ1)*H*2.
              B(K) = - TBR(NS1,MZ1)*FALFA(NS1,MZ1) - QPR(NS1,MZ1)
C
              DO 500 L=1,12
                     A(K,L) = 0.0
                     IF (L .EQ. KP)  A(K,L) = FKP
                     IF (L .EQ. K )  A(K,L) =-FKF-FKD-FKP-FALFA(NS1,MZ1)
                     IF (L .EQ. KD)  A(K,L) = FKD
                     IF (L .EQ. KF)  A(K,L) = FKF
                     IF (L .EQ. KF)  A(K,L) = FKF
  500                CONTINUE
 1000         CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CCLAD3 (A,B,RSTAR,R,H,J,S)
C -----------------------------------------------------------------------
C     COMPUTES THE COEFFICIENTS FOR THE EQUATIONS OF THERMAL CONDUCTION
C     WITHIN THE FUEL IN CASE OF POWER GENERATED IN THE FUEL.
C     CASE OF HALF ROD.
C
C
      DIMENSION A(13,13),B(13)
C
      COMMON /SC01C/ NCAN( 19),LIPS( 19,10)
      COMMON /SC05C/ JZUR( 19, 42)
      COMMON /SC11C/ FALFA( 42,3)
      COMMON /SC12C/ GEO( 42,3)
      COMMON /SC08R/ QPR( 42,3)
      COMMON /SC20C/ CGAP
      COMMON /SUB22/ TW( 42,3)
      COMMON /SC26C/ TBR( 42,3),TBS( 18,2)
C
      REAL KMET
C
      NS1 = LIPS(J,1)
      NS2 = LIPS(J,2)
      NS3 = LIPS(J,3)
      MZ1 = JZUR(J,NS1)
      MZ2 = JZUR(J,NS2)
      MZ3 = JZUR(J,NS3)
C
      TM12 = ( TW(NS1,MZ1) + TW(NS2,MZ2)) *0.5
      TM23 = ( TW(NS2,MZ2) + TW(NS3,MZ3)) *0.5
C
C
C
      F12 = KMET(TM12) * H * S / (R*( GEO(NS1,MZ1)+GEO(NS2,MZ2)))
      F23 = KMET(TM23) * H * S / (R*( GEO(NS3,MZ3)+GEO(NS2,MZ2)))
C
      FGAP1 = CGAP * RSTAR * GEO(NS1,MZ1) * H * 2.
      FGAP2 = CGAP * RSTAR * GEO(NS2,MZ2) * H * 2.
      FGAP3 = CGAP * RSTAR * GEO(NS3,MZ3) * H * 2.
C
      A(1,1) = -FALFA(NS1,MZ1) - FGAP1 - F12
      A(1,2) = F12
      A(1,4) = FGAP1
C
      A(2,1) = F12
      A(2,2) = - FALFA(NS2,MZ2) - FGAP2 - F12 - F23
      A(2,3) = F23
      A(2,5) = FGAP2
C
      A(3,2) = F23
      A(3,3) = - FALFA(NS3,MZ3) - FGAP3 - F23
      A(3,6) = FGAP3
C
      B(1) = -TBR(NS1,MZ1) * FALFA(NS1,MZ1) - QPR(NS1,MZ1)
      B(2) = -TBR(NS2,MZ2) * FALFA(NS2,MZ2) - QPR(NS2,MZ2)
      B(3) = -TBR(NS3,MZ3) * FALFA(NS3,MZ3) - QPR(NS3,MZ3)
C
      RETURN
      END
```

```
      SUBROUTINE CEWA(K,NS,IRH,PROV,PB,RH,AA,DD,GG,AM1,DETOT,H1,ALFA,
     *I,JJJ,H,PR1,PR2,SQDPG,AMT,TT,DDDD,TE,SUR,ITYP,III,HPLUS1,HPLUS2,
     *TIE,SIGMA,PHI,*,D,TWI,TI,C)
C-----------------------------------------------------------------------
C     SUBROUTINE CEWA   EVALUATES FRICTION FACTORS AND APPROXIMATE
C     VALUES OF MASS FLOW RATES AND TEMPERATURE FOR 'CENTRAL-TYPE'  SUB-
C     SUBCHANNELS ( CENTRAL AND WALL CHANNELS ).
C
      REAL LAMI,KI,KAPPA,NUI
      COMMON/GRID2/YY(100, 42,3)/HEA5/QQ( 42,3)/DAT/PIG/MART/ITCORR
     1      /QPAR1/QDEV/COLAM1/COLAMB/SUB22/TW( 42,3)
     2      /GRAV/IGRAV/GAGR/DPSI
      COMMON /MART5/ NSTR
      COMMON /HEA6 / NPIN( 42),JPIN( 42,3)
      COMMON /IND3 / NTYP( 42)
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC13C/ GEO1( 42,3)
C
      IF(IRH.EQ.1)GOTO 1000
C
C     IN THE CASE OF SMOOTH RODS SINGLE VALUES OF THE SUB-SUBCHANNEL
C     PIN TEMPERATURES ARE NOT COMPUTED
      R1=D*0.5
      R0=0.5*SQRT(D**2+DD*D)
      FACHE=TIS(R1,R0,IRH)
      R1DR0=R1/R0
      YDH=(R0-R1)/RH
C
 1000 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC OLD CCCCCCCCCC
C     QROD=QQ(NS,JJJ)*QDEV
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      JP=JPIN(NS,JJJ)
      QAD=QJ(JP,NS)*GEO1(NS,JJJ)/H1
      QROD=QQ(NS,JJJ)*QDEV+QAD
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      Q=QROD*ALFA/(2.*PIG)*H1
      QA=QROD/SUR
      TI=TIE
C     ..................................................................
C     THE ITERATION PROCEDURE STARTS ASSUMING UNIFORM MASS-FLOW
C     DISTRIBUTION
C
      DO 10 ITW=1,10
C
      DO 4 IT=1,50
      DELTAT=(Q+PHI*AA)/(AM1*CP(PB,TI))
      TI=TIE+0.5*DELTAT
      IF(ITW.EQ.1 .AND. I.EQ.1) TWI=TI
      ETAI=ETA(PB,TI)
      RHOI=RHO(PB,TI)
      REI=AM1*DD/(AA*ETAI)
      ETAIW=ETA(PB,TWI)
      RHOIW=RHO(PB,TWI)
      REIW=(ETAI*RHOIW)/(ETAIW*RHOI)*REI
      IF(IT.EQ.1 .AND. ITW.EQ.1) GOTO 30
C     ..................................................................
C     AFTER 1.ST ITERATION FRICTION FACTORS ARE EVALUATED FROM THE
C     VALUES OBTAINED IN THE PRECEEDING ITERATION
C
      IF(REI.GT.0. .AND. SQ8LI.GT.0.)GOTO 700
```

```
 1001 WRITE(6,699)NS,JJJ,I,REI,SQ8LI
  699 FORMAT(//5X,'SUB. CEWA',
     >        /5X,'NS=',I5,5X,'M=',I2,5X,'I=',I3/5X,'RE=',E15.5,5X,'SQRT
     *(8/LAMBDA)=',E15.5)
      RETURN 1
  700 CONTINUE
      IF(IRH.EQ.2)GOTO 1
      SQ8LI=2.5*ALOG(REI/SQ8LI)+5.5-GG
      GOTO 3
    1 IF(SQ8LI.LE.0.)GOTO 1001
      HPLUSB=RH/DD*REI/SQ8LI
      HPLUSW=HPLUSB*REIW/REI
      GOTO 31
C     ...........................................................
C     1.ST ITERATION: FRICTION FACTORS ARE EVALUATED BY MEANS OF THE
C     EQUATION (LAMBDAI*RHOI*UI**2/DI) = (LAMBDA*RHO*U**2/D) TOT.
C
   30 IF(IRH.EQ.2)GOTO 2
      SQ8LI=2.5*ALOG(PROV/ETAI*SQRT((DD/DETOT)**3*RHOI))+5.5-GG
      GOTO 3
    2 HPLUSB=RH/DETOT*PROV/ETAI*SQRT(DD/DETOT*RHOI)
      HPLUSW=RH/DETOT*PROV/ETAIW*SQRT(DD/DETOT*RHOIW)
C
   31 CONTINUE
      QPLUS=QA*AA/(AM1*(TE+273.16)*CP(PB,TI))
      RHPL=RHPLUS(HPLUSB,TWI,TE,QPLUS,HPLUSW,TI,YDH)
      SQ8LI=2.5*ALOG(DD/RH)+RHPL-GG
    3 LAMI=8./SQ8LI**2*COLAMB
      SSS=AA/SQRT(LAMI*H/(2.*RHOI*DD))
      SQDPGI=SQRT(ABS(SQDPG**2*DPSI-IGRAV*RHOI*980.665*H))
      AM2=SSS*SQDPGI+SIGMA*AA
      IF(IT.EQ.1 .AND. ITW.EQ.1)GOTO 50
      IF(ABS(PLAMI/LAMI-1.).LE.1.E-04)GOTO 6
      PLAM1=PLAMI
      AM3=AM1
   50 PLAMI=LAMI
    4 AM1=AM2
C     ...........................................................
C     END OF LOOP IT
C
      WRITE(6,5)I,NS,K,ITW,ITCORR,AA,DD,ALFA,LAMI,PLAM1,AM3,AM2,TI,TIE,
     1TWI,PHI,SIGMA
    5 FORMAT(1H1,5X,'SUB. CEWA',
     >          /5X,'CALCULATION STOPS: IT=10 FOR SUBCH.',I3,2X,'(CHANNE
     *L NR.',I4,2X,'AXIAL SECTION NR.',I3,')',2X,'ITW=',I2,2X,'ITCORR=',
     *I4/5X,'AA=',E15.5/5X,'DD=',E15.5/5X,'ALFA=',E15.5/5X,'LAMI=',E15.5
     */5X,'PLAMI=',E15.5/5X,'AM1=',E15.5/5X,'AM2=',E15.5/5X,'TI=',E15.5/
     *5X,'TIE=',E15.5/5X,'TWI=',E15.5/5X,'PHI=',E15.5/5X,'SIGMA=',E15.7)
      RETURN 1
C
    6 IF(QQ(NS,JJJ).LE.1.E-06)GOTO 12
      IF(IRH.EQ.1)GOTO 13
C     ...........................................................
C     ITERATION TO FIND ROD TEMPERATURE FOR THE ROUGH PART
C
      KI=KAPPA(PB,TI)
      PRI=ETAI*CP(PB,TI)/KI
CCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      CALL SPANU(REI,PRI,NS,JJJ,YYI)
C     CALL RNU(HPLUSW,TWI,LAMI,REI,PRI,TI,YDH,R1DR0,0.,1.,REIW,YY(K,NS,
C    1 JJJ),NUI,GHPL)
C     CALL RNU(HPLUSW,TWI,LAMI,REI,PRI,TI,YDH,R1DR0,0.,1.,REIW,YYI,
C    1 NUI,GHPL,1)
C      FURTHER MODIFIED AT 09.10.1980
      BK=2.0
       IF(ITYP.EQ.1 .AND. NS.GT.NSTR) BK=1.0
```

```
      CALL RNU(HPLUSW,TWI,LAMI,REI,PRI,TI,YDH,R1DRO,0.,1.,REIW,YYI,
     1 NUI,GHPL,1,R0,R1,0.0,BK)
CCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      ALFAI=NUI*KI/DD*FACHE
      TIW=TI+QA/ALFAI
      IF(ABS(TIW/TWI-1.).LE.1.E-04)GOTO12
   10 TWI=TIW
C     .................................................................
C     END OF LOOP ITW
C
      WRITE(6,11)I,JJJ,NS
   11 FORMAT(     5X,'SUB. CEWA ',
     >          /5X,'CALCULATION STOPS: ITW=10 FOR SUBCH.',I3,2X, '(M=',
     *I2,2X,'NS=',I5,' )')
      RETURN 1
C
   13 TWI=TW(NS,JJJ)
C
   12 AMT=AMT+AM2
      TT=TT+TI*AM2
      DDDD=DDDD+SSS
      IF(IRH.EQ.1)RETURN
C
      HPLUS1=HPLUS1+HPLUSB*AA
      HPLUS2=HPLUS2+HPLUSW*AA
      RETURN
      END
```

```
      SUBROUTINE CEWACO(N,NN,NTYP,ALFA,D,X,AT,DET,MFLOW,ATOT,AREA,DE,ME)
C-----------------------------------------------------------------------
C     SUBROUTINE CEWACO EVALUATES GEOMETRICAL  PARAMETERS AND INLET MASS
C     FLOW RATES FOR 'CENTRAL-TYPE' AND CORNER SUB-SUBCHANNELS.
C
      REAL MFLOW,ME
      DIMENSION AREA(NN),DE(NN),ME(NN)
      COMMON/CEN1/G(46)/ANG1/RR2(30),ALF12(30)/ANG2/PER(30)
      PEROD=ALFA*D*0.5
      ARROD=PEROD*0.25*D
      E1=0.
      DO 3 I=1,NN
      AI=I
      E2=X*TAN(ALFA*AI)
      DELTAE=E2-E1
      AREA(I)=X*DELTAE*0.5-ARROD
      DE(I)=4.*AREA(I)/PEROD
      IF(NTYP.EQ.3)GOTO 1
      EPS=SQRT(1.+DE(I)/D)
      G(I)=GSTAR(EPS)
      GOTO 2
    1 PER(I)=DELTAE
      RR2(I)=SQRT(D**2+DE(I)*D)*0.5
       ALF12(I)=D*0.5/RR2(I)
      DE(I)=4.*AREA(I)/(PEROD+PER(I))
    2 CONTINUE
      ME(I) =MFLOW*AREA(I)/ATOT
    3 E1=E2
      IF(NTYP.EQ.3)GOTO 5
      WRITE(6,4)
    4 FORMAT(////130('*')/////
     *            5X,'GEOMETRY OF CENTRAL CHANNELS (REFERENCE TO 1/6)'//)
      GOTO 7
    5 WRITE(6,6)
    6 FORMAT(////130('*')////5X,'GEOMETRY OF ANGULAR CHANNELS (REFERENCE
     * TO 1/2)'//)
    7 CONTINUE
      WRITE(6,8)AT,DET
    8 FORMAT(5X,'TOTAL FLOW AREA=',F5.2,1X,'SQCM',5X,'TOTAL EQUIVALENT D
     *IAMETER=',F4.1,1X,'CM'//)
      WRITE(6,9)
    9 FORMAT(5X,'SECTION NR.',5X,'FLOW AREA (SQCM)',4X,'EQUIV. DIAMETER(
     *CM)'/)
      WRITE(6,10)(I,AREA(I),DE(I),I=1,N)
   10 FORMAT(7X,I3,15X,F7.5,17X,F5.3)
      RETURN
      END
```

```
      SUBROUTINE CFC1 (P,D,PIG,FC1)
C ---------------------------------------------------------------------
C     COMPUTES THE VIEW FACTOR OF TYPE FC1
C
      P3 = PIG/3.
      R  = D/2.
      PZD= P/D
C
      IF (PZD .GT. 1.0) GO TO 50
      FC1=0.5
      GO TO 999
C
   50 A = R/(P-R)
      ALFA = ACOS(A)
      DELT = P3 - ALFA
      IF (ALFA .GE. P3) GO TO 100
      AC = R*TAN(ALFA) + R*DELT
      GO TO 200
  100 AC=SQRT(R**2+(P-R)**2-R*(P-R))
  200 AB=P-R
      GR=(2.*SQRT(3.)+3.)/6.
      IF (PZD .GE. GR) GO TO 300
      AB=(R*TAN(ALFA)+R*(PIG/6.-ALFA))*2.
  300 DC=P-D
C
      FC1=(2*AC-AB-DC)
C
  999 RETURN
      END
```

```
      SUBROUTINE CFC2 (P,R,Z,ZWC,PIG,FC2)
C ---------------------------------------------------------------------
C     COMPUTES THE VIEW FACTOR FOR TWO ADJACENTS WALL SECTORS.
C
      CHL = ACOS(R/(P-R))
      PHM = ATAN(P*0.5/(Z-ZWC))
      PHL = ACOS(R/SQRT((Z-ZWC)**2+P**2/4.))
C
C --  CB  -------------------------------------------------------------
C
      CB = P-2.*R
C
C --  AC  -------------------------------------------------------------
C
      XY=(P*0.5-R)/TAN(CHL)
      IF (XY .LE. (Z-ZWC)) GO TO 200
      WRITE(6,100)
  100 FORMAT ( 5X,' SUB. CFC2 ',/5X,
     >' THE PRESENT COMBINATION OF P,D,Z,ZWC IS NOT PROVIDED IN THE PROG
     >RAM.',/5X,' YOU HAVE THE FOLLOWING POSSIBILITIES : ',/5X,
     >' 1. INSERT THIS CASE IN CFC2.',/5X,
     >' 2. COMPUTE FC2 BY HAND AND INTRODUCE IT IN CFC2.',//5X,
     >' CALCULATION STOPS.')
      STOP
C
  200 AC = R*TAN(CHL)+R*(PIG/2.-CHL)
C
C --  AD  ------------------------------------------------------------
C
      IF ((Z-R) .LT. ZWC) GO TO 300
      AD = P
      GO TO 500
C
  300 AD = 2.*(SQRT((Z-ZWC)**2+P**2/4.)*SIN(PHL)+R*(PHM-PHL))
  500 CONTINUE
C
C ------------------------------------------------------------------
      FC2 = (2.*AC-CB-AD)
C
      RETURN
      END
```

```
      SUBROUTINE CFC3 (P,D,PIG,FC3)
C ----------------------------------------------------------------------
C     COMPUTES THE VIEW FACTOR OF TYPE FC3
C
      P3 = PIG/3.
      PZD= P/D
      R  = D/2.
C
      IF (PZD .GT. 1.000) GO TO 50
      FC3=0.0
      GO TO 999
C
   50 IF (PZD .LT. 1.500) GO TO 100
C
      AC=(P-R)*SQRT(3.)
      AD=SQRT(D**2 + 3.*P**2 - 3.*D*P)
      AB=R
      FC3=(2.*AD-2.*AC)/(D*P3)
      GO TO 999
C
  100 A= R/(P-R)
      ALFA=ACOS(A)
      DELTA= P3 - ALFA
      AC=2*((P-R)*SIN(ALFA) + R*DELTA)
      AD=SQRT(D**2 + 3.*P**2 - 3.*D*P)
      FC3=(2*AD-2*AC)
C
  999 RETURN
      END
```

```
      SUBROUTINE CFC4 (P,D,PIG,FC4)
C ---------------------------------------------------------------------
C     COMPUTES THE VIEW FACTOR OF TYPE FC4
C
      P3 = PIG/3.
      R  = D/2.
      SQ3 = SQRT(3.)
      PZD=P/D
C
         A= R/(P-R)
         ALFA=ACOS(A)
         DELTA= P3 - ALFA
C -BMD-----------------------------------------------------------------
      BMD=(P-R) *SIN(ALFA) + R*(P3+DELTA)
C -ALD-----------------------------------------------------------------
         AXKS=R/(P/2.)
         XKS=ACOS(AXKS)
         FK=P-D/4.
         AK=SQRT(FK**2 + (R*SQ3*0.5)**2)
         CBETA= R/AK
         BETA=ACOS(CBETA)
         TGAM= R*SQ3*0.5/(P-D/4.)
         GAMMA=ATAN(TGAM)
         EPSI=2*P3-GAMMA-BETA
         FKL=GAMMA+BETA
         FKS=P3+XKS
      ALD= R*TAN(BETA)+R*EPSI
         IF (FKS .GE. FKL) GO TO 100
      ALD= R*TAN(ALFA)+R*(P3-ALFA-XKS)+2.*R*TAN(XKS)+R*(EPSI+(FKL-FKS))
  100    CONTINUE
C -BMC-----------------------------------------------------------------
      BMC=(P-R)*SIN(ALFA) + R*DELTA
         IF (ALFA .GE. P3) BMC=SQRT(R**2+(P-R)**2-(P-R)*R)
C -AC -----------------------------------------------------------------
      AC=P-R
         GR=(2.*SQRT(3.)+3.)/6.
         IF (PZD .LT. GR) AC=2*(R*TAN(ALFA)+R*(PIG/6.-ALFA))
C ---------------------------------------------------------------------
      FC4=(AC + BMD - BMC - ALD)
C
      IF (PZD .LE. 1.0) FC4=0.0
C
      RETURN
      END
```

```
      SUBROUTINE CFC5 (P,D,PIG,FC5)
C ----------------------------------------------------------------------
C     COMPUTES THE VIEW FACTOR OF TYPE FC5
C
      P3 = PIG/3.
      R  = D/2.
C
         A= R/(P-R)
         ALFA=ACOS(A)
         DELTA= P3 - ALFA
         B= D/P
         BETA=ACOS(B)
         EPSI=P3-BETA
C BMD -----------------------------------------------------------------
      BMD=R*TAN(ALFA)+R*DELTA
      IF ( DELTA .LE. 0) BMD=SQRT((P-R)**2+R**2-(P-R)*R)
C AC  -----------------------------------------------------------------
      AC=BMD
C BC  -----------------------------------------------------------------
      BC=P-D
C AFLD-----------------------------------------------------------------
      AFLD = 2.*(R*TAN(BETA)+R*EPSI)
      IF (BETA .GE. P3) AFLD=2.*SQRT(R**2+P**2/4.-R*P*0.5)
C ----------------------------------------------------------------------
C
      FC5=(AFLD+BC-2*BMD)
C
      IF (PZD .LE. 1.0) FC4=0.0
C
      RETURN
      END
```

```
      SUBROUTINE CFC9 (P,D,PIG,FC9)
C -----------------------------------------------------------------------
C     COMPUTES THE VIEW FACTOR OF TYPE FC9
C
      R=D/2.
      PZD=P/D
C
      BC1=2.*(P-R)*0.86603
      BC=BC1
      BKF=ACOS(R/(P-R))
      IF(PZD.LT.1.5) BC=D*(TAN(BKF)+(PIG/3.-BKF))
C
      AC=SQRT(BC1**2+R**2)
C
      IF(PZD.GE. 1.2) GO TO 100
      WRITE(6,50)
   50 FORMAT( ' SUB. CFC9:  P/D< 1.2 ; CASE NOT PROVIDED.'/5X,
     >         ' FC9 IS  SET TO 0.0. CALCULATION PROCEEDING')
      FC9=0.0
      GO TO 9999
  100 CONTINUE
      HX=(2.*P-R)*0.86603
      AH=SQRT(HX**2+R**2/4.)
      AHL=ACOS(R/AH)
      AHX=ASIN(R/2./AH)
      DHL=PIG/2.+AHX-AHL
      AD=R*(DHL+TAN(AHL))
C
      EKM=ACOS(R/(P/2.))
      EKF=PIG*2./3.-EKM-BKF
      BD=R*(2.*TAN(EKM)+(PIG/3.-EKM)+TAN(BKF)+EKF)
C
      FC9=AD+BC-AC-BD
C
 9999 RETURN
      END
```

```
      SUBROUTINE CFUEL (NCA,A,B,RSTAR,H,J,XF)
C ---------------------------------------------------------------------
C     COMPUTES THE COEFFICIENTS FOR THE EQUATIONS OF THERMAL CONDUCTION
C     WITHIN THE FUEL IN CASE OF POWER GENERATED IN THE FUEL.
C
C
      DIMENSION A(13,13),B(13),XF(13)
C
      COMMON /SC01C/ NCAN( 19),LIPS( 19,10)
      COMMON /SC05C/ JZUR( 19, 42)
      COMMON /SC07C/ H1
      COMMON /SC12C/ GEO( 42,3)
      COMMON /SC20C/ CGAP
      COMMON /HEA5 / QQ( 42,3)
      COMMON /QPAR1/ QDEV
C
      REAL KFUEL
C
      PIG=3.141593
      NCA1 = NCA+1
      NCA2 = NCA*2
      NCA3 = NCA*2+1
C
      A(NCA3,NCA3)=0.0
C
      DO 1000 K=NCA1,NCA2
C
              KP=K-1
              IF(K .EQ. NCA1)   KP=NCA2
              KD=K+1
              IF(K .EQ. NCA2)   KD=NCA1
              KC=K-NCA
              K1=KP-NCA
              K2=KD-NCA
C
              NS1 = LIPS(J,KC)
              NSP = LIPS(J,K1)
              NSD = LIPS(J,K2)
              MZ1 = JZUR(J,NS1)
              MZP = JZUR(J,NSP)
              MZD = JZUR(J,NSD)
C
              TFP = (XF(K)+XF(KP))*0.5
              TFD = (XF(K)+XF(KD))*0.5
              TFX = (XF(K)+XF(NCA3))*0.5
C
              FKP = KFUEL(TFP)*H*0.5/(GEO(NS1,MZ1)+GEO(NSP,MZP))
              FKD = KFUEL(TFD)*H*0.5/(GEO(NS1,MZ1)+GEO(NSD,MZD))
              FKC = CGAP*RSTAR*GEO(NS1,MZ1)*H*2.
              FKF = KFUEL(TFX)*H*GEO(NS1,MZ1)
              B(K) = - QQ(NS1,MZ1)*QDEV*H1*(GEO(NS1,MZ1)/PIG)*0.75
C
              DO 500 L=1,12
                     A(K,L) = 0.0
                     IF (L .EQ. KC)    A(K,L) = FKC
                     IF (L .EQ. KP)    A(K,L) = FKP
                     IF (L .EQ. K )    A(K,L) =-FKD-FKP-FKC-FKF
                     IF (L .EQ. KD)    A(K,L) = FKD
  500                CONTINUE
              A(K,NCA3)= FKF
               A(NCA3,K)=-A(K,NCA3)
```

```
                 A(NCA3,NCA3)=A(NCA3,NCA3)+A(K,NCA3)
                 B(NCA3)=QQ(NS1,MZ1)*QDEV*H1*0.25
 1000            CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE CFUEL3 (A,B,RSTAR,H,J,XF)
C --------------------------------------------------------------------
C     COMPUTES THE COEFFICIENTS FOR THE EQUATIONS OF THERMAL CONDUCTION
C     WITHIN THE FUEL IN CASE OF POWER GENERATED IN THE FUEL.
C     HALF OF THE PIN.
C
C
      DIMENSION A(13,13),B(13),XF(13)
C
      COMMON /SC01C/ NCAN( 19),LIPS( 19,10)
      COMMON /SC05C/ JZUR( 19, 42)
      COMMON /SC07C/ H1
      COMMON /SC12C/ GEO( 42,3)
      COMMON /SC20C/ CGAP
      COMMON /HEA5 / QQ( 42,3)
      COMMON /QPAR1/ QDEV
C
      REAL KFUEL
C
      PIG=3.141593
C
      NS1 = LIPS(J,1)
      NS2 = LIPS(J,2)
      NS3 = LIPS(J,3)
      MZ1 = JZUR(J,NS1)
      MZ2 = JZUR(J,NS2)
      MZ3 = JZUR(J,NS3)
C
      TF12 = (XF(4)+XF(5))*0.5
      TF23 = (XF(6)+XF(5))*0.5
      TFC4 = (XF(4)+XF(7))*0.5
      TFC5 = (XF(4)+XF(7))*0.5
      TFC6 = (XF(4)+XF(7))*0.5
C
      F12 = KFUEL(TF12)*H*0.5/(GEO(NS1,MZ1)+GEO(NS2,MZ2))
      F23 = KFUEL(TF23)*H*0.5/(GEO(NS2,MZ2)+GEO(NS3,MZ3))
C
      FGAP1 = CGAP * RSTAR * GEO(NS1,MZ1) * H * 2.
      FGAP2 = CGAP * RSTAR * GEO(NS2,MZ2) * H * 2.
      FGAP3 = CGAP * RSTAR * GEO(NS3,MZ3) * H * 2.
C
      A(4,1) = FGAP1
      A(4,4) = - FGAP1 - F12 - KFUEL(TFC4)*H*GEO(NS1,MZ1)
      A(4,5) = F12
      A(4,7) = KFUEL(TFC4)*H*GEO(NS1,MZ1)
C
      A(5,2) = FGAP2
      A(5,4) = F12
      A(5,5) = - FGAP2 - F12 - F23 - KFUEL(TFC5)*H*GEO(NS2,MZ2)
      A(5,6) = F23
      A(5,7) = KFUEL(TFC5)*H*GEO(NS2,MZ2)
C
      A(6,3) = FGAP3
      A(6,5) = F23
      A(6,6) = - FGAP3 - F23 - KFUEL(TFC6)*H*GEO(NS3,MZ3)
      A(6,7) = KFUEL(TFC6)*H*GEO(NS3,MZ3)
C
      A(7,4) = KFUEL(TFC4)*H*GEO(NS1,MZ1)
      A(7,5) = KFUEL(TFC5)*H*GEO(NS2,MZ2)
      A(7,6) = KFUEL(TFC6)*H*GEO(NS3,MZ3)
      A(7,7) = -A(7,4)-A(7,5)-A(7,6)
```

```
C
      B(4) = -QQ(NS1,MZ1) * QDEV * H1 * (GEO(NS1,MZ1)/PIG) * 0.75
      B(5) = -QQ(NS2,MZ2) * QDEV * H1 * (GEO(NS2,MZ2)/PIG) * 0.75
      B(6) = -QQ(NS3,MZ3) * QDEV * H1 * (GEO(NS3,MZ3)/PIG) * 0.75
      B(7) = -QQ(NS1,MZ1) * QDEV * H1 * 0.25
C
      RETURN
      END
```

```
      SUBROUTINE CF1(X1,X2,Y1,Y2,DP1,DP2,ITVIA,XYT,YT)
C-----------------------------------------------------------------------
C     CF1 IS USED IN THE CALCULATION OF THE AVERAGE CROSS-FLOW TEMPERA=
C     TURES AND VELOCITIES
C
      COMMON/GAMAR/CXX
C
C     XYT=(X1*Y1+X2*Y2)*CXX+XYT
C     YT=(Y1+Y2)*CXX+YT
C                         13.03.1980
      XYT=X2*Y2*CXX+XYT
      YT=Y2*CXX+YT
      RETURN
      END
```

```
      SUBROUTINE CF11(P,D,Z,ZWC,PIG,F11)
C ----------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 11
C
      AC = Z-D*0.5
C     ------------------------------------------------------------------
      HD=SQRT(P**2/4.+(Z-ZWC)**2)
      EHD=ACOS(D/2./HD)
      DHF=ACOS((P/2.)/HD)
C
      AD = SQRT((Z-D/2.-ZWC)**2+(P/2.)**2)
      IF (ZWC .LT. (Z-D/2.)) AD = D/2.*(PIG/2.-EHD-DHF)+D/2.*TAN(EHD)
C     ------------------------------------------------------------------
      BD = SQRT((Z-ZWC)**2 + ((P-D)/2.)**2)
C     ------------------------------------------------------------------
      CHG=ACOS(D/2./Z)
      BC = D/2.*TAN(CHG)+D/2.*(PIG/2.-CHG)
C     ------------------------------------------------------------------
      F11=(BC+AD-BD-AC)
C
      RETURN
      END
```

```
      SUBROUTINE CF12(P,D,Z,ZWC,PIG,F12)
C ---------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 12
C
      AD = SQRT(((P-D)/2.)**2+(Z-ZWC)**2)
C     -----------------------------------------------------------------
      HD=SQRT(P**2/4.+(Z-ZWC)**2)
      EHD=ACOS(D/2./HD)
      DHF=ASIN((P/2.)/HD)
C
      BD = SQRT((Z-D/2.-ZWC)**2+(P/2.)**2)
      IF (ZWC .GT. (Z-D/2.)) BD = D/2.*(PIG/2.-EHD-DHF)+D/2.*TAN(EHD)
C     -----------------------------------------------------------------
      BC = SQRT((Z-D/2.)**2+P**2)
      IF(((Z-D/2.)/2.) .LT. ZWC) BC=SQRT((P/2.)**2+ZWC**2)+BD
C     -----------------------------------------------------------------
      ACL=ATAN((P-D/2.)/Z)
      GCL=ASIN(D/2./Z)
      ALM=ACOS(D/2./(P-D/2.))
      CLG=PIG/2.-GCL
C
      AC = SQRT(Z**2+(P-D/2.)**2)
      IF (GCL .GT. ACL)
     >AC = D/2.*TAN(CLG)+D/2.*TAN(ALM)+D/2.*(PIG/2.-CLG-ALM)
C     -----------------------------------------------------------------
      F12=(BC+AD-BD-AC)
C
      RETURN
      END
```

```
      SUBROUTINE CF13(P,D,Z,ZWC,PIG,F13)
C ---------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 13
C
      BHE=ACOS(D/2./(P-D/2.))
      HCB=ATAN((P-D/2.)/Z)
      HCF=ASIN(D/2./Z)
C
      BC = SQRT(Z**2+(P-D/2.)**2)
      IF (HCF .GT. HCB)
     >BC = D/2.*(TAN(BHE)+TAN(PIG/2.-HCF)+(HCF-BHE))
C     -----------------------------------------------------------------
      HD=SQRT(P**2/4.+(Z-ZWC)**2)
      DHL=ACOS(D/2./HD)
      DHM=ACOS((P/2.)/HD)
C
      GD = SQRT((Z-D/2.-ZWC)**2+(P/2.)**2)
      IF (ZWC .LT. (Z-D/2.)) GD = D/2.*(PIG/2.-DHL-DHM)+D/2.*TAN(DHL)
C
      AC = SQRT((Z-D/2.)**2+P**2)
      IF (((Z-D/2.)*0.5) .LT. ZWC)
     >AC = SQRT((P/2.)**2+ZWC**2)+GD
C     -----------------------------------------------------------------
      AD = 3.*GD
      IF(ZWC .LT. (Z-D/2.)) AD=SQRT((Z-D/2.-ZWC)**2+(P*1.5)**2)
C     -----------------------------------------------------------------
      BD = D/2.*(TAN(BHE)+(PIG/2.-BHE))+GD
      IF(ZWC .LT. (Z-D/2.))
     >BD = D/2.*TAN(BHE)+HD*SIN(DHL)+D/2.*(PIG-BHE-DHL-DHM)
C     -----------------------------------------------------------------
      F13=(BC+AD-BD-AC)
C
C
      RETURN
      END
```

```
      SUBROUTINE CF14(P,D,Z,ZWC,PIG,F14)
C ---------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 14
C
      AC = Z-D/2.
C     -----------------------------------------------------------------
      EHD=ACOS(D/2./Z)
C
      AD = SQRT((D/2.)**2+Z**2-D/2.*Z)
      IF (Z .LT. D) AD = D/2.*TAN(EHD)+D/2.*(PIG/3.-EHD)
C     -----------------------------------------------------------------
      F14=2.*(AD-AC)
C
C
      RETURN
      END
```

```
      SUBROUTINE CF15(P,D,Z,ZWC,PIG,F15)
C ----------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 15
C
C     -----------------------------------------------------------------
      CEG=ACOS(D/2./Z)
      AEL=ACOS(D/2./(P-D/2.))
      AC = D/2.*(TAN(CEG)+TAN(AEL)+(PIG*5./6.-AEL-CEG))
C     -----------------------------------------------------------------
      XB=D/4.
      XY=P*SQRT(3.)/2.-D/2.*SQRT(3.)/2.
      ECB=ATAN((P/2.+XB)/(Z+XY))
      ECG=PIG/2.-CEG
      BEY=ATAN(XY/(P/2.+XB))
      BEF=ACOS(D/2./SQRT(XY**2+(P/2+XB)**2))
      BC = SQRT((P/2.+XB)**2+(Z+XY)**2)
      IF (ECG .GT. ECB)
     >BC = D/2.*(TAN(CEG)+TAN(BEF)+(PIG/2.-CEG-BEF+BEY))
C     -----------------------------------------------------------------
      F15= (BC-AC)
C
C
      RETURN
      END
```

```
      SUBROUTINE CF16(P,D,Z,ZWC,PIG,F16)
C ----------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 16
C
      BC = Z-D/2.
C     -----------------------------------------------------------------
      HD=SQRT(P**2/4.+(Z-ZWC)**2)
      EHD=ACOS(D/2./HD)
      DHF=ACOS((P/2.)/HD)
C
      BD = SQRT((Z-D/2.-ZWC)**2+(P/2.)**2)
      IF (ZWC .LT. (Z-D/2.)) BD = D/2.*(PIG/2.-EHD-DHF)+D/2.*TAN(EHD)
C     -----------------------------------------------------------------
      AD = PIG*D/4.+BD
      IF( ZWC .LT. (Z-D/2.)) AD=D/2.*TAN(EHD)+D/2.*(PIG-EHD-DHF)
C     -----------------------------------------------------------------
      CHG=ACOS(D/2./Z)
      AC = D/2.*TAN(CHG)+D/2.*(PIG/2.-CHG)
C     -----------------------------------------------------------------
      F16=(BC+AD-BD-AC)
C
C
      RETURN
      END
```

```
      SUBROUTINE CF17(P,D,Z,ZWC,PIG,F17)
C -----------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 17
C
      BC = Z-D/2.
C     ------------------------------------------------------------------
      EHA=ACOS(D/2./Z)
C
      AC = SQRT((D/2.)**2+Z**2-D/2.*Z)
      IF (Z.LT. D) AC = D/2.*TAN(EHA)+D/2.*(PIG/3.-EHA)
C     ------------------------------------------------------------------
      CHG=ACOS(D/2./Z)
      BD = D/2.*TAN(CHG)+D/2.*(PIG/2.-CHG)
C     ------------------------------------------------------------------
      AD = PIG*D/4.+AC
      IF( Z .GT. D) AD=D/2.*TAN(EHA)+D/2.*(PIG*5./6.-EHA)
C     ------------------------------------------------------------------
      F17=(BC+AD-BD-AC)
C
C
      RETURN
      END
```

```
      SUBROUTINE CF18(P,D,Z,ZWC,PIG,F18)
C -----------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 18
C
C     -------------------------------------------------------------
      FHA=ACOS(D/2./Z)
      AG = SQRT((D/2.)**2+Z**2-D/2.*Z)
      IF (Z.LT. D) AG = D/2.*TAN(FHA)+D/2.*(PIG/3.-FHA)
      CHL=ACOS(D/2./(P-D/2.))
C
      AC = D/2.*(TAN(CHL)+(PIG/2.-CHL))+AG
C     -------------------------------------------------------------
      HE=SQRT(P**2/4.+(Z-ZWC)**2)
      EHM=ACOS(D/2./HE)
      EHC=ACOS(P/2./HE)
      EG=SQRT((Z-ZWC-D/2.)**2+(P/2.)**2)
      IF (ZWC.GT. (Z-D/2.)) EG=D/2.*(TAN(EHM)+(PIG/2.-EHM-EHC))
C
      BD = SQRT((Z-D/2.)**2+P**2)
      IF (((Z-D/2.)*0.5) .LT. ZWC) BD = SQRT((P/2.)**2+ZWC**2) + EG
C     -------------------------------------------------------------
      CBH=ATAN((P-D/2.)/Z)
      HBN=ASIN(D/2./Z)
C
      BC = SQRT(Z**2+(P-D/2.)**2)
      IF (CBH .LT. HBN) BC = D/2.*(COTAN(HBN)+TAN(CHL)+(HBN-CHL))
C     -------------------------------------------------------------
      AD = AG+2.*EG
      IF (Z.GT. D) WRITE(6,100)
  100 FORMAT( ' CASE NOT PREVIDED IN CF18. F18 ARE ERRATED.')
C     -------------------------------------------------------------
      F18=(BC+AD-BD-AC)
C
C
      RETURN
      END
```

```
      SUBROUTINE CF24(P,D,Z,ZWC,PIG,F24)
C ---------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 24
C
      BC = Z-D/2.
C     ----------------------------------------------------------------
      EHC=ACOS(D/2./Z)
C
      AC = SQRT((D/2.)**2+Z**2-D/2.*Z)
      IF (Z.LT. D) AC = D/2.*TAN(EHC)+D/2.*(PIG/3.-EHC)
C     ----------------------------------------------------------------
      HD=SQRT(P**2/4.+(Z-ZWC)**2)
      GHD=ACOS(D/2./HD)
      DHF=ACOS((P/2.)/HD)
C
      BD = SQRT((Z-D/2.-ZWC)**2+(P/2.)**2)
      IF (ZWC .LT. (Z-D/2.)) BD = D/2.*(PIG/2.-GHD-DHF)+D/2.*TAN(GHD)
C     ----------------------------------------------------------------
C     AD = PIG*D/6.+ BD
      AD=D/2.*TAN(GHD)+D/2.*(PIG*5./6.-GHD-DHF)
C     ----------------------------------------------------------------
      F24= BC+AD-BD-AC
C
C
      RETURN
      END
```

```
      SUBROUTINE CF31(P,D,Z,ZWC,PIG,F31)
C -----------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 31
C
      BD = SQRT(P**2/4.+ZWC**2)
C     ------------------------------------------------------------------
      AC = Z
C     ------------------------------------------------------------------
      HD=SQRT(P**2/4.+(Z-ZWC)**2)
      EHD=ACOS(D/2./HD)
      FHD=ASIN((P/2.)/HD)
C
      GD = SQRT((Z-D/2.-ZWC)**2+(P/2.)**2)
      IF (ZWC .GT. (Z-D/2.)) GD = D/2.*(FHD-EHD)+D/2.*TAN(EHD)
C
      AHL=ACOS(D/2./Z)
      AM=Z*0.866
      AG=SQRT(AM**2+(D/2.-Z/2.)**2)
      IF(Z/2. .LT. D/2.) AG=D/2.*(TAN(AHL)+(PIG/3.-AHL))
C
      AD=AG+GD
C
      PN=P/2.+Z*0.866
      BS=ZWC+P/2.*PN*(Z/2.-ZWC)
      IF(BS.LT.(Z-D/2.)) AD=SQRT(PN**2+(Z/2.-ZWC)**2)
C     ------------------------------------------------------------------
      F31=(AC+BD-AD)
C
C
      RETURN
      END
```

```
      SUBROUTINE CF32(P,D,Z,ZWC,PIG,F32)
C -----------------------------------------------------------------------
C     VIEW-FACTORS OF TYPE 32
C
C     ------------------------------------------------------------------
      HD=SQRT(P**2/4.+(Z-ZWC)**2)
      EHD=ACOS(D/2./HD)
      DHF=ACOS((P/2.)/HD)
C
      AD = P
      IF (ZWC .GT. (Z-D/2.)) AD= D *(PIG/2.-EHD-DHF)+D*TAN(EHD)
C     ------------------------------------------------------------------
      AC = SQRT(P**2/4.+ZWC**2)
C     ------------------------------------------------------------------
      F32=(2*AC-AD)
C
C
      RETURN
      END
```

```
      SUBROUTINE CONNIJ(NSTR,NSTOT,NROMA,NSEL)
C-----------------------------------------------------------------------
C     CONNIJ EVALUATES FOR EACH CHANNEL I THE NUMBER  NER(I) OF
C     INTERACTIONS WITH OTHER CHANNELS J AND WHICH  CHANNELS INTERACT
C     WITH I.
C
      COMMON/IND1/NROW( 42),NUMS( 42)/IND2/NOT(4,30)/IND3/NTYP( 42)
     1      /IND4/NUM3(4),NUM6(4),NUM12(4),NUM18(4),NUM24(4),NUM30(4),
     2      NUM36(4)/IJ1/NER( 42),NIS( 42,3)
      IF(NSEL.EQ.4)GOTO 99
      NAN=NROMA+2
      NBN=1
      NCN=-1
      DO 43 NS=1,NSTOT
      NRO=NROW(NS)
      NUM=NUMS(NS)
      NUMA3=NUM3(NRO)
      NUMA6=NUM6(NRO)
      NUMA12=NUM12(NRO)
      NUMA18=NUM18(NRO)
      NUMA24=NUM24(NRO)
      NUMA30=NUM30(NRO)
      NUMA36=NUM36(NRO)
      IF(NS.GT.NSTR)GOTO 29
      IF(NUM.GT.1)GOTO 5
      IF(NSEL-2)1,2,4
    1 NER(NS)=3
      NIS(NS,3)=NOT(NRO,NUMA36)
      GOTO 3
    2 NER(NS)=2
    3 NIS(NS,1)=NS+1
      GOTO 13
    4 IF(NRO.GT.1)GOTO 2
      NER(1)=1
      NIS(1,1)=3
      GOTO 43
    5 IF(NSEL-2)6,7,8
    6 NUMSP=NUMA36
      GOTO 9
    7 NUMSP=NUMA18
      GOTO 9
    8 NUMSP=NUMA3
    9 IF(NUM.EQ.NUMSP)GOTO 10
      NER(NS)=3
      NIS(NS,3)=NS+1
      GOTO 12
   10 IF(NSEL.EQ.1)GOTO 11
      NER(NS)=2
      GOTO 12
   11 NER(NS)=3
      NIS(NS,3)=NOT(NRO,1)
   12 NIS(NS,1)=NS-1
   13 IF(NUM.GT.NUMA6)GOTO 14
      NAM=NUM
      GOTO 19
   14 IF(NUM.GT.NUMA12)GOTO 15
      NAM=NUM-NUMA6
      GOTO 19
   15 IF(NUM.GT.NUMA18)GOTO 16
      NAM=NUM-NUMA12
      GOTO 19
```

```
16 IF(NUM.GT.NUMA24)GOTO 17
   NAM=NUM-NUMA18
   GOTO 19
17 IF(NUM.GT.NUMA30)GOTO 18
   NAM=NUM-NUMA24
   GOTO 19
18 NAM=NUM-NUMA30
19 IF(NAM.EQ.(NAM/2*2))GOTO 21
   I1=1
   IF(NRO.EQ.NROMA)GOTO 20
   I2=1
   I3=0
   GOTO 22
20 I2=2
   I3=1
   GOTO 22
21 I1=-1
   I2=1
   I3=0
22 NRO1=NRO+I1
   IF(NUM.GT.NUMA6)GOTO 23
   NUMA=(NUM+I1)/I2+I3
   GOTO 28
23 IF(NUM.GT.NUMA12)GOTO 24
   NUMA=(NUM+I1-NUMA6)/I2    +NUM6(NRO1)
   GOTO 28
24 IF(NUM.GT.NUMA18)GOTO 25
   NUMA=(NUM+I1-NUMA12)/I2    +NUM12(NRO1)
   GOTO 28
25 IF(NUM.GT.NUMA24)GOTO 26
   NUMA=(NUM+I1-NUMA18)/I2    +NUM18(NRO1)
   GOTO 28
26 IF(NUM.GT.NUMA30)GOTO 27
   NUMA=(NUM+I1-NUMA24)/I2    +NUM24(NRO1)
   GOTO 28
27 NUMA=(NUM+I1-NUMA30)/I2    +NUM30(NRO1)
28 NIS(NS,2)=NOT(NRO1,NUMA)
   GOTO 43
29 IF(NUM.GT.1)GOTO 32
   IF(NSEL.EQ.1)GOTO 30
   NER(NS)=1
   GOTO 31
30 NER(NS)=2
   NIS(NS,2)=NSTOT
31 NIS(NS,1)=NS+1
   GOTO 43
32 IF(NSEL-2)33,34,40
33 NUMSP=NUMA36
   GOTO 35
34 NUMSP=NUMA18
35 IF(NUM.EQ.NUMSP)GOTO 37
   NIS(NS,1)=NS+1
   NIS(NS,2)=NS-1
   IF(NUM.EQ.NAN)GOTO 36
   NER(NS)=3
   NUMA=(NUM-NBN)*2+NCN
   NIS(NS,3)=NOT(NRO-1,NUMA)
   GOTO 43
36 NER(NS)=2
   NAN=NAN+NRO
   NBN=NBN+NRO
   NCN=NCN+2*NROMA-1
   GOTO 43
37 IF(NSEL.EQ.1)GOTO 38
   NER(NS)=1
   GOTO 39
```

```
   38 NER(NS)=3
      NIS(NS,2)=NOT(NRO,1)
      NIS(NS,3)=NIS(NS,2)-1
   39 NIS(NS,1)=NS-1
      GOTO 43
   40 IF(NUM.EQ.NUMA3)GOTO 41
      NER(NS)=3
      NIS(NS,3)=NS+1
      GOTO 42
   41 NER(NS)=2
   42 NIS(NS,1)=NS-1
      NUMA=(NUM-1)*2-1
      NIS(NS,2)=NOT(NRO-1,NUMA)
   43 CONTINUE
   99 CONTINUE
      DO 100  NS=1,NSTOT
      NI=NER(NS)
      WRITE(6,200)NS,NTYP(NS),(NIS(NS,M),M=1,NI)
  200 FORMAT(5X,'NS=',I2,5X,'TYPE=',I1,5X,'CHANNELS CONNECTED:',3I5)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONSHR
C ---------------------------------------------------------------------
C     THERMAL CONDUCTION WITHIN THE SHROUD
C
      DIMENSION A( 34, 34),B( 34),X( 34)
      COMMON /SC22C/ NTOT
C
C     DETERMINES THE ARRAY OF COEFFICIENTS.
      CALL MATBUS(A,B)
C
C     SOLVES THE SYSTEM
      CALL GAUSS(A,B,X)
C
C     ASSIGNES THE COMPUTED VALUES TO THE SHROUD TEMPERATURES
      CALL TNEWS(X)
C
C     COMPUTES THE HEAT TRANSMITTED TO THE GAS
      CALL QDEFIS
C
      RETURN
      END
```

```
      SUBROUTINE CONTRO(FA,FA1,ITCORR,INDICE)
C ----------------------------------------------------------------------
C     PRINTS INFORMATIONS ON THE CONVERGENCE PROCESS.
C
      DELTA=ABS(FA/FA1-1.0)
      WRITE(6,1001) ITCORR,FA,FA1,DELTA,INDICE
C
      RETURN
 1001 FORMAT(5X,'CONTRO, ITCORR =',I3,' LAM = ',E12.6,' LAM1 = ',E12.6,
     >       ' DELTA=',E12.6,'  INDICE=',I3)
      END
```

```
      SUBROUTINE CORKA
C     -----------------------------------------------------------------
C     CORKA MODIFIES THE COMPUTED LAMINAR FRICTION FACTOR IF WALL AND
C     CORNER CHANNELS ARE COMPUTED TOGETHER.
C     (CASE OF JLAM=1)
C
      COMMON /GEN2/   A( 42)
      COMMON /GEN5/   DE( 42)
      COMMON /INPAR/  IPA
      COMMON /LAMINK/ BKAPPA(7,3)
      COMMON /LAMIN1/ AKAPPA( 42)
      COMMON /LAMIN2/ FATIP(3),FDTIP(3)
      COMMON /IND3/   NTYP( 42)
      COMMON /IJ1  /  NER( 42),NIS( 42,3)
      COMMON /GASD1/  NSTOT
      COMMON /MART2/  NS1,NS2
      COMMON /SC03C/  NRODS
C
      IF (NRODS .NE. 7) GO TO 50
         NS1=7
         NS2= 18
         GO TO 9999
C
   50 DO 1000 NS=1,NSTOT
         ITYP=NTYP(NS)
         IF(ITYP .NE. 3) GO TO 1000
C
         PP = A(NS)*FATIP(ITYP)*(DE(NS)*FDTIP(ITYP))**2/BKAPPA(IPA,1)
         AT = A(NS)*FATIP(ITYP)
C
         NR = NER(NS)
         DO 100 M=1,NR
            NA=NIS(NS,M)
            IYP=NTYP(NA)
            PP=PP+A(NA)*FATIP(IYP)*(DE(NA)*FDTIP(IYP))**2/BKAPPA(IPA,2)
            AT=AT+A(NA)*FATIP(IYP)
  100       CONTINUE
C
         PP=AT/PP
         AKAPPA(NS)=(DE(NS)*FDTIP(ITYP))**2*PP
C
         DO 500 M=1,NR
            NA=NIS(NS,M)
            IP=NTYP(NA)
            AKAPPA(NA)=(DE(NA)*FDTIP(IP))**2*PP
  500       CONTINUE
 1000    CONTINUE
C
 9999 RETURN
      END
```

```
      SUBROUTINE CORRTE(TW,TB,PB,     NS,M,I,BIOT,TWINF)
C-----------------------------------------------------------------------
C     CORRTE CORRECTS THE COMPUTED TEMPERATURES FOR THE BIOT EFFECT AND
C     THE POSITION OF THE THERMOCOUPLE INSIDE THE CANNING
C
      COMMON /CEV03/  LAMOP2
      COMMON /CEV04/  LAMOP3
      COMMON /IROSMO/ IRH
      COMMON /BIDE/   IBIDE
      COMMON /CORRE/  QHRDAR,QRMDAR,QLAMR
      COMMON /LAMINO/ I2TIP( 42,3)
C
      REAL KMET,KINF,KAPPA
C
      TWINF=TW
      IF(IRH.EQ.1 ) GO TO 100
      IF(LAMOP3 .EQ. 1 .AND. LAMOP2 .EQ. 2 ) GO TO 100
C
C     ONLY FOR ROUGHENED RODS
      IF(I2TIP(NS,M).NE.1)GOTO 9
C     ..................................................................
C     FOR ROUGHENED RODS AND LAMINAR FLOW
C
      TW=TW+QLAMR/KAPPA(PB,TW)
      GOTO 100
C     ..................................................................
C     FOR ROUGHENED RODS AND TURBULENT FLOW
C
    9   TWBI=TWINF
      DTWINF=TW-TB
      DO 10 IT=1,10
      TWP=TW
      IF(IBIDE.EQ.1)TWBI=TW
      BIOT=QHRDAR/((TWBI -TB) *KMET(TWBI))
      TW=DTWINF/KINF(BIOT)+TB
      IF(ABS(TWP/TW-1.).LE.1.E-04)GOTO 13
   10 CONTINUE
      WRITE(6,12)NS,M,I,BIOT,TWP,TW
   12 FORMAT(1H1,5X,'CALCULATION STOPS IN SUBROUTINE CORRTE: NS=',I5,' M
     *=',I2,' I=',I3/5X,'BIOT=',E15.5,5X,'TWP=',E15.5,5X,'TW=',E15.5)
      STOP
C
   13 IF(QRMDAR.LE.1.E-06)TW=DTWINF/EINF(BIOT)+TB
C     ..................................................................
C     FOR SMOOTH AND ROUGHENED RODS, TURBULENT AND LAMINAR FLOW
C
  100 TW=TWCTEP(QRMDAR,TW)
      RETURN
      END
```

```
      FUNCTION CP(P,T)
C-----------------------------------------------------------------------
C     FUNCTION CP EVALUATES THE SPECIFIC HEAT OF THE COOLANT (CAL/G K)
C
      COMMON/GASD4/IGAS
      GOTO(10,20,30,40),IGAS
   10 CONTINUE
C     CASE OF HELIUM COOLANT
C
      CP=1.242
      RETURN
C
   20 CONTINUE
C     CASE OF CO2 COOLANT
C
      PP=P
      TT=T
      P=PP/1.0333
      T=TT+273.16
      TO=273.16
      TF=TO/T
      IF(P-1.) 1,1,2
    1 ECP = P -1.
      GO TO 3
    2 ECP = (P -1.)**1.05
    3 CPO=.118+3.51E-4*T-2.34E-7*T*T+6.00E-11*T*T*T
      CPF = CPO*(1.+1.089E-2*ECP*(TF**3.35))
      CP=CPF
      T=TT
      P=PP
      RETURN
C
   30 CONTINUE
C     CASE OF N2 COOLANT
C
      TT=T
      T=TT+273.16
      CPO=.2579-7.425E-5*T+1.604E-7*T*T-6.483E-11*T**3
      CP=CPO*(1.+1.886E-3*(P/1.033-1.)*(273.16/T)**2.4)
      T=TT
      RETURN
   40 CONTINUE
      CP=0.
      RETURN
      END
```

```
      SUBROUTINE CRFL1(ITGL,DPJAV,FREL,AJT,JMAX,AJ,MJ,DPJ,WCFJ,WCF1J,
     *EP1J)
C-----------------------------------------------------------------------
C     CRFL1 EVALUATES THE CROSS FLOW SOLUTIONS
C
      REAL MJ
      DIMENSION AJ(JMAX),MJ(JMAX),DPJ(JMAX),WCFJ(JMAX),WCF1J(JMAX),
     *EP1J(JMAX)
      IF(ITGL-2)1,3,5
C     ...................................................................
C     FIRST ITERATION : ASSUMED WCFJ(J)=0
C
    1 CONTINUE
      DO 2 J=1,JMAX
      WCFJ(J)=0.
    2 WCF1J(J)=0.
      RETURN
C     ...................................................................
C     SECOND ITERATION : ASSUMED WCFJ(J)=-0.5*(DPJ(J)-DPJ AV)*
C                                          MJ(J)/DPJ(J)
C
    3 CONTINUE
      WCFJT=0.
      DO 4 J=1,JMAX
      EP1J(J)=DPJ(J)-DPJAV
      WCFJ(J)=-0.5*EP1J(J)*MJ(J)/DPJ(J)
    4 WCFJT=WCFJT+WCFJ(J)
      GOTO 7
    5 CONTINUE
C     ...................................................................
C     ITGL>2: WCFJ(J) ARE OBTAINED BY USE OF THE TANGENT METHOD
C
      WCFJT=0.
      DO 6 J=1,JMAX
      EPJ=DPJ(J)-DPJAV
      IF(ABS(EP1J(J)-EPJ).LT.1.E-20)GOTO 6
      WCFJP=WCFJ(J)
      WCFJ(J)=WCFJP-FREL*EPJ*(WCFJP-WCF1J(J))/(EPJ-EP1J(J))
      WCF1J(J)=WCFJP
      EP1J(J)=EPJ
    6 WCFJT=WCFJT+WCFJ(J)
    7 CONTINUE
C     ...................................................................
C     NORMALIZATION OF THE WCF(J): THEIR SUMMATION MUST BE =0
C
      WCFJT=WCFJT/AJT
      DO 8 J=1,JMAX
    8 WCFJ(J)=WCFJ(J)-WCFJT*AJ(J)
      RETURN
      END
```

```
      FUNCTION CSFUN(IRH,REAI,SQ8LIA,SQ8LIB,GA)
C-----------------------------------------------------------------------
C     CSFUN COMPUTES THE FACTOR CS=AS/2.5 FOR THE VELOCITY PROFILE
C     IN THE ZONES OUTSIDE THE TAU=0 LINE (IN THE CASE OF SMOOTH
C     RODS CSFUN=1)
C
      COMMON/COLAM2/COLAMA
      IF(IRH.EQ.2)GOTO 1
      CSFUN=1.
      RETURN
    1 PROV=SQRT(1.056+0.005*(SQ8LIA/SQ8LIB)**2)
      SQ8LIA=ABS(SQ8LIA)
      SQ8LIA=(2.5*ALOG(REAI/(SQ8LIA*PROV))+5.5*COLAMA-5.699)     /PROV
      SQ8LIA=ABS(SQ8LIA)
      CSFUN=(SQ8LIA-5.5*COLAMA)/(2.5*ALOG(REAI/SQ8LIA)-GA)
      RETURN
      END
```

```
      FUNCTION DAREA(L)
C ---------------------------------------------------------------------
C     DAREA = 2 * AREA OF THE SECTOR L.
C
      COMMON /SC01R/ NSECT,NSECP
      COMMON /SC06R/ ISU(132,2)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /SC02R/ P,D,Z,ZWC,H,LENGTH
      COMMON /DAT/   PIG
      COMMON /IND3/  NTYP( 42)
C
              NS=ISU(L,1)
              NTY=NTYP(NS)
              IF(L.GT.NSECP) GO TO 500
              GO TO (100,200,100),NTY
  100         DAREA   =PIG*D/3.0
              GO TO 1000
  200         DAREA   =PIG*D*0.5
              GO TO 1000
C
  500         GO TO (600,700,800),NTY
  600         WRITE(6,610)
  610         FORMAT( ' ERROR IN DAREA ')
              DAREA   =1.
              GO TO 1000
  700         DAREA   =PSTAR(ZWC,P)
              GO TO 1000
  800         DAREA   =4.*Z/SQRT(3.)
C
 1000 IF(DAREA.GT. 1.0E-06) GO TO 9999
      WRITE(6,1100) L,PIG,Z,ZWC,P,D,NTY,DAREA
 1100 FORMAT( ' DAREA, DAREA LESSER THAN 0.0 FOR L = ',I4,/,
     >      5X,' PIG      = ',E12.6,/,
     >      5X,' Z        = ',E12.6,/,
     >      5X,' ZWC      = ',E12.6,/,
     >      5X,' P        = ',E12.6,/,
     >      5X,' D        = ',E12.6,/,
     >      5X,' NTY      = ',I2,/,
     >      5X,' DAREA    = ',E12.6,/,
     >          ' CALCULATION STOPS.')
      STOP
C
 9999 RETURN
      END
```

```
      SUBROUTINE DDONNE(TWO,TBT,GHPL,RODR2,R1DR2,YDH,R2MROH,FF,T2,T1,TE)
C-----------------------------------------------------------------------
C     DDONNE EVALUATES THE TEMPERATURES T1 AND T2 OF THE TWO REGIONS OF
C     CORNER CHANNELS AND OF THE 'WALL PART' OF WALL SUBCHANNELS
C
      RODR22=RODR2**2
      R1DR22=R1DR2**2
      F1=1.-RODR22
      F2=1.-R1DR22
      F3=RODR22-R1DR22
      T2=TWO-FF*(GHPL+2.5/F1*(F2*ALOG(YDH+R2MROH)-F3*ALOG(YDH)-0.5*(1.+
     +2.*R1DR2-RODR22-2.*R1DR2*RODR2)))
      T1=F2/F3*TBT-F1/F3*T2
      IF(T1.GE.TE .AND. T2.GE.TE)RETURN
C
      T2=TE
      T1=F2/F3*TBT-F1/F3*T2
      RETURN
      END
```

```
      SUBROUTINE DECP(IPRINT,X1,X2,STLEN)
C ----------------------------------------------------------------------
C     PRINT CONTROL
C
C     NPRINT <  0  PRINTS THE RESULTS AT EACH AXIAL SECTION.
C     NPRINT =  0  PRINTS ONLY AT THE SECTIONS BETWEEN AR(1) AND AR(2)
C     NPRINT >  0  PRINTS AT THE SECTIONS BY A(1),...A(NPRINT)
C                  (NPRINT <= 10)
C
      COMMON /SC25C/ NPRINT,AR(10)
C
      D1=X1+STLEN
      D2=X2+STLEN
      DM=(D1+D2)*0.5
      IF(NPRINT) 100,200,300
  100 IPRINT=1
      GO TO 999
  200 IPRINT=0
      IF(DM.GT.AR(1) .AND. DM.LT.AR(2)) IPRINT=1
      GO TO 999
  300 IPRINT=0
      DO 400 IPR=1,NPRINT
             IF( AR(IPR).GE.D1 .AND.  AR(IPR).LE.D2) IPRINT=1
  400        CONTINUE
  999 RETURN
      END
```

```
      SUBROUTINE DELIP
C ------------------------------------------------------------------
C     DETERMINES THE ARRAYS LIPS AND NCAN
C
      DIMENSION LIP(6)
C
      COMMON /SC01C/ NCAN( 19),LIPS( 19,10)
      COMMON /SC03C/ NRODS
      COMMON /MART5/ NSTR
      COMMON /GASD1/ NSTOT
      COMMON /HEA6 / NPIN( 42),JPIN( 42,3)
      COMMON /IJ1  / NER( 42),NIS( 42,3)
C
      DO 1000 J=1,NRODS
C
         K=0
         DO 400 NS=1,NSTOT
            NP=NPIN(NS)
            DO 300 M=1,NP
               IF(JPIN(NS,M).NE. J) GO TO 300
               IF(K.LT.1) GO TO 200
               DO 100 L=1,K
                  IF(LIP(L).EQ.NS) GO TO 400
  100             CONTINUE
  200          K=K+1
               LIP(K)=NS
  300          CONTINUE
  400       CONTINUE
         IF(K .LE. 0) GO TO 1000
         NCAN(J)=K
C
C
  450    I=1
         LIPS(J,I)=LIP(I)
  500    CONTINUE
         IF(I.EQ.K) GO TO 990
         I=I+1
         DO 900 L=I,K
            NS=LIP(L)
            NR=NER(NS)
            DO 800 M=1,NR
               IF(NIS(NS,M).NE.LIPS(J,I-1)) GO TO 800
               LIPS(J,I)=NS
               IF(L.NE. I) LIP(L)=LIP(I)
               GO TO 500
  800          CONTINUE
  900       CONTINUE
         NN=LIP(1)
         K1=K-1
         DO 950 LL=1,K1
                LIP(LL)=LIP(LL+1)
  950           CONTINUE
         LIP(K)=NN
         GO TO 450
  990    CONTINUE
 1000    CONTINUE
      WRITE(6,1001)
      DO 2000 J=1,NRODS
            NC=NCAN(J)
            IF(NC.LE.0) GO TO 2000
            WRITE(6,*) J,NC,(LIPS(J,M),M=1,NC)
```

```
 2000         CONTINUE
C
      RETURN
 1001 FORMAT(///5X,'SUBR. DELIP',/10X,
     >              'PIN    N. OF SECTORS      |--> CHANNEL ADJACENT TO ',
     >              'EACH SECTOR')
      END
```

```
      FUNCTION DPIN(MFLOW,PE,TE,CINL)
C -----------------------------------------------------------------------
C     DETERMINES THE PRESSURE LOSS AT THE BUNDLE INLET.
C
      COMMON /GEO2/ ATOT,DETOT,ASEC
      COMMON /SC01L/ STLEN
      REAL   MFLOW
C
      RE=MFLOW*DETOT/(ATOT*ETA(PE,TE))
      DPIN=CINL+2000.0/RE
      IF (STLEN.GT. 0.0000) DPIN=0.0
      IF (DPIN.GT.2.) DPIN=2.
      RETURN
      END
```

```
      FUNCTION EINF(BIOT)
C     EINF EVALUATES THE E INFINITE VALUE
C------------------------------------------------------------------------
C
      COMMON/BIDAT1/BI4,BI5,BI6,BI7,BI8,BI9,BI10
      IF(BIOT.GT.BI4)GOTO 1
      EINF=BI5+BI6*BIOT+BI7*BIOT**2
      RETURN
    1 EINF=BI8+BI9*BIOT+BI10*BIOT**2
      RETURN
      END
```

```
      SUBROUTINE ENFRCO
C     ------------------------------------------------------------------
C     ENFRCO COMPUTES AN AVERAGE GAMMA VALUE FOR THE LAMINAR
C     CALCULATIONS IF WALL AND CORNER CHANNELS ARE COMPUTED TOGETHER
C     (CASE OF NS1 ¬= NS2 ¬= 0 )
C
      REAL LAM,LAMSCH,LAMWC
      COMMON/MART5/NSTR
      COMMON/SUB5/LAMSCH( 42,3)/GEN1/LAM( 42)/GEN2/A( 42)/GEN5/DE( 42)
     1      /LAMIN1/AKAPPA( 42)/LAMIN2/FATIP(3),FDTIP(3)/IND3/NTYP( 42)
     2      /MART2/NS1,NS2/HEA6/NPIN( 42),JPIN( 42,3)/GAMCO/CGAMMA( 18)
     3      /WCSE3/LAMWC( 18,2,2)
      P=0.
      PP=0.
      DO 100 NS=NS1,NS2
      ITYP=NTYP(NS)
      ADDK=A(NS)*FATIP(ITYP)*(DE(NS)*FDTIP(ITYP))**2/AKAPPA(NS)
      P=P+ADDK
  100 PP=PP+ADDK/CGAMMA(NS-NSTR)
      P=P/PP
      DO 120 NS=NS1,NS2
      ITYP=NTYP(NS)
      PDCG=P/CGAMMA(NS-NSTR)
      LAM(NS)=LAM(NS)*PDCG
      NP=NPIN(NS)
      DO 120 M=1,NP
      LAMSCH(NS,M)=LAMSCH(NS,M)*PDCG
      IF(ITYP.EQ.3)GOTO 120
      DO 110 JWC=1,2
  110 LAMWC(NS-NSTR,M,JWC)=LAMWC(NS-NSTR,M,JWC)*PDCG
  120 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ENFRC1
C     ------------------------------------------------------------------
C     ENFRCO COMPUTES AN AVERAGE GAMMA VALUE FOR THE LAMINAR
C     CALCULATIONS IF WALL AND CORNER CHANNELS ARE COMPUTED TOGETHER
C     (CASE OF JLAM=1)
C
      REAL LAM,LAMSCH,LAMWC
C
      COMMON /MART5/ NSTR
      COMMON /SUB5/  LAMSCH( 42,3)
      COMMON /GEN1/  LAM( 42)
      COMMON /GEN2/  A( 42)
      COMMON /GEN5/  DE( 42)
      COMMON /LAMIN1/AKAPPA( 42)
      COMMON /LAMIN2/FATIP(3),FDTIP(3)
      COMMON /IND3/  NTYP( 42)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /GAMCO/ CGAMMA( 18)
      COMMON /WCSE3/ LAMWC( 18,2,2)
      COMMON /IJ1  / NER( 42),NIS( 42,3)
      COMMON /GASD1/ NSTOT
C
      DO 1000 NS=1,NSTOT
         ITYP=NTYP(NS)
         IF(ITYP .NE. 3) GO TO 1000
C
         P  = A(NS)*FATIP(ITYP)*(DE(NS)*FDTIP(ITYP))**2/AKAPPA(NS)
         PP = P/CGAMMA(NS-NSTR)
C
         NR = NER(NS)
         DO 100 M=1,NR
            NA=NIS(NS,M)
            I1TYP=NTYP(NA)
            AD=A(NA)*FATIP(I1TYP)*(DE(NA)*FDTIP(I1TYP))**2/AKAPPA(NA)
            P=P+AD
            PP=PP+AD/CGAMMA(NA-NSTR)
  100       CONTINUE
C
         P=P/PP
         GCOR=P/CGAMMA(NS-NSTR)
         LAM(NS)=LAM(NS)*GCOR
         DO 500 M=1,NR
            NA=NIS(NS,M)
            GCOR=P/CGAMMA(NA-NSTR)
            LAM(NA)=LAM(NA)*GCOR
            NP=NPIN(NA)
            DO 400 L=1,NP
               LAMSCH(NA,L)=LAMSCH(NA,L)*GCOR
               DO 300 J=1,2
                  LAMWC(NA-NSTR,L,J)=LAMWC(NA-NSTR,L,J)*GCOR
  300             CONTINUE
  400          CONTINUE
  500       CONTINUE
 1000    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE ENTRFR(K,I,ITYP,R1,R0,R2,NS,III,JJJ,DE,A,M,P,TB,LAMLAM)
C     ------------------------------------------------------------------
C     ENTRFR COMPUTES THE GAMMA FACTORS TO CORRECT THE FRICTION FACTORS
C     IN THE HYDRODYNAMIC ENTRANCE REGION
C
      REAL M,LAMLAM
      COMMON /SC01L/ STLEN
      COMMON/GRID3/X(100)/RETEM/TNY/LAMIN1/AKAPPA( 42)/GAMCO/CGAMMA( 18)
     1      /ENTR1/CKAPPA(2),DEA(2),GAMMA(2),WGAMMA(2),A1/HEA6/NPIN( 42)
     2        ,JPIN( 42,3)
      RE=M*DE/(A*RHO(P,TB))*RHO(P,TNY)/ETA(P,TNY)
      IF(ITYP.EQ.1 .OR. I.EQ.2)CALL NEWTON(R0,R1,R2)
      R1DR2=R1/R2
      R0DR1=R0/R1
      DEA(I)=2.*(R2-R1)
      CKAPPA(I)=FKAPPA(R1DR2)
      DKAPPA=AKAPPA(NS)
      IF(I.EQ.2)DKAPPA=    GKAPPA(R0DR1)
      REA=RE*DEA(I)/DE
      IF(ITYP.EQ.1 .OR. I.EQ.2)REA=RE *DKAPPA/CKAPPA(I)*(DEA(I)/DE)**3
      PHIDX=4./(DEA(I)*REA)
      PHIA1=PHIDX*(X(K)    + STLEN)
      PHIA2=PHIDX*(X(K+1) + STLEN)
      AKA1=AKA(R1DR2,PHIA1)
      AKA2=AKA(R1DR2,PHIA2)
      GAMMA(I)=1.+4./CKAPPA(I)*(AKA2-AKA1)/(PHIA2-PHIA1)
      IF(ITYP.EQ.2)GOTO 10
      LAMLAM=LAMLAM*GAMMA(1)
      IF(ITYP.EQ.3)CGAMMA(III)=GAMMA(1)
      RETURN
C     ..................................................................
C     ONLY FOR THE WALL SUBCHANNELS
C
   10 IF (I .NE.1) GO TO 11
      A1=A
      RETURN
   11 C1=A1*DEA(1)**2/CKAPPA(1)
      C2=A*DE**2/DKAPPA
      WGAMMA(JJJ)=(C1+C2)/(C1/GAMMA(1)+C2/GAMMA(2))
      LAMLAM=LAMLAM*WGAMMA(JJJ)
      IF(JJJ.LT.NPIN(NS))RETURN
      CGAMMA(III)=0.
      NP=NPIN(NS)
      DO 20 JJ=1,NP
   20 CGAMMA(III)=CGAMMA(III)+WGAMMA(JJ)
      CGAMMA(III)=CGAMMA(III)/FLOAT(NP)
      RETURN
      END
```

```
      FUNCTION EPS(T,L)
C -----------------------------------------------------------------
C     DETERMINES THE EMISSIVITY OF THE SURFACES.
C     IEPS=0 ==> UNIFORM EMISSIVITY OVER THE WHOLE BUNDLE
C                (BUT INDIPENDENT VALUES FOR PINS AND SHROUD)
C     IEPS=1 ==> EMISSIVITY FUNCTION OF THE TEMPERATURE OF THE SURFACE
C                (T IN CELTIUS)
C
      COMMON /SC01R/  NSECT,NSECP
      COMMON /SC07R/  EPSR,EPSS,SIGMA
      COMMON /SC15R/  IEPS
C
      IF (IEPS .GT. 0) GO TO 100
             EPS=EPSR
             IF(L.GT.NSECP) EPS=EPSS
             GO TO 1000
  100 CONTINUE
      EPS=0.42+0.42*(T-500.0)/350.0
      IF(T.LT.500.0) EPS=0.42
      IF(T.GT.850.0) EPS=0.84
C
 1000 RETURN
      END
```

```
      FUNCTION ETA(P,T)
C-----------------------------------------------------------------------
C     ETA EVALUATES THE DYNAMIC VISCOSITY OF THE COOLANT (G/CM S)
C
      COMMON/GASD4/IGAS
      GOTO(10,20,30,40),IGAS
   10 CONTINUE
C     CASE OF HELIUM COOLANT
C
      ETA=18.84E-05*((T+273.16)/273.16)**0.66
      RETURN
C
   20 CONTINUE
C     CASE OF CO2 COOLANT
C
      PP=P
      TT=T
      P=PP/1.0333
      T=TT+273.16
      TO=273.16
      TF=TO/T
      ETAO=(1.54E-7*SQRT(T))/(1.+(228./T))
      ETAF=ETAO*(1.+4.78E-3*(P-1.)*(TF**3))
      ETA=ETAF*98.068
      P=PP
      T=TT
      RETURN
C
   30 CONTINUE
C     CASE OF N2 COOLANT.
C
      TT=T
      T=TT+273.16
      ETA0=1.425E-7*T**0.5/(1.+107./T)*98.0665
      ETA=ETA0*(1.+8.E-4*(P/1.033-1.)*(273.16/T))
      T=TT
C
      RETURN
   40 CONTINUE
      ETA=0.
      RETURN
      END
```

```
      FUNCTION EXPCL(T)
C     ------------------------------------------------------------------
C     EXPCL COMPUTES THE EXPANSION COEFFICIENTS FOR THE CORRECTION OF
C     THE GEOMETRICAL DIMENSIONS OF THE LINER
C
      COMMON/EXDAT1/EX4(7),EX5(7),EX6(7)/INPAR/IPA
      EXPCL=EX4(IPA)+EX5(IPA)*T+EX6(IPA)*T**2
      RETURN
      END
```

```
      FUNCTION EXPCO(T)
C-----------------------------------------------------------------------
C     EXPCO COMPUTES THE EXPANSION COEFFICIENTS FOR THE CORRECTION OF TH
C     GEOMETRICAL DIMENSIONS OF THE RODS
C
      COMMON/EXDAT/ EX1(7),EX2(7),EX3(7) /INPAR/IPA
      EXPCO=EX1(IPA)+EX2(IPA)*T+EX3(IPA)*T**2
      RETURN
      END
```

```
      SUBROUTINE FFA31(L,NAFF,NS,J)
C ----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE FA31
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      DO 1000 M=1,NR
              NS1=NIS(NS,M)
              NP1=NPIN(NS1)
              DO 500 M1=1,NP1
                     IF(JPIN(NS1,M1).NE.J) GO TO 500
                     NW1=NS1-NSTR
                     L1=ISS(NW1,M1)
                     NAFF=NAFF+1
                     KAFF(L,NAFF)=L1
                     VFAC(L,NAFF)=F31
  500                CONTINUE
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FFS13(L,NAFF,NS,J)
C ---------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE FW13
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NP=NPIN(NS)
      NR=NER(NS)
      DO 1000 M=1,NR
             NS1=NIS(NS,M)
             IF(NTYP(NS1).NE.2) GO TO 1000
             NP1=NPIN(NS1)
             DO 100 M1=1,NP1
                   DO 50 M2=1,NP
                        IF(JPIN(NS,M2).NE.JPIN(NS1,M1)) GO TO 50
                        IF(JPIN(NS1,M1).NE. J) GO TO 1000
                        GO TO 150
   50                   CONTINUE
  100              CONTINUE
             WRITE(6,110)
  110        FORMAT( ' ERROR IN FFS13, LOOP 100')
  150        DO 200 M1=1,NP1
                   IF(JPIN(NS1,M1).EQ. J) GO TO 200
                   L1=IGI(NS1,M1)
                   NAFF=NAFF+1
                   VFAC(L,NAFF)=F13
                   KAFF(L,NAFF)=L1
                   GO TO 1000
  200              CONTINUE
 1000        CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FFS16(L,NAFF,NS,J)
C -----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE FS16
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      DO 1000 M=1,NR
              NS1=NIS(NS,M)
              IF(NTYP(NS1).NE.2) GO TO 1000
              NP1=NPIN(NS1)
              DO 100 M1=1,NP1
                     IF(JPIN(NS1,M1).NE.J) GO TO 100
                     L1=IGI(NS1,M1)
                     NAFF=NAFF+1
                     VFAC(L,NAFF)=F16
                     KAFF(L,NAFF)=L1
                     GO TO 1000
  100                CONTINUE
 1000          CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FFS24(L,NAFF,NS,J)
C ----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE FS24
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      DO 1000 M=1,NR
              NS1=NIS(NS,M)
              IF(NTYP(NS1).NE.3)   GO TO 1000
              IF(JPIN(NS1,1).NE.J) GO TO 1000
              L1=IGI(NS1,1)
              NAFF=NAFF+1
              VFAC(L,NAFF)=F24
              KAFF(L,NAFF)=L1
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FFW31(L,NAFF,NS,J)
C -----------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE FW31
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      DO 1000 M=1,NR
              NS1=NIS(NS,M)
              IF(NTYP(NS1) .NE. 3) GO TO 1000
              IF(JPIN(NS1,1).NE.J) GO TO 1000
                     NW1=NS1-NSTR
                     L1=ISS(NW1,1)
                     NAFF=NAFF+1
                     KAFF(L,NAFF)=L1
                     VFAC(L,NAFF)=F31
  500                CONTINUE
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FF32 (L,NAFF,NS,J)
C -----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F32
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      NP=NPIN(NS)
      DO 1000 M=1,NR
              NS1=NIS(NS,M)
              IF(NTYP(NS1) .NE. 2) GO TO 1000
              NP1=NPIN(NS1)
              DO 500 M1=1,NP1
                     IF(JPIN(NS1,M1).NE.J) GO TO 500
                     NW1=NS1-NSTR
                     L1=ISS(NW1,M1)
                     NAFF=NAFF+1
                     KAFF(L,NAFF)=L1
                     VFAC(L,NAFF)=F32
  500                CONTINUE
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FGEO (VDIAM,RSTAR,S,R,RINT)
C -----------------------------------------------------------------------
C     COMPUTES SOME GEOMETRICAL FACTORS FOR CONDUCTION.
C
      COMMON /SC04C/ RFUEL
C
      S = VDIAM/2. - RINT
      R = (VDIAM/2. + RINT)/2.
      RSTAR=(RFUEL+RINT)*0.5
C
      RETURN
      END
```

```
      SUBROUTINE FINDA(L,NAFF,NS,J)
C ----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS FOR A CORNER SECTOR.
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /MART5/ NSTR
C
      NW=NS-NSTR
      L1=ISS(NW,1)
      NAFF=NAFF+1
      VFAC(L,NAFF)=F14
      KAFF(L,NAFF)=L1
C
      NR=NER(NS)
      DO 1000 M=1,NR
              NS1=NIS(NS,M)
              NP1=NPIN(NS1)
              DO 100 M1=1,NP1
                     IF(JPIN(NS1,M1).NE.J) GO TO 100
                     NW1=NS1-NSTR
                     L1=ISS(NW1,M1)
                     NAFF=NAFF+1
                     VFAC(L,NAFF)=F24
                     KAFF(L,NAFF)=L1
  100                CONTINUE
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FINDF1(L,NAFF,NS,J)
C ----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F1
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
C
      NP=NPIN(NS)
      DO 1000 M=1,NP
              JP=JPIN(NS,M)
              IF(J .EQ.JP) GO TO 1000
              NAFF=NAFF+1
              L1=IGI(NS,M)
              VFAC(L,NAFF)=F1
              KAFF(L,NAFF)=L1
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FINDF3(L,NAFF,NS,J)
C ----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F3 / F15
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /MART5/ NSTR
      COMMON /IND3/  NTYP( 42)
C
      NR=NER(NS)
      DO 1000 M=1,NR
C
              NS1=NIS(NS,M)
              NP1=NPIN(NS1)
              DO 100 M1=1,NP1
                     IF(JPIN(NS1,M1).EQ.J) GO TO 1000
  100                CONTINUE
              IF(NTYP(NS1).GT.1) GO TO 500
              NP=NPIN(NS)
              DO 300 M1=1,NP1
                     DO 200 K=1,NP
                            IF(JPIN(NS1,M1).EQ.JPIN(NS,K)) GO TO 300
  200                       CONTINUE
                     J1=JPIN(NS1,M1)
                     GO TO 400
  300                CONTINUE
              WRITE(6,310)
  310         FORMAT( ' ERROR IN FINDF3, LOOP M1, CALC PROCEEDING')
  400         CONTINUE
C
              L1=IGI(NS1,M1)
              NAFF=NAFF+1
              VFAC(L,NAFF)=F3
              KAFF(L,NAFF)=L1
              GO TO 9999
C
  500         NSW1=NS1-NSTR
              L1=ISS(NSW1,1)
              L2=ISS(NSW1,2)
C
              VFAC(L,NAFF+1)=F15
              VFAC(L,NAFF+2)=F15
              KAFF(L,NAFF+1)=L1
              KAFF(L,NAFF+2)=L2
              NAFF=NAFF+2
              GO TO 9999
C
 1000         CONTINUE
      WRITE(6,1100)
 1100 FORMAT( ' ERROR IN FINDF3, LOOP M, CALC. PROCEEDING')
C
 9999 RETURN
      END
```

```
      SUBROUTINE FINDF4(L,NAFF,NS,J)
C -------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F4 / F8
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      NP=NPIN(NS)
      DO 1000 M=1,NR
C
            NS1=NIS(NS,M)
            NP1=NPIN(NS1)
            DO 100 M1=1,NP1
                  IF(JPIN(NS1,M1).EQ.J) GO TO 500
  100             CONTINUE
            IF(NTYP(NS1).GT.1) GO TO 300
            DO 200 M1=1,NP1
                  DO 150 M2=1,NP
                        IF(JPIN(NS1,M1).EQ.JPIN(NS,M2)) GO TO 250
  150                   CONTINUE
                  GO TO 200
  250             L1=IGI(NS1,M1)
                  NAFF=NAFF+1
                  VFAC(L,NAFF)=F4
                  KAFF(L,NAFF)=L1
  200             CONTINUE
            GO TO 1000
  300       CONTINUE
C
            DO 400 M1=1,NP1
                  NAFF=NAFF+1
                  L1=IGI(NS1,M1)
                  VFAC(L,NAFF)=F8
                  KAFF(L,NAFF)=L1
  400             CONTINUE
            GO TO 1000
C
  500       CONTINUE
            IF(NTYP(NS1).GT.1) GO TO 1000
            DO 700 M1=1,NP1
                  DO 600 M2=1,NP1
                        IF(JPIN(NS1,M1).EQ.JPIN(NS,M2)) GO TO 700
  600                   CONTINUE
                  NAFF=NAFF+1
                  L1=IGI(NS1,M1)
                  VFAC(L,NAFF)=F4
                  KAFF(L,NAFF)=L1
                  GO TO 1000
  700             CONTINUE
C
 1000       CONTINUE

      RETURN
      END
```

```
      SUBROUTINE FINDF5(L,NAFF,NS,J)
C ----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F5
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      NP=NPIN(NS)
      DO 1000 M=1,NR
C
            NS1=NIS(NS,M)
            NP1=NPIN(NS1)
            DO 100 M1=1,NP1
                  IF(JPIN(NS1,M1).EQ.J) GO TO 110
  100             CONTINUE
            GO TO 1000
  110       CONTINUE
            DO 200 M1=1,NP1
                  IF (JPIN(NS1,M1).EQ.J) GO TO 200
                  DO 150 M2=1,NP
                        IF(JPIN(NS1,M1).EQ.JPIN(NS,M2)) GO TO 180
  150                   CONTINUE
                  GO TO 200
  180             CONTINUE
                  F=F5
                  IF(NTYP(NS1).NE.1) F=F7
                  L1=IGI(NS1,M1)
                  NAFF=NAFF+1
                  VFAC(L,NAFF)=F
                  KAFF(L,NAFF)=L1
  200             CONTINUE
C
 1000 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FINDW2(L,NAFF,NS,J)
C ---------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F2
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
C
      NP=NPIN(NS)
      DO 1000 M=1,NP
            IF(JPIN(NS,M).EQ.J) GO TO 1000
            L1=IGI(NS,M)
            NAFF=NAFF+1
            VFAC(L,NAFF)=F2
            KAFF(L,NAFF)=L1
 1000       CONTINUE
C
      RETURN
      END
```

```
      FUNCTION FKAPPA(R)
C     -----------------------------------------------------------------
C     FKAPPA EVALUATES THE KAPPA VALUES FOR THE CORNER CHANNELS AND THE
C     WALL PORTION OF THE WALL SUBCHANNELS ( VALIDITY FOR CORNER CHANNE-
C     LS 1.2< W/D <1.5)
C
      FKAPPA=62.146*(1.-R)**2/(1.+R**2+(1.-R**2)/ALOG(R))
      RETURN
      END
```

```
      FUNCTION FQDEV(A,N,X1,X2)
C-----------------------------------------------------------------------
C     FQDEV INTEGRATES THE PROFILES OF POWER
C
      DIMENSION A(N)
      FQDEV=0.
      X1AI=0.
      DO 10 I=1,N
      AI=I
      IF(X1.GT.0.)X1AI=X1**AI
   10 FQDEV=FQDEV+A(I)*(X2**AI-X1AI)
      RETURN
      END
```

```
      SUBROUTINE FSA  (L,NAFF,NS,J)
C ----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS FOR A CORNER SECTOR OF THE SHROUD.
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /MART5/ NSTR
C
      L1=IGI(NS,1)
      NAFF=NAFF+1
      VFAC(L,NAFF)=F14
      KAFF(L,NAFF)=L1
C
      NR=NER(NS)
      DO 1000 M=1,NR
            NS1=NIS(NS,M)
            NP1=NPIN(NS1)
            DO 100 M1=1,NP1
                  F=F17
                  IF(JPIN(NS1,M1).NE.J) F=F18
                  L1=IGI(NS1,M1)
                  NAFF=NAFF+1
                  VFAC(L,NAFF)=F
                  KAFF(L,NAFF)=L1
  100             CONTINUE
 1000       CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FS1112(L,NAFF,NS,J)
C ---------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE FS11-FS12
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /MART5/ NSTR
C
      NP=NPIN(NS)
      DO 1000 M1=1,NP
      F=F11
      IF(JPIN(NS,M1).NE.J) F=F12
C
      L1=IGI(NS,M1)
      NAFF=NAFF+1
      VFAC(L,NAFF)=F
      KAFF(L,NAFF)=L1
C
 1000 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FS2   (L,NAFF,NS,J)
C -----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE FS15
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      NP=NPIN(NS)
      DO 1000 M=1,NR
              NS1=NIS(NS,M)
              NP1=NPIN(NS1)
              IF(NTYP(NS1).NE. 1) GO TO 1000
              DO 500 M1=1,NP1
                     DO 400 M2=1,NP
                            IF(JPIN(NS1,M1).EQ.JPIN(NS,M2)) GO TO 500
  400                       CONTINUE
                     L1=IGI(NS1,M1)
                     NAFF=NAFF+1
                     KAFF(L,NAFF)=L1
                     VFAC(L,NAFF)=F15
  500                CONTINUE
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FW1112(L,NAFF,NS,J)
C ----------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE FW11-FW12
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /MART5/ NSTR
C
      NP=NPIN(NS)
      DO 1000 M1=1,NP
      F=F11
      IF(JPIN(NS,M1).NE.J) F=F12
C
      N1=NS-NSTR
      L1=ISS(N1,M1)
      NAFF=NAFF+1
      VFAC(L,NAFF)=F
      KAFF(L,NAFF)=L1
C
 1000 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FW13(L,NAFF,NS,J)
C ------------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE FW13
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NP=NPIN(NS)
      NR=NER(NS)
      DO 1000 M=1,NR
              NS1=NIS(NS,M)
              IF(NTYP(NS1).NE.2) GO TO 1000
              NP1=NPIN(NS1)
              DO 100 M1=1,NP1
                     IF(JPIN(NS1,M1).EQ.J) GO TO 1000
  100                CONTINUE
              DO 300 M1=1,NP1
                  DO 200 M2=1,NP
                         IF(JPIN(NS,M2).EQ.JPIN(NS1,M1)) GO TO 250
  200                    CONTINUE
                  GO TO 300
  250             NW1=NS1-NSTR
                  L1=ISS(NW1,M1)
                  NAFF=NAFF+1
                  VFAC(L,NAFF)=F13
                  KAFF(L,NAFF)=L1
                  GO TO 1000
  300             CONTINUE
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FW16(L,NAFF,NS,J)
C -------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F16
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      DO 1000 M=1,NR
              NS1=NIS(NS,M)
              IF(NTYP(NS1).NE.2) GO TO 1000
              NP1=NPIN(NS1)
              DO 100 M1=1,NP1
                     IF(JPIN(NS1,M1).NE.J) GO TO 100
                     NW1=NS1-NSTR
                     L1=ISS(NW1,M1)
                     NAFF=NAFF+1
                     VFAC(L,NAFF)=F16
                     KAFF(L,NAFF)=L1
                     GO TO 1000
  100                CONTINUE
 1000          CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FW1718(L,NAFF,NS,J)
C ---------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F17-F18
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC11R/ ISS ( 18,2)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      NR=NER(NS)
      DO 1000 M=1,NR
          NS1=NIS(NS,M)
          IF(NTYP(NS1).NE.3) GO TO 1000
          F=F18
          IF(JPIN(NS1,1).EQ.J) F=F17
          NW1=NS1-NSTR
          L1=ISS(NW1,1)
          NAFF=NAFF+1
          VFAC(L,NAFF)=F
          KAFF(L,NAFF)=L1
 1000     CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FW7    (L,NAFF,NS,J)
C ---------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F7
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
C
      NR=NER(NS)
      NP=NPIN(NS)
      DO 1000 M=1,NR
           NS1=NIS(NS,M)
           IF (NTYP(NS1).NE.1) GO TO 1000
           NP1=NPIN(NS1)
           DO 500 M1=1,NP1
                IF(JPIN(NS1,M1).EQ.J) GO TO 500
                DO 400 M2=1,NP
                     IF(JPIN(NS1,M1).NE.JPIN(NS,M2)) GO TO 400
                     L1=IGI(NS1,M1)
                     NAFF=NAFF+1
                     VFAC(L,NAFF)=F7
                     KAFF(L,NAFF)=L1
  400                CONTINUE
  500           CONTINUE
 1000      CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FW8    (L,NAFF,NS,J)
C ---------------------------------------------------------------------
C     DETECTS THE VIEW-FACTORS OF TYPE F8
C
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC12R/ IGI ( 42,3)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IJ1 /  NER( 42),NIS( 42,3)
      COMMON /IND3/  NTYP( 42)
C
      NR=NER(NS)
      NP=NPIN(NS)
      DO 1000 M=1,NR
            NS1=NIS(NS,M)
            IF (NTYP(NS1).NE.1) GO TO 1000
            NP1=NPIN(NS1)
            DO 500 M1=1,NP1
                  DO 400 M2=1,NP
                        IF(JPIN(NS1,M1).EQ.JPIN(NS,M2)) GO TO 500
  400                   CONTINUE
                  L1=IGI(NS1,M1)
                  NAFF=NAFF+1
                  VFAC(L,NAFF)=F8
                  KAFF(L,NAFF)=L1
  500             CONTINUE
 1000       CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE GAUDEL (A,B,X,PERC,ITMAX)
C ------------------------------------------------------------------
C     SOLVES A SYSTEM OF LINEAR EQUATIONS USING THE GAUSS-SEIDEL METHOD.
C
      DIMENSION A(132,13),B(132),X(132)
      COMMON /SC01R/ NSECT,NSECP
      COMMON /SC13R/ NAFF(132)
      COMMON /SC14R/ KAFF(132, 13)
C
      DO 100 I=1,NSECT
      X(I)=0.0
  100 CONTINUE
C
      DO 2000 IT=1,ITMAX
              BIG=0.0
              DO 1000 L1=1,NSECT
                      P=B(L1)
                      NAF=NAFF(L1)
                      NAF1=NAF+1
                      DO 500 J=1,NAF
                             L2=KAFF(L1,J)
                             P=P-A(L1,J)*X(L2)
  500                        CONTINUE
                      TEMP=P/A(L1,NAF1)
                      TEMP1= ABS( X(L1)/TEMP -1.)
                      IF (TEMP1 .GT. BIG) BIG=TEMP1
                      X(L1)=TEMP
 1000                 CONTINUE
              IF (BIG . LT. PERC) GO TO 3000
 2000         CONTINUE
C
C     NO CONVERGENCE -----------------------------------------------------
C
      BIG1=BIG*100.
      PER1=PERC*100.
      WRITE(6,2500) IT,BIG1,PER1
 2500 FORMAT( 5X, 'SUB. GAUDEL ',/5X,
     >            'NO CONVERGENCE OBTAINED FOR ',I4,' ITERATIONS.',/5X,
     >            'THE MAXIMAL DELTA X    IS ',E12.6,' %',/5X,
     >            'THE REQUIRED PRECISION IS ',E12.6,' %',/5X,
     >            'CALCULATION STOPS.')
      STOP
C
C     -- CONVERGENCE -----------------------------------------------------
C
 3000 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE GAUSS(A,B,X)
C -------------------------------------------------------------------
C     SOLVES THE SYSTEM WITH THE GAUSS METHOD WITH PIVOTAL CONDENSATION
C
      DIMENSION A( 34, 34),B( 34),X( 34)
      COMMON /SC22C/ NTOT
C
C
      N1=NTOT-1
      DO 1000 N=1,N1
              CALL PIVOT(A,B,N)
              I1=N+1
              DO 800 I=I1,NTOT
                     DM=A(I,N)/A(N,N)
                     DO 600 J=I1,NTOT
                            A(I,J)=A(I,J)-DM*A(N,J)
  600                       CONTINUE
                     A(I,N)=0.0
                     B(I)=B(I)-DM*B(N)
  800                CONTINUE
 1000         CONTINUE
C
C --- BACK SUBSTITUTION ---------------------------------------------
C
      X(NTOT)=B(NTOT)/A(NTOT,NTOT)
      I=NTOT-1
 1100 J1=I+1
      S=0.0
      DO 1200 J=J1,NTOT
              S=S+A(I,J)*X(J)
 1200         CONTINUE
      X(I)=(B(I)-S)/A(I,I)
      I=I-1
      IF(I.GT.0) GO TO 1100
C
      RETURN
      END
```

```
      SUBROUTINE GAUSS1(A,B,X,NTOT,N1)
C ---------------------------------------------------------------------
C     SOLVES A SYSTEM OF LINEAR EQUATIONS USING THE GAUSS ELIMINATION
C     METHOD WITH PIVOTAL CONDENSATION
C
      DIMENSION A(N1,NTOT),B(NTOT),X(NTOT)
C
C --- ELIMINATION -----------------------------------------------------
      N2=NTOT-1
      DO 1000 N=1,N2
              CALL PIVOT1(A,B,N,NTOT,N1)
              I1=N+1
              DO 800 I=I1,NTOT
                     DM=A(I,N)/A(N,N)
                     DO 600 J=I1,NTOT
                            A(I,J)=A(I,J)-DM*A(N,J)
  600                       CONTINUE
                     A(I,N)=0.0
                     B(I)=B(I)-DM*B(N)
  800                CONTINUE
 1000         CONTINUE
C
C --- BACK SUBSTITUTION -----------------------------------------------
      X(NTOT)=B(NTOT)/A(NTOT,NTOT)
      I=NTOT-1
 1100 J1=I+1
      S=0.0
      DO 1200 J=J1,NTOT
              S=S+A(I,J)*X(J)
 1200         CONTINUE
      X(I)=(B(I)-S)/A(I,I)
      I=I-1
      IF(I.GT.0) GO TO 1100
C
      RETURN
      END
```

```
      FUNCTION GHPLUS(HPLUSW,TW,TBT,PR,YDH,REW,R2MROH,RX,R1)
C-----------------------------------------------------------------------
C     GHPLUS EVALUATES THE FUNCTION G(H+)=G(HW+,PRANDTL,TW/TB,Y/RH)
C
      COMMON/DAT1/A1,A2,A3,A4,A5,A6,A7,A8,A9,A10
      GHPL  =(A1*HPLUSW**A2+A3/HPLUSW**A4)
      IF(GHPL.LE.A9)GHPL=A10
      GHPLUS=GHPL*PR**A5*((TW+273.16)/(TBT+273.16))**A6/(A7*(YDH+R2MROH)
     *)**A8
      RETURN
      END
```

```
      FUNCTION GKAPPA(X)
C     ------------------------------------------------------------------
C     GKAPPA EVALUATES THE KAPPA VALUES FOR THE CENTRAL SUBCHANNELS AND
C     THE CENTRAL PORTIONS OF THE WALL SUBCHANNELS ( VALIDITY FOR THE
C     CENTRAL CHANNELS AT 1.2< P/D <1.5)
C
      GKAPPA=54.237*(X**2-1.)**3*X**0.342/ABS(3.*X**4-4.*X**2-4.*X**4*
     *ALOG(X)+1.)
      RETURN
      END
```

```
      FUNCTION GRIFUN(EPS)
C     ------------------------------------------------------------------
C     GRIFUN EVALUATES THE COEFFICIENT K0/2=(KI+KO)/2 FOR THE LOCAL
C     PRESSURE LOSSES AT THE INLET AND AT THE OUTLET OF A SPACER
C
      GRIFUN     =0.5*(EPS*0.5+EPS**2)/(1.-EPS)**2
      RETURN
      END
```

```
      FUNCTION GSTAR(EPS)
C-----------------------------------------------------------------------
C     GSTAR EVALUATES THE FUNCTION G(EPSILON)
C
      COMMON/SUBLA/CLASUB
      GSTAR=(3.75*CLASUB+1.25*EPS)/(1.+EPS)+2.5*ALOG(2.*(EPS+1))
      RETURN
      END
```

```
      SUBROUTINE HEATBA(IND,PBT,INDQ,TE,MFLOW)
C-----------------------------------------------------------------------
C     HEAT BALANCE FOR THE RODS, THE SHROUD AND THE COOLANT
C
      COMMON /GASD1/ NSTOT
      COMMON /MART5/ NSTR
      COMMON /SC03C/ NRODS
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /SUB2/  TSCH( 42,3),MSCH( 42,3)
      COMMON /SUB6/  TSCH1( 42,3)
      COMMON /SUB8/  MSCH1( 42,3)
      COMMON /HEA5/  QQ( 42,3)
      COMMON /QPAR1/ QDEV
      COMMON /SC02R/ P,D,Z,ZWC,H,LENGTH
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /IND3/  NTYP( 42)
      COMMON /SC06L/ SHQ( 18,2)
      COMMON /SC21C/ SHQC( 18,2)
      COMMON /SC13C/ GEO1( 42,3)
C
      DIMENSION TOCAN( 42,3),TOPIN( 19),TC( 42,3),TGP( 19)
      REAL LENGTH,MFLOW,MOUT,MSCH,MSCH1,MMED
C
      IF(IND) 100,500,1000
C
  100 CONTINUE
      TOLIN=0.0
      TOROD=0.0
      TOGAS=0.0
      MOUT =0.0
      DO 300 NS=1,NSTOT
             DO 200 M=1,3
                    TOCAN(NS,M) = 0.0
                    TC(NS,M)    = 0.0
  200               CONTINUE
  300        CONTINUE
      DO 400 J=1,NRODS
             TOPIN(J)     = 0.0
             TGP(J)       = 0.0
  400        CONTINUE
      GO TO 9999
C
  500 CONTINUE
      TMED=0.0
      MMED=0.0
      DO 900 NS=1,NSTOT
             NP=NPIN(NS)
             DO 800 M=1,NP
                    J=JPIN(NS,M)
                    TMED=TMED+MSCH(NS,M)*TSCH1(NS,M)
                    MMED=MMED+MSCH(NS,M)
                    QG=QQ(NS,M)*QDEV/LENGTH/GEO1(NS,M)*H
                    TOROD=TOROD+QG
                    TGP(J)=TGP(J)+QG
                    TOPIN(J)=TOPIN(J)+QG+QJ(J,NS)
                    TC(NS,M)=TC(NS,M)+QG+QJ(J,NS)
                    DT=(TSCH1(NS,M)-TSCH(NS,M))*2.
                    Q=MSCH(NS,M)*CP(PBT,TSCH(NS,M))*DT
                    TOCAN(NS,M)=TOCAN(NS,M)+Q
                    TOGAS=TOGAS+Q
                    IF( NTYP(NS) .EQ. 1 ) GO TO 800
```

```
                 NSW=NS-NSTR
                 TOLIN=TOLIN+SHQ(NSW,M)
                 TC(NS,M)=TC(NS,M)+SHQ(NSW,M)+SHQC(NSW,M)
                 TOGAS=TOGAS+SHQ(NSW,M)+SHQC(NSW,M)
  800            CONTINUE
  900        CONTINUE
      TMED=TMED/MMED
      GO TO 9999
C
 1000 CONTINUE
      F=1.000
      IF(INDQ.EQ.2) F=4.186
      TOLIN=TOLIN*F
      TOROD=TOROD*F
      TOGAS=TOGAS*F
      DO 1100 J=1,NRODS
              TOPIN(J) = TOPIN(J)*F
              TGP(J)   = TGP(J)*F
 1100         CONTINUE
      DO 1300 NS=1,NSTOT
              NP=NPIN(NS)
              DO 1200 M=1,NP
                      MOUT=MOUT+MSCH1(NS,M)
                      TOCAN(NS,M)=TOCAN(NS,M)*F
                      TC(NS,M)    =TC(NS,M)*F
 1200                 CONTINUE
 1300         CONTINUE
      WRITE(6,2000)
      IF(INDQ.EQ.1) WRITE(6,2100)
      IF(INDQ.EQ.2) WRITE(6,2200)
      WRITE(6,2300)
      DO 1400 J=1,NRODS
              WRITE(6,2400) J,TGP(J),TOPIN(J)
 1400         CONTINUE
      WRITE(6,2500)
      DO 1600 NS=1,NSTOT
              NP=NPIN(NS)
              DO 1500 M=1,NP
                      WRITE(6,2600) NS,M,TC(NS,M),TOCAN(NS,M)
 1500                 CONTINUE
 1600         CONTINUE
      TOGEN=TOROD+TOLIN
      DELTA=ABS((TOGEN/TOGAS)-1.)*100.
      DM=   ABS((MFLOW/MOUT )-1.)*100.
      DELT = TMED - TE
      WRITE(6,2700) TOROD,TOLIN,TOGEN,TOGAS,DELTA
      WRITE(6,2800) MFLOW,MOUT,DM
      WRITE(6,2900) TE,TMED,DELT
C
 9999 RETURN
C
 2000 FORMAT(1H1,//5X,' FINAL BALANCE                UNIT=')
 2100 FORMAT(1H+,40X,'(CAL/SEC), (G/SEC)')
 2200 FORMAT(1H+,40X,'(WATT)    , (G/SEC)')
 2300 FORMAT(///4X,' PIN       HEAT GENERATED',15X,'HEAT TO GAS',/)
 2400 FORMAT( 4X,I3,10X,E12.6,15X,E12.6)
 2500 FORMAT(///4X,' CHANNEL     M    HEAT TRANSMITTED          ',
     >               ' HEAT INCREASE',/)
 2600 FORMAT(6X,I3,7X,I1,5X,E12.6,12X,E12.6)
 2700 FORMAT(///5X,' TOTAL POWER GENERATED IN THE RODS    =',E12.6,
     >         /5X,' TOTAL POWER GENERATED IN THE SHROUD =',E12.6,
     >         /5X,' TOTAL POWER GENERATED               =',E12.6,
     >         /5X,' TOTAL POWER INCREASE IN THE GAS     =',E12.6,
     >         /5X,' (SUM OF THE PARTIAL M*CP*DT)         ',
     >         /5X,' PERCENTUAL ERROR                    =',F5.2,' %')
```

```
 2800 FORMAT(///5X,' INLET  MASS FLOW RATE                  =',E12.6,
     >          /5X,' OUTLET MASS FLOW RATE                  =',E12.6,
     >          /5X,' ERROR                                  =',F5.2,' %')
 2900 FORMAT(///5X,' INLET  TEMPERATURE                     =',F6.2,' C',
     >          /5X,' OUTLET TEMPERATUE (AVERAGE)            =',F6.2,' C',
     >          /5X,' TEMPERATURE INCREASE                   =',F6.2,' C')
C
C
      END
```

```
      SUBROUTINE HEATI(NSTOT,NSTR,NSEL,NROMA,IPA)
C-----------------------------------------------------------------------
C     HEATI EVALUATES THE HEAT FLUXES QQ(NS,I) FOR THE RODS ADJACENT TO
C     EACH CHANNEL NS AND THE TOTAL FLUXES QT(NS) ENTERING EACH
C     CHANNEL NS.  HEATI IDENTIFIES ALSO THE CONNECTIONS BETWEEN THE
C     SUBCHANNELS I AND THE ADJACENT RODS BY MEANS OF THE MATRIX JPIN
C     ( NPIN(NS)= NR. OF SUBCH. IN CH. NS = NR. OF PINS ADJ. TO CH. NS)
C
C                VERSION FOR HEXAGONAL BUNDLES
C     ..................................................................
      COMMON/IND1/NROW( 42),NUMS( 42)/HEA2/Q(3, 18),Q0/HEA3/QT( 42)
     1      /HEA5/QQ( 42,3)/HEA6/NPIN( 42),JPIN( 42,3)/HEA7/IDPIN(3, 18)
     2      /IND4/NUM3(4),NUM6(4),NUM12(4),NUM18(4),NUM24(4),NUM30(4),
     3      NUM36(4)/HEA10/QSCH( 42,3)/HEA1/QQQ( 19)/IND3/NTYP( 42)
     4      /GASD2/RAPPAI( 42,3)
C
      IF(NSEL.EQ.4)GOTO 100
      CALL HEATR(NROMA)
C
      NAN=1
      NBN=-NROMA
      NN=1-NROMA
      DO 15 NS=1,NSTOT
      NUM=NUMS(NS)
      NRO=NROW(NS)
      IF(NS.GT.NSTR)GOTO 12
C
C     CENTRAL CHANNELS AND SUBCHANNELS
      IF(NUM.GT.NUM6(NRO))GOTO 1
      NAM=NUM
      N1=0
      N2=0
      GOTO 6
    1 IF(NUM.GT.NUM12(NRO))GOTO 2
      NAM=NUM-NUM6(NRO)
      N1=NRO
      N2=N1-1
      GOTO 6
    2 IF(NUM.GT.NUM18(NRO))GOTO 3
      NAM=NUM-NUM12(NRO)
      N1=2*NRO
      N2=N1-2
      GOTO 6
    3 IF(NUM.GT.NUM24(NRO))GOTO 4
      NAM=NUM-NUM18(NRO)
      N1=3*NRO
      N2=N1-3
      GOTO 6
    4 IF(NUM.GT.NUM30(NRO))GOTO 5
      NAM=NUM-NUM24(NRO)
      N1=4*NRO
      N2=N1-4
      GOTO 6
    5 NAM=NUM-NUM30(NRO)
      N1=5*NRO
      N2=N1-5
    6 IF(NAM.EQ. NAM/2*2 )GOTO 8
      NUR=(NAM+1)/2+N1
      Q1=Q(NRO,NUR)
      JPIN(NS,1)=IDPIN(NRO,NUR)
```

```
      IF(NUR.EQ.6*NRO) NUR=0
      Q2=Q(NRO,NUR+1)
      JPIN(NS,3)=IDPIN(NRO,NUR+1)
      IF(NRO.EQ.1)GOTO 7
      NUR=(NAM+1)/2+N2
      IF(NUR.EQ. 6*NRO-5 ) NUR=1
      Q3=Q(NRO-1,NUR)
      JPIN(NS,2)=IDPIN(NRO-1,NUR)
      GOTO 9
    7 Q3=Q0
      JPIN(NS,2)=1
      GOTO 9
    8 NUR=NAM/2+N2
      Q1=Q(NRO-1,NUR)
      JPIN(NS,1)=IDPIN(NRO-1,NUR)
      IF(NUR.EQ.6*NRO-6) NUR=0
      Q2=Q(NRO-1,NUR+1)
      JPIN(NS,3)=IDPIN(NRO-1,NUR+1)
      NUR=(NAM+2)/2+N1
      Q3=Q(NRO,NUR)
      JPIN(NS,2)=IDPIN(NRO,NUR)
    9 CONTINUE
      QQ(NS,1)=Q1
      QQ(NS,2)=Q3
      QQ(NS,3)=Q2
      IF(NSEL.EQ.3 .AND. NUM.EQ.NRO)GOTO 10
      NPIN(NS)=3
      GOTO 11
   10 Q2=0.
      Q3=Q3/2.
      NPIN(NS)=2
   11 QT(NS)=(Q1+Q2+Q3)/6.
      QSCH(NS,1)=Q1/6.
      QSCH(NS,2)=Q3/6.
      QSCH(NS,3)=Q2/6.
      GOTO 15
C
C
   12 IF(NUM.LT.NAN)GOTO 14
C
C     CORNER CHANNELS
      NN=NN+NROMA
      NAN=NAN+NRO
      NBN=NBN+NRO
      NPIN(NS)=1
      QQ(NS,1)=Q(NROMA,NN)
      JPIN(NS,1)=IDPIN(NROMA,NN)
      IF(NSEL.EQ.3)GOTO 13
      IF((NSEL.EQ.2 .AND. NUM.EQ.1).OR.(NSEL.EQ.2 .AND. NUM.EQ.NUM18(NRO
     *)))GOTO 13
      QT(NS)=Q(NROMA,NN)/6.
      GOTO 29
   13 QT(NS)=Q(NROMA,NN)/12.
   29 QSCH(NS,1)=QT(NS)
      GOTO 15
C
C     WALL CHANNELS AND SUBCHANNELS
   14 NUR=NUM-NBN+NN-1
      Q1=Q(NROMA,NUR)
      JPIN(NS,1)=IDPIN(NROMA,NUR)
      IF(NS.EQ.NSTOT .AND. NSEL.EQ.1) NUR=0
      Q2=Q(NROMA,NUR+1)
      JPIN(NS,2)=IDPIN(NROMA,NUR+1)
      QQ(NS,1)=Q1
      QQ(NS,2)=Q2
```

```
      IF(NSEL.EQ.3 .AND. NUM.EQ.(NRO/2+1) .AND. NRO.EQ.NRO/2*2)GOTO 30
      NPIN(NS)=2
      GOTO 31
   30 Q2=0.
      NPIN(NS)=1
   31 CONTINUE
      QT(NS)=(Q1+Q2)/4.
      QSCH(NS,1)=Q1*0.25
      QSCH(NS,2)=Q2*0.25
   15 CONTINUE
C     ADDED AT GA(NSEL=4)
      GOTO 104
  100 CONTINUE
      DO 103 NS=1,NSTOT
      NP=NPIN(NS)
      QT(NS)=0.
      DO 102 M=1,NP
      JPINNM=JPIN(NS,M)
      QQ(NS,M)=QQQ(JPINNM)
      IF(NTYP(NS).EQ.2)GOTO 101
      QSCH(NS,M)=QQ(NS,M)/6.*RAPPAI(NS,M)
      GOTO 102
  101 QSCH(NS,M)=QQ(NS,M)*0.25
  102 QT(NS)=QT(NS)+QSCH(NS,M)
  103 CONTINUE
  104 CONTINUE
 9998 CONTINUE
      IF(IPA.NE.IPA/2*2)RETURN
C
C
      WRITE(6,16)
   16 FORMAT(////5X,'RESULTS OF HEATI'////8X,'CHANNEL',3(21X,'ROD',2X)/)
      DO 19 NS=1,NSTOT
      NP=NPIN(NS)
      WRITE(6,18)NS,(M,NS,M,JPIN(NS,M),M=1,NP)
   18 FORMAT(2X,I10,3(3X,I1,') JPIN(',I5,',',I2,')=',I5))
   19 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE HEATR(NROMA)
C-----------------------------------------------------------------------
C     HEATR PROVIDES INDICES TO THE ROD HEAT FLUXES ( Q(NRO,NUM) ) AND
C     IDENTIFIES THE PINS BY MEANS OF THE MATRIX IDPIN
C
C               EXISTS ONLY IN THE VERSION FOR HEXAGONAL BUNDLES
C     ..................................................................
      COMMON/HEA1/Q( 19)/HEA2/QQ(3, 18),QQ0/HEA7/IDPIN(3, 18)
      I=1
      QQ0=Q(1)
      DO 2 NRO=1,NROMA
      NR36=6*NRO
      DO 1 NUM=1,NR36
      I=I+1
      IDPIN(NRO,NUM)=I
    1 QQ(NRO,NUM)=Q(I)
    2 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INDEX(NSEL,NROMA,NSTR,NSTOT,NRO)
C-----------------------------------------------------------------------
C     INDEX PROVIDES INDICES TO THE CHANNELS
C
C                  VERSION FOR HEXAGONAL BUNDLES
C     ..................................................................
      COMMON/IND1/NROW( 42),NUMS( 42)/IND2/NOT(4,30)/IND3/NTYP( 42)
     1/IND4/NUM3(4),NUM6(4),NUM12(4),NUM18(4),NUM24(4),NUM30(4),NUM36(4)
      IF(NSEL.EQ.4)GOTO 100
      NS=1
      DO  6 NRO=1,NROMA
      NUM3(NRO)=NRO
      NUM6(NRO)=2*NRO-1
      NUM12(NRO)=2*NUM6(NRO)
      NUM18(NRO)=3*NUM6(NRO)
      NUM24(NRO)=4*NUM6(NRO)
      NUM30(NRO)=5*NUM6(NRO)
      NUM36(NRO)=6*NUM6(NRO)
      IF(NSEL-2)1,2,3
    1 NUMSP=NUM36(NRO)
      GOTO4
    2 NUMSP=NUM18(NRO)
      GOTO4
    3 NUMSP=NUM3(NRO)
    4 CONTINUE
      DO 5 NUM=1,NUMSP
      NUMS(NS)=NUM
      NROW(NS)=NRO
      NOT(NRO,NUM)=NS
      NTYP(NS)=1
    5 NS=NS+1
    6 CONTINUE
      NSTR=NS-1
      NRO=NROMA+1
      NUM3(NRO)=NRO/2+1
      NUM6(NRO)=NRO+1
      NUM12(NRO)=NUM6(NRO)+NRO
      NUM18(NRO)=NUM12(NRO)+NRO
      NUM24(NRO)=NUM18(NRO)+NRO
      NUM30(NRO)=NUM24(NRO)+NRO
      NUM36(NRO)=NUM30(NRO)+NROMA
      IF(NSEL-2)7,8,9
    7 NUMSP=NUM36(NRO)
      GOTO 10
    8 NUMSP=NUM18(NRO)
      GOTO 10
    9 NUMSP=NUM3(NRO)
   10 NAN=1
      DO 13 NUM=1,NUMSP
      IF(NUM.EQ.NAN)GOTO 11
      NTYP(NS)=2
      GOTO 12
   11 NTYP(NS)=3
      NAN=NAN+NRO
   12 NUMS(NS)=NUM
      NROW(NS)=NRO
      NOT(NRO,NUM)=NS
   13 NS=NS+1
      NSTOT=NS-1
C     ADDED AT GA(NSEL=4)
  100 IF(NSEL.EQ.4)NRO=NROMA+1
```

```
      WRITE(6,14)NRO,NSEL,NSTR,NSTOT
   14 FORMAT(  //4X,'RESULTS OF INDEX'//5X,'NROWS=',I2,5X,'TYPE OF SECTI
     *ON=',I1,5X,'NR. OF CENTRAL CHANNELS=',I4,5X,'TOTAL NUMBER OF CHANN
     *ELS=',I4//)
      CALL CONNIJ(NSTR,NSTOT,NROMA,NSEL)
      RETURN
      END
```

```
      SUBROUTINE INGE(NROMA,NSEL,NSTR,NSTOT,C,A,D,ATC,ATW,ATA,PIG,PCORR,
     *CTU1,CTU2,DETC,DETW,EM1)
C-----------------------------------------------------------------------
C     INGE EVALUATES THE TURBULENT MIXING CONSTANTS CTURB(I,J) FOR THE
C     THE CHANNEL EXCHANGES AND CTURB1(K) (K=1,2) FOR THE SUBCHANNEL
C     EXCHANGES. FURTHERMORE INGE EVALUATES THE CONSTANTS CCOND(I,J)
C     AND CCOND1(K) FOR THE ENTHALPY EXCHANGE DUE TO CONDUCTION IN GAS
C
C                VERSION FOR HEXAGONAL BUNDLES
C     ..................................................................
      COMMON /IND3/NTYP( 42)/IJ1/NER( 42),NIS( 42,3)
      COMMON/IND1/NROW( 42),NUMS( 42)
     1       /TUR1/CTURB( 42,3)/GEN5/DE( 42)/GEO0/ACH(3)/TUR2/CTURB1(2)
     2       /COND0/FCOND/COND1/CCOND( 42,3)/COND2/CCOND1(2)
     3       /GEN2/AREA( 42)
      REAL NGAPS( 42)
      DIMENSION SUM( 42)
      WRITE(6,101)
  101 FORMAT(///5X,'MIXING COEFFICIENTS (WITHOUT PCORR CORRECTION:'/)
      SQ3=SQRT(3.)
      R=D*0.5
      A2=A**2
      A3=A*A2
      R2=R**2
      R3=R*R2
      APIN=PIG*R2
      EM2=C*0.5-EM1
      ZWC=EM2/SQ3
      ATW3=EM2*ZWC
      GAP1=C-D
      GAP2=GAP1*0.5
      GAP3=A-R
      YBC=C*0.5/SQ3
      YBW3=A-ZWC/3.
      XBWS3=C*0.5-EM2/3.
      YBW=(A**2*C*0.5-2./3.*R3-YBW3*ATW3)/ATW
      XBWS=2.*(A*C**2*0.125-R3/3.-XBWS3*ATW3*0.5)/ATW
      XBA=(5./36.*A3-(A/SQ3-R/PIG)*APIN/6.)/(A2/SQ3-APIN/6.)
      YBA=XBA*SQ3
      DELTA1=2.*YBC
      DELTA2=YBC+YBW
      DELTA3=C
      DELTA4=SQRT((A-YBW-YBA)**2+(C*0.5+A/SQ3-XBA)**2)
      RA1=1.+APIN/(2.*ATC)
      RA2=1.+APIN/(ATC+ATW)
      RA3=1.+APIN/(2.*ATW)
      RA4=1.+APIN*2./(3.*(ATW+ATA))
      ALFAW=ATAN(YBW*2./C)
      AP1=YBC*C*0.5-APIN/6.
      AP2=YBW*C*0.5-ALFAW*R2
      AP3=(ATW-AP2)*0.5
      AP4=A2*0.5/SQ3-YBA*XBA*0.5-APIN/12.
      AS1=GAP1*YBC
      AS2=GAP1*YBW
      AS3=C*0.5*GAP3
      AS4=(A/SQ3-XBA)*GAP3
      R1A1=AS1/AP1
      R1A2=AS2/AP2
      R1A3=AS3/AP3
      R1A4=AS4/AP4
      DO 10 I=1,NSTOT
```

```
      ITYP=NTYP(I)
      GOTO (1,2,4),ITYP
    1 SUM(I)=3.*R1A1
      GOTO 3
    2 SUM(I)=R1A2+2.*R1A3
    3 NGAPS(I)=3.
      GOTO 10
    4 SUM(I)=2.*R1A4
      NGAPS(I)=2.
   10 CONTINUE
      DO 24 I=1,NSTOT
      NTYPI=NTYP(I)
      AREAI=ACH(NTYPI)
      RAPPI=AREA(I)/AREAI
      NI=NER(I)
      DO 23 M=1,NI
      J=NIS(I,M)
      NTYPJ=NTYP(J)
      AREAJ=ACH(NTYPJ)
      RAPPJ=AREA(J)/AREAJ
      RAPGAP=1.
      IF(ABS(RAPPI-1.).GT.0.1 .AND. ABS(RAPPJ-1.).GT.0.1)RAPGAP=0.5
      IF(I.GT.NSTR)GOTO 16
      IF(NTYP(J).EQ.2)GOTO 15
      DELTA     =DELTA1
      RAPPA=RA1
      GOTO 17
   15 DELTA     =DELTA2
      RAPPA=RA2
   17 GAP=GAP1*RAPGAP
      GOTO 22
   16 IF(NTYP(I).EQ.3)GOTO 20
      IF(NTYP(J)-2)15,19,20
   19 DELTA     =DELTA3
      RAPPA=RA3
      GOTO 21
   20 DELTA     =DELTA4
      RAPPA=RA4
   21 GAP     =GAP3
   22 CONTINUE
      DEIJ=(AREAI+AREAJ)/(AREAI/DE(I)+AREAJ/DE(J))
      YH  =    1.14*SQRT((NGAPS(I)+NGAPS(J))/(SUM(I)+SUM(J)))*RAPPA**2
      WRITE(6,100)M,I,J,YH
  100 FORMAT(3(5X,I5,') YH(',I3,',',I2,')=',E15.7))
      CTURB( I,M)=YH*GAP/DELTA*DEIJ*0.05*PCORR
      CCOND(I,M)=GAP/DELTA*FCOND*0.5
   23 CONTINUE
   24 CONTINUE
      WRITE(6,102)
  102 FORMAT(///)
      DELSC1=C-(7.*C**3/48.-R3)/(0.25*C**2-PIG*R2*SQ3/6.)
      DELSC2=C-2.*XBWS
      CTURB1(1)=CTU1*0.05*DETC*YBC/DELSC1
      CTURB1(2)=CTU2*0.05*DETW*(A-ZWC)/DELSC2
      CCOND1(1)=YBC/DELSC1*FCOND*0.5
      CCOND1(2)=(A-ZWC)/DELSC2*FCOND*0.5
      RETURN
      END
```

```
      SUBROUTINE INLCON(NSTOT,MFLOW,ATOT,TE,IREAD1,NSTR)
C-----------------------------------------------------------------------
C     SUBROUTINE INLCON  FIXES THE INLET CONDITIONS FOR MASS FLOW RATES
C     AND BULK TEMPERATURES OF THE CHANNELS AND THE SUBCHANNELS AND FOR
C     THE BULK TEMPERATURES OF THE TWO PORTIONS OF THE WALL SUBCHANNELS
C
      REAL MFLOW,MI,MSCH1,MSCWC1,MSCH
      COMMON/IND3/NTYP( 42)/SUB2/TSCH( 42,3),MSCH( 42,3)/SUB22/TW( 42,3)
     1      /GEN2/A( 42)/GEN3/MI( 42)/GEN4/TEMP( 42)
     2      /HEA6/NPIN( 42),JPIN( 42,3)
     3      /SUB1/ASCH( 42,3)/SUB6/TSCH1( 42,3)/SUB8/MSCH1( 42,3)
     4      /WCSE2/MSCWC1( 18,2,2)/WCSE5/TSCWC1( 18,2,2)
      IF(IREAD1.EQ.2)GOTO 3
C     ..................................................................
C     IREAD1=1 MEANS UNIFORM DISTRIBUTIONS
C
      DO 2 NS=1,NSTOT
      MI(NS)=MFLOW*A(NS)/ATOT
      TEMP(NS)=TE
      NP=NPIN(NS)
      DO 1 M=1,NP
      MSCH1(NS,M)=MFLOW*ASCH(NS,M)/ATOT
      TSCH1(NS,M)=TE
      IF(NTYP(NS).NE.2)GOTO 1
      DO 6 JWC=1,2
    6 TSCWC1(NS-NSTR,M,JWC)=TE
    1 CONTINUE
    2 CONTINUE
      GOTO 1000
C     ..................................................................
C     IREAD1=2 MEANS NON-UNIFORM DISTRIBUTIONS
C
    3 CONTINUE
      READ(5,4)(MI(NS),TEMP(NS),NS=1,NSTOT)
      DO 5 NS= 1,NSTOT
      NSW=NS-NSTR
      NP=NPIN(NS)
      READ(5,4)(MSCH1(NS,M),TSCH1(NS,M),M=1,NP)
      IF(NTYP(NS).EQ.2)READ(5,4)((MSCWC1(NSW,M,JWC),TSCWC1(NSW,M,JWC),
     *  JWC=1,2),M=1,2)
    4 FORMAT(8F10.5)
    5 CONTINUE
 1000 CONTINUE
      DO 1001 NS=1,NSTOT
      NP=NPIN(NS)
      DO 1001 M=1,NP
      MSCH(NS,M)=MSCH1(NS,M)
      TSCH(NS,M)=TSCH1(NS,M)
      TW(NS,M)=TSCH(NS,M)
 1001 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INQUA(NSEL,NSTOT,NROMA,ATC,ATW,ATA,DETC,DETW,DETA)
C-----------------------------------------------------------------------
C     INQUA PROVIDES INDICES TO CHANNEL FLOW AREAS AND EQUIVALENT
C     DIAMETERS AND TO SUBCHANNEL  FLOW AREAS
C
C                VERSION FOR THE EXAGONAL BUNDLES
C     ..................................................................
      COMMON/IND1/NROW( 42),NUMS( 42)/IND3/NTYP( 42)/SUB1/ASCH( 42,3)
     1      /GEN2/A( 42)/GEN5/DE( 42)
     2      /HEA6/NPIN( 42),JPIN( 42,3)/GASD2/RAPPAI( 42,3)
      II=0
      KK=0
      DO 10 NS=1,NSTOT
      A(NS)=0.
      NP=NPIN(NS)
      IF(NTYP(NS)-2)1,3,6
C***** CENTRAL CHANNELS AND SUBCHANNELS*********************************
    1 DE(NS)=DETC
      IF(NSEL.EQ.4)GOTO 100
      ASCH(NS,1)=ATC/3.
      IF((NSEL.EQ.3).AND.(NROW(NS).EQ.NUMS(NS)))GOTO 2
      A(NS)=ATC
      ASCH(NS,2)=ASCH(NS,1)
      ASCH(NS,3)=ASCH(NS,1)
      GOTO 10
    2 CONTINUE
      A(NS)=ATC/2.
      ASCH(NS,2)=ATC/6.
      GOTO 10
C     ADDED AT GA (NSEL=4)
  100 CONTINUE
      DO 101 M=1,NP
      ASCH(NS,M)=ATC/3.*RAPPAI(NS,M)
  101 A(NS)=A(NS)+ASCH(NS,M)
      GOTO 10
C***** WALL    CHANNELS AND SUBCHANNELS*********************************
    3 DE(NS)=DETW
C     MODIFIED AT GA
      DO 4 M=1,NP
      ASCH(NS,M)=ATW*0.5
    4 A(NS)=A(NS)+ASCH(NS,M)
      GOTO 10
C***** CORNER  CHANNELS AND SUBCHANNELS*********************************
    6 DE(NS)=DETA
      IF(NSEL.EQ.4)GOTO 5
      IF(NSEL.EQ.1)GOTO 7
      IF(NSEL.EQ.3)GOTO 9
      IF(II.EQ.0 .OR. KK.EQ.2) GOTO 8
      KK=KK+1
    7 CONTINUE
      A(NS)=ATA
      ASCH(NS,1)=A(NS)
      GOTO 10
    8 II=1
    9 CONTINUE
      A(NS)=ATA /2.
      ASCH(NS,1)=A(NS)
C     ADDED AT GA (NSEL=4)
      GOTO 10
    5 ASCH(NS,1)=ATA*RAPPAI(NS,1)
      A(NS)=ASCH(NS,1)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE JELLA (JL)
C ---------------------------------------------------------------------
C     PRINT THERMAL DATA
C
      COMMON /SUB22/  TW( 42,3)
      COMMON /HEA6 /  NPIN( 42),JPIN( 42,3)
      COMMON /GASD1/  NSTOT
      COMMON /MART5/  NSTR
      COMMON /SC03C/  NRODS
      COMMON /SC99C/  TD( 42,3)
      COMMON /SC14C/  TBOLD( 42,3)
      COMMON /SC26C/  TBR( 42,3),TBS( 18,2)
      COMMON /SUB2 /  TSCH( 42,3),MSCH( 42,3)
      COMMON /SC02C/  QJ( 19, 42)
      COMMON /MART /  ITCORR
      COMMON /SC06L/  SHQ( 18,2)
      COMMON /SC21C/  SHQC( 18,2)
      COMMON /SC15C/  ALFA( 42,3)
      COMMON /SC17C/  SALFA( 18,2)
      COMMON /SC08R/  QPR( 42,3)
      COMMON /SC09R/  QSR( 18,2)
      COMMON /HEA5 /  QQ( 42,3)
      COMMON /QPAR1/  QDEV
      COMMON /SC02R/  P,D,Z,ZWC,H,LENGTH
      COMMON /SC13C/  GEO1( 42,3)
      COMMON /SHROUD/ TLINER( 18,2)
      COMMON /SC08C/  TLD( 18,2)
      COMMON /SC10C/  ANU( 42,3)
      COMMON /SC16C/  SNU( 18,2)
      COMMON /SC18C/  RE,RI,ALFW,ALFC
      COMMON /DAT/    PIG
      COMMON /IND3/   NTYP( 42)
      COMMON /LAMIN5/ RTIP(7)
      COMMON /INPAR/  IPA
      COMMON /CEV04/  LAMOP3
      COMMON /SC33C/  TWINF( 42,3)
      COMMON /SC01Z/  YH( 42,3)
C
      REAL LENGTH
C
      DU=D
      IF(LAMOP3.EQ.1) DU=RTIP(IPA)*2.
C
      WRITE(6,50) ITCORR
      WRITE(6,100)
C
      WRITE(6,110)
C
  200 DO 1010 JP=1,NRODS
C
      DO 1000 NS=1,NSTOT
              NSW=NS-NSTR
              NP=NPIN(NS)
              DO 500 M=1,NP
                     JJ=JPIN(NS,M)
                     IF(JJ.NE.JP) GO TO 500
                     AREA=PIG*DU*H/GEO1(NS,M)
                     QUNIF=QQ(NS,M)*QDEV/LENGTH/GEO1(NS,M)*4.186*H
                     QTOT=(QUNIF+QJ(JP,NS)*4.186)/AREA
                     QRAD=QPR(NS,M)*4.186
                     WALFA=ALFA(NS,M)*4.186
```

```
                 DELT4=    TW(NS,M)-TD(NS,M)
                 DELT5=    TSCH(NS,M)-TBOLD(NS,M)
                 IF ( JL.LT. 0) GO TO 400
                 IF ( JL.EQ. JP)GO TO 400
                 GO TO 500
  400            WRITE(6,9000) NS,JP,QUNIF,QRAD,QTOT,
     >                 ANU(NS,M),YH(NS,M),WALFA,TSCH(NS,M),
     >                 TBR(NS,M),TWINF(NS,M),TW(NS,M)
  500            CONTINUE
 1000        CONTINUE
 1010        CONTINUE
C
      IF (JL .GT. 0) GO TO 9999
C
      WRITE(6,9200)
      WRITE(6,100)
      WRITE(6,110)
C
      NSTR1=NSTR+1
      DO 2000 NS=NSTR1,NSTOT
              NSW=NS-NSTR
              NP=NPIN(NS)
              DO 1500 M=1,NP
                     ALF=ALFW
                     IF(NTYP(NS).EQ.3) ALF=ALFC/GEO1(NS,M)*6.0
                     AREA=RI*ALF*H
                     JP=JPIN(NS,M)
                     QLINER=SHQ(NSW,M)*4.1860
                     QTOT=(SHQ(NSW,M)+SHQC(NSW,M))/AREA*4.1860
                     QRAD=QSR(NSW,M)*4.1860
                     WALFA=SALFA(NSW,M)*4.1860
                     ZZ=1.0
                     WRITE(6,9000) NS,JP,QLINER,QRAD,QTOT,
     >                     SNU(NSW,M),ZZ,WALFA,TSCH(NS,M),TBS(NSW,M),
     >                     TLINER(NSW,M),TLINER(NSW,M)
 1500                CONTINUE
 2000         CONTINUE
C
      WRITE(6,9200)
C
 9999 RETURN
   50 FORMAT( ' JELLA, ITCORR = ',I3,//)
  100 FORMAT(  ' CHA. ROD     QGEN ',
     >         T26,' QRAD',
     >         T37,' QFLUX',
     >         T50,'  NU ',
     >         T62,'  YH ',
     >         T75,'ALFA ',
     >         T87,'TBULK',
     >         T98,' TB R ',
     >        T110,' TWINF',
     >        T122,'TWALL ')
  110 FORMAT(  '                    W  ',
     >         T26,'  W   ',
     >         T37,'W/CM**2',
     >         T50,'      ',
     >         T61,'      ',
     >         T72,'W/CM**2 C',
     >         T88,' C   ',
     >         T99,' C    ',
     >        T110,'   C  ',
     >        T122,'  C   ',/)
 9000 FORMAT(' ',I3,2X,I3,10(2X,E10.4))
 9200 FORMAT( //)
      END
```

```
      SUBROUTINE JELLB (JL,PBT)
C ---------------------------------------------------------------------
C     PRINT SUBCHANNEL DATA.
C     CAUTION: JELLB CAN BE CALLED ONLY BY THE MAIN PROGRAM.
C              (BECAUSE OF PBT)
C              THE MEANING OF TSCH1,MSCH1 VARIES TROUGH THE PROGRAM.
C
      COMMON /HEA6 /  NPIN( 42),JPIN( 42,3)
      COMMON /GASD1/  NSTOT
      COMMON /MART5/  NSTR
      COMMON /SUB1 /  ASCH( 42,3)
      COMMON /SUB2 /  TSCH( 42,3),MSCH( 42,3)
      COMMON /SUB6 /  TSCH1( 42,3)
      COMMON /SUB8 /  MSCH1( 42,3)
      COMMON /SUB5 /  LAMSCH( 42,3)
      COMMON /MOB8 /  DPNS( 42)
      COMMON /GEN5 /  DE( 42)
      COMMON /LAMIN4/ F2ATIP( 42,3), F2DTIP( 42,3)
      COMMON /SC02C/  QJ( 19, 42)
      COMMON /HEA5 /  QQ( 42,3)
      COMMON /QPAR1/  QDEV
      COMMON /SC13C/  GEO1( 42,3)
      COMMON /SC02R/  P,D,Z,ZWC,H,LENGTH
C
      REAL   LAMSCH,MSCH,MSCH1,KAPPA,LENGTH
C
C
      WRITE(6,1001)
      WRITE(6,1002)
      WRITE(6,1003)
      DO 1000 NS=1,NSTOT
             NSW=NS-NSTR
             NP=NPIN(NS)
             DO 500 M=1,NP
                   JP=JPIN(NS,M)
                   QUNIF=QQ(NS,M)*QDEV/LENGTH/GEO1(NS,M)*4.186*H
                   QTOT=(QUNIF+QJ(JP,NS)*4.186)
                   CPX=CP(PBT,TSCH(NS,M))*4.186
                   ROX=RHO(PBT,TSCH(NS,M))
                   ETX=ETA(PBT,TSCH(NS,M))
                   XKA=KAPPA(PBT,TSCH(NS,M))*4.186
                   M2=2.*MSCH(NS,M)-MSCH1(NS,M)
CCCCCC             T=(2.*MSCH(NS,M)*TSCH(NS,M)-MSCH1(NS,M)*TSCH1(NS,M))/
CCCC >                MSCH1(NS,M)
                   RE=MSCH(NS,M)*DE(NS)*F2DTIP(NS,M)/
     >                (ASCH(NS,M)*F2ATIP(NS,M)*ETA(PBT,TSCH(NS,M)))
C                  DT=T-TSCH(NS,M)
                   DT=(TSCH(NS,M)-TSCH1(NS,M))*2.0
                   PD=DPNS(NS)*0.980667
                   IF ( JL.LT. 0) GO TO 400
                   IF ( JL.EQ. JP)GO TO 400
                   GO TO 500
  400              WRITE(6,1004) NS,JP,QTOT,MSCH(NS,M),CPX,DT,
     >                   LAMSCH(NS,M),PD,RE,ROX,XKA,ETX
  500              CONTINUE
 1000        CONTINUE
      WRITE(6,1005)
C
 9999 RETURN
 1001 FORMAT(1H1,5X,' SUBR. JELLB')
```

```
1002 FORMAT(    'CHA. ROD      QTRA  ',
    >         T26,' MASS',
    >         T37,'   CP ',
    >         T51,'T2-T1',
    >         T62,'LAMDA',
    >         T75,' DP  ',
    >         T87,' REB ',
    >         T98,'DENSITY',
    >        T109,'TH.COND.',
    >        T122,' ETA  ')
1003 FORMAT(  '                    W   ',
    >         T26,'G/SEC ',
    >         T38,'J/G C  ',
    >         T51,'  C  ',
    >         T60,'         ',
    >         T75,' BAR ',
    >         T88,'     ',
    >         T99,'G/CM3  ',
    >        T110,'W/CM C',
    >        T122,'G/CM S',/)
1004 FORMAT(' ',I3,2X,I3,10(2X,E10.4))
1005 FORMAT( //)
     END
```

```
      SUBROUTINE JELLC (JL,TE,TE1,BIOT)
C ----------------------------------------------------------------------
C     PRINT SUBCHANNEL DATA.
C     CAUTION: JELLC CAN BE CALLED ONLY BY THE MAIN PROGRAM.
C              (BECAUSE OF PBT,TE,TE1,BIOT)
C
      DIMENSION BIOT( 42,3)
C
      COMMON /HEA6 /  NPIN( 42),JPIN( 42,3)
      COMMON /GASD1/  NSTOT
      COMMON /MART5/  NSTR
      COMMON /SUB2 /  TSCH( 42,3),MSCH( 42,3)
      COMMON /SUB22/  TW( 42,3)
      COMMON /SUB23/  HPLUSB( 42,3),HPLUSW( 42,3),QPLUS( 42,3),
     >                PRB( 42,3),YODH( 42,3)
      COMMON /PARTB/  TEMPB( 42,3),XMASSB( 42,3),YDH(42,3)
      COMMON /SC32C/  GHPIU( 42,3)
      COMMON /DAT1 /  O0,O1,O2,O3,O4,O5,O16,O17,O18,O19
      COMMON /DAT2 /  O6,O7,O8,O9,O10,O11,O12,O13,O14,O15
      COMMON /DAT6 /  IRHPL
C
C
      REAL MSCH
C
C
      WRITE(6,1001)
      WRITE(6,1002)
      DO 1000 NS=1,NSTOT
             NSW=NS-NSTR
             NP=NPIN(NS)
             DO 500 M=1,NP
                    GOTO(100,200),IRHPL
  100               HPLUS=HPLUSW(NS,M)
                    GO TO 300
  200               HPLUS=HPLUSB(NS,M)
  300               CONTINUE
                    JP=JPIN(NS,M)
                    RHP=RHPLUS(HPLUSB(NS,M),TW(NS,M),TE1,QPLUS(NS,M),
     >                  HPLUSW(NS,M),TEMPB(NS,M),YODH(NS,M))
                    RO1=(O6+O7/HPLUS**O8)**O9
                    RSM=2.5+5.5*ALOG(HPLUS)
                    TWE=(TW(NS,M)+273.16)/(TE+273.16)
                    TWB=(TW(NS,M)+273.16)/(TSCH(NS,M)+273.16)
                    GO1=GHPIU(NS,M)/(PRB(NS,M)**O4*TWB**O5)*
     >                  (O16*YDH(NS,M))**O17
                    IF ( JL.LT. 0) GO TO 400
                    IF ( JL.EQ. JP)GO TO 400
                    GO TO 500
  400               WRITE(6,1004) NS,JP,HPLUSB(NS,M),HPLUSW(NS,M),
     >                    RSM,RO1,RHP,GO1,GHPIU(NS,M),TWB,TWE,BIOT(NS,M)
  500               CONTINUE
 1000        CONTINUE
      WRITE(6,1005)
C
 9999 RETURN
 1001 FORMAT(1H1,5X,' SUBR. JELLC',///)
 1002 FORMAT(  ' CHA. ROD      H+B   ',
     >          T26,' H+W ',
     >          T37,'  RSM ',
     >          T50,' RO1 ',
```

```
     >         T62,' RH+ ',
     >         T75,' G01 ',
     >         T87,' GH+ ',
     >         T98,' TW/TB ',
     >        T109,' TW/TE    ',
     >        T122,' BIOT ')
 1004 FORMAT(' ',I3,2X,I3,10(2X,E10.4))
 1005 FORMAT( //)
      END
```

```
      SUBROUTINE JZURU
C ---------------------------------------------------------------------
C     DETERMINES THE ARRAY JZUR(J,NS)   (INDEX OF THE SUBCHANNEL OF CHAN
C     NEL NS ADJACENT TO PIN J).
C
      COMMON /GASD1/ NSTOT
      COMMON /SC03C/ NRODS
      COMMON /SC05C/ JZUR( 19, 42)
      COMMON /HEA6 / NPIN( 42),JPIN( 42,3)
C
      DO 300 J=1,NRODS
             DO 200 NS=1,NSTOT
                    NP=NPIN(NS)
                    DO 100 M=1,NP
                           IF(JPIN(NS,M).EQ.J) JZUR(J,NS)=M
  100                      CONTINUE
  200               CONTINUE
  300        CONTINUE
      RETURN
      END
```

```
      SUBROUTINE KAPCOR(NSTOT,NSTR)
C     -----------------------------------------------------------------
C     KAPCOR COMPUTES THE KAPPA VALUES FOR THE LAMINAR CALCULATIONS
C     (IF IKAPPA=1, OTHERWISE SAVES THE VALUES OF BLOCK DATA) AND
C     CORRECTS THE KAPPA VALUES OF THE CORNER AND WALL CHANNELS IF IT IS
C      DESIRED TO HAVE THERE ABOUT THE SAME VALUE OF (LAMBDA/EQ. DIAM.)
C
      COMMON/LAMIN2/FATIP(3),FDTIP(3)/IND3/NTYP( 42)/GEN2/A( 42)/GEN5/
     1      DE( 42)/INPAR/IPA/LAMINK/BKAPPA(7,3)/LAMIN1/AKAPPA( 42)
     2      /MART2/NS1,NS2/WALLCO/WFCO1( 18,2),WFCO( 18,2)
     3      /WAKA1/IKAPPA
      COMMON /WALLKA/ AKAWC(2)
      COMMON /SCO2L/ JLAM
C
      AKAWC(1)=BKAPPA(IPA,2)
      AKAWC(2)=BKAPPA(IPA,2)
      IF(IKAPPA.EQ.1)CALL SELAWA
      DO 5 NS=1,NSTOT
      IF(NS.LE.NSTR)GOTO 3
      DO 4 I=1,2
      WFCO (NS-NSTR,I)=1.
    4 WFCO1(NS-NSTR,I)=1.
    3 ITYP=NTYP(NS)
      AKAPPA(NS)=BKAPPA(IPA,ITYP)
    5 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF (JLAM .EQ. 1) CALL CORKA
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(NS1.EQ.0 .AND. NS2.EQ.0)GOTO 35
      IF(NS1.GT.NSTR .AND. NS2.LE.NSTOT)GOTO 9
      WRITE(6,6)NS1,NS2
    6 FORMAT(1H1,5X,'SUB. KAPCOR',
     >          /5X,'STOP BECAUSE NS1=',I5,' AND NS2=',I5)
      STOP
C
    9 AT=0.
      PP=0.
      DO 10 NS=NS1,NS2
      ITYP=NTYP(NS)
CCCCCCCCCCCCCCCCCCCCCCCCCCC 21.09.1979
C     ATIP=A(NS)*FATIP(NS)
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      ATIP=A(NS)*FATIP(ITYP)
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      PP=PP+ATIP*(DE(NS)*FDTIP(ITYP))**2/BKAPPA(IPA,ITYP)
   10 AT=AT+ATIP
      PP=AT/PP
      DO 20 NS=NS1,NS2
      ITYP=NTYP(NS)
   20 AKAPPA(NS)=(DE(NS)*FDTIP(ITYP))**2*PP
C
   35 CONTINUE
      DO 29 NS=1,NSTOT
      ITYP=NTYP(NS)
   29 WRITE(6,30)NS,AKAPPA(NS),BKAPPA(IPA,ITYP)
   30 FORMAT(    5X,'CHANNEL',I5,' : USED KAPPA=',F10.3,' (INPUT KAPPA='
     *      ,F10.3,')')
      RETURN
      END
```

```
      REAL FUNCTION KAPPA(P,T)
C-----------------------------------------------------------------------
C     KAPPA EVALUATES THE THERMAL CONDUCTIVITY OF THE COOLANT
C     (CAL/CM S K)
C
      COMMON/GASD4/IGAS
      GOTO(10,20,30,40),IGAS
   10 CONTINUE
C     CASE OF HELIUM COOLANT
C
      KAPPA=35.1E-05*((T+273.16)/273.16)**0.66
      RETURN
C
   20 CONTINUE
C     CASE OF CO2 COOLANT
C
      PP=P
      TT=T
      P=PP/1.0333
      T=TT+273.16
      TO=273.16
      TF=TO/T
      IF(P-1.) 4,4,5
    4 ECL = P-1.
      GO TO 6
    5 ECL = (P -1.)**1.25
    6 IF(T-TO-725.) 1,1,2
    1 CA=3.4943E2
      CB=1.6768E5
      CC=2.7331E7
      GO TO 3
    2 CA=4.0476E2
      CB=1.5904E5
      CC=-1.9206E7
    3 CLAMO=(SQRT(T))/(CA+(CB/T)+(CC/(T*T)))
      CLAMF=CLAMO*(1.+2.14E-3*ECL*(TF**2.36))
      KAPPA=CLAMF/360.
      P=PP
      T=TT
      RETURN
C
   30 CONTINUE
C     CASE OF N2 COOLANT.
C
      TT=T
      T=TT+273.16
      XLAO = 1.99E-3*T**.5/(1.+188./T*10.**(-12./T))
      KAPPA = XLAO*(1.+1.74E-3*(P/1.033-1.)*(273.16/T))/360.
      T=TT
C
      RETURN
   40 CONTINUE
      KAPPA=0.
      RETURN
      END
```

```
      REAL FUNCTION KFUEL (TEMP)
C ---------------------------------------------------------------------
C     COMPUTES THE THERMAL CONDUCTIVITY OF THE FUEL. ( CAL/SEC*CM*C)
C
      KFUEL = (16.5 + 0.0132 * TEMP) / 4.186 * 1.E-02
C
      RETURN
      END
```

```
      SUBROUTINE KGAP
C ---------------------------------------------------------------------
C     DETERMINES THE HEAT TRANSFER COEFFICIENT BETWEEN FUEL AND CLAD.
C                                                ( CAL/SEC CM2 C)
C
      COMMON /SC20C/ CGAP
C
      CGAP = 2200.0 / 4.186 * 1.E-04
      RETURN
      END
```

```
      REAL FUNCTION KINF(BIOT)
C----------------------------------------------------------------------
C     KINF EVALUATES THE K INFINITE VALUE
C
      COMMON/BIDAT/BI1,BI2,BI3
      KINF=BI1+BI2*BIOT+BI3*BIOT**2
      RETURN
      END
```

```
      REAL FUNCTION KMET(TW)
C     KMET EVALUATES THE CONDUCTIVITY OF THE PIN CANNING
C----------------------------------------------------------------------
C
      COMMON/DATKM/D1(7),D2(7) /INPAR/IPA
      COMMON /SC06C/ D3(3)
C
      KMET = D3(1) + D3(2)*TW + D3(3)*TW**2
C     KMET=D1(IPA)+D2(IPA)*TW
      RETURN
      END
```

```
      SUBROUTINE LEIST (VDIAM,PIG,H,VDIA1,RINT)
C -----------------------------------------------------------------------
C     ORGANIZES THE CALCULATION OF THE THERMAL CONDUCTION WITHIN THE PIN
C     IN THE CASE OF CONDUCTION WITHIN THE PIN ONLY ( POWER GENERATED IN
C     THE CAN).
C
      DIMENSION A(6,6),B(6),X(6)
C
      COMMON /SC01C/ NCAN( 19),LIPS( 19,10)
      COMMON /SC03C/ NRODS
      COMMON /SC05C/ JZUR( 19, 42)
      COMMON /SC07C/ H1
      COMMON /SC11C/ FALFA( 42,3)
      COMMON /SC12C/ GEO( 42,3)
      COMMON /SC08R/ QPR( 42,3)
      COMMON /HEA5 / QQ( 42,3)
      COMMON /SUB22/ TW( 42,3)
      COMMON /SC26C/ TBR( 42,3),TBS( 18,2)
      COMMON /QPAR1/ QDEV
C
      REAL KMET
C
C     ALFAC PROVIDES FOR EACH CHANNEL THE CONVECTIVE HEAT TRANSFER CO.
      CALL ALFAC(VDIA1,H)
C
      CALL FGEO(VDIAM,RS,S,R,RINT)
C -----------------------------------------------------------------------
C
C     DETERMINES THE COEFFICIENTS FOR THE LINEAR SYSTEM.
C
      DO 1000 J=1,NRODS
              NCA=NCAN(J)
              IF (NCA-1) 1000,900,100
  100         IF (NCA .EQ. 3) GO TO 710
              DO 700 K=1,NCA
                     KP=K-1
                     KD=K+1
                     IF (K .EQ. NCA) KD=1
                     IF (K .EQ. 1  ) KP=NCA
C
                     NS1=LIPS(J,K)
                     NSD=LIPS(J,KD)
                     NSP=LIPS(J,KP)
                     MZ1= JZUR(J,NS1)
                     MZD= JZUR(J,NSD)
                     MZP= JZUR(J,NSP)
C
                     TP=( TW(NS1,MZ1) + TW(NSP,MZP)) * 0.5
                     TD=( TW(NS1,MZ1) + TW(NSD,MZD)) * 0.5
                     FP=KMET(TP)*H*S/(R*(GEO(NS1,MZ1)+GEO(NSP,MZP)))
                     FD=KMET(TD)*H*S/(R*(GEO(NS1,MZ1)+GEO(NSD,MZD)))
C
                     DO 500 L=1,6
                        A(K,L)=0.0
                        IF(L.EQ.K ) A(K,L)=FALFA(NS1,MZ1) + FP + FD
                        IF(L.EQ.KP) A(K,L)= - FP
                        IF(L.EQ.KD) A(K,L)= - FD
                        TBULK=TBR(NS1,MZ1)
                        B(K) = QQ(NS1,MZ1)*QDEV*H1*(GEO(NS1,MZ1)/PIG) +
     >                         TBULK*FALFA(NS1,MZ1) + QPR(NS1,MZ1)
```

```
  500                   CONTINUE
C
  700                CONTINUE
          GO TO 790
C
  710     DO 730 K=1,NCA
                 DO 720 L=1,6
                        A(K,L)=0.0
  720                   CONTINUE
  730            CONTINUE
C
          NS1=LIPS(J,1)
          NS2=LIPS(J,2)
          NS3=LIPS(J,3)
C
          MZ1=JZUR(J,NS1)
          MZ2=JZUR(J,NS2)
          MZ3=JZUR(J,NS3)
C
          T12=( TW(NS1,MZ1) + TW(NS2,MZ2)) * 0.5
          T23=( TW(NS2,MZ2) + TW(NS3,MZ3)) * 0.5
          F12=KMET(T12)*H*S/(R*(GEO(NS1,MZ1)+GEO(NS2,MZ2)))
          F23=KMET(T23)*H*S/(R*(GEO(NS2,MZ2)+GEO(NS3,MZ3)))
C
          A(1,1) = FALFA(NS1,MZ1) + F12
          A(1,2) = - F12
          A(2,1) = - F12
          A(2,2) = FALFA(NS2,MZ2) + F12 + F23
          A(2,3) = - F23
          A(3,2) = - F23
          A(3,3) = FALFA(NS3,MZ3) + F23
C
          B(1) = QQ(NS1,MZ1)*QDEV*H1*(GEO(NS1,MZ1)/PIG) +
     >           TBR(NS1,MZ1)*FALFA(NS1,MZ1) +
     >           QPR(NS1,MZ1)
          B(2) = QQ(NS2,MZ2)*QDEV*H1*(GEO(NS2,MZ2)/PIG) +
     >           TBR(NS2,MZ2)*FALFA(NS2,MZ2) +
     >           QPR(NS2,MZ2)
          B(3) = QQ(NS3,MZ3)*QDEV*H1*(GEO(NS3,MZ3)/PIG) +
     >           TBR(NS3,MZ3)*FALFA(NS3,MZ3) +
     >           QPR(NS3,MZ3)
C
  790     CONTINUE
C ------------------------------------------------------------------
C         LINAL SOLVES THE SYSTEM. THE NEW TEMPERATURES ARE IN THE
C         FIELD X.
C
C         CALL LINAL(A,NCA,6,A,B,1,6,X)
          CALL GAUSS1(A,B,X,NCA,6)
C
C ------------------------------------------------------------------
C         ASSIGNES THE NEW TEMPERATURES TO THE ARRAY TW.
          CALL TNEW(NCA,X,J,6)
C ------------------------------------------------------------------
C         DETERMINES THE HEAT EXCHANGED BY CONDUCTION.
  900     CALL QCOC (NCA,J,X,6)
C
 1000   CONTINUE
C ------------------------------------------------------------------
C
      RETURN
      END
```

```
      SUBROUTINE LINPOW
C-----------------------------------------------------------------------
C     INITIALIZATION OF THE HEAT GENERATED IN THE SHROUD (CAL/SEC)
C
      COMMON /SC06L/ SHQ( 18,2)
      COMMON /GASD1/ NSTOT
      COMMON /QPAR2/ QLINM,QLDEV
      COMMON /SC03L/ IPHUD
      COMMON /SC04L/ TAMB,HUDFAC,TMS
      COMMON /MART5/ NSTR
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /SC02R/ DUM1,DUM2,DUM3,DUM4,H,LENGTH
      COMMON /SUB2/  TSCH( 42,3),MSCH( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /QPAR3/ PERL(3)
      COMMON /SUB1/  ASCH( 42,3)
      COMMON /GEO0/  ACH(3)
      COMMON /SC05L/ PERLT
C
      REAL LENGTH
C
      NSTR1=NSTR+1
      DO 1000 NS=NSTR1,NSTOT
              NSW=NS-NSTR
              ITYP=NTYP(NS)
              NP=NPIN(NS)
              DO 900 M=1,NP
                     SHFLUX=QLINM*QLDEV/LENGTH
                         IF(IPHUD .EQ. 1)
     >               SHFLUX=HUDFAC/PERLT*(TSCH(NS,M)-TAMB)
                     SHQ(NSW,M)=SHFLUX*H*PERL(ITYP)*ASCH(NS,M)/ACH(ITYP)
  900                CONTINUE
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE MATBUI(A,B)
C ----------------------------------------------------------------------
C     BUILDS THE ARRAY OF COEFFICIENTS FOR THE SYSTEM OF EQUATIONS OF
C     RADIANT EXANGE.
C
      DIMENSION A(132, 13),B(132)
C
      COMMON /SC01R/  NSECT,NSECP
      COMMON /SC04R/  VFAC(132, 13)
      COMMON /SC06R/  ISU(132,2)
      COMMON /SC07R/  EPSR,EPSS,SIGMA
      COMMON /SC13R/  NAFF(132)
      COMMON /SC14R/  KAFF(132,13)
      COMMON /MART5/  NSTR
      COMMON /SUB22/  TW( 42,3)
      COMMON /SHROUD/ TLINER( 18,2)
C
      DO 1000 L1=1,NSECT
              NAF1=NAFF(L1)
              NAF11=NAF1+1
              DA=DAREA(L1)
              NS1=ISU(L1,1)
              M1 =ISU(L1,2)
              NW1=NS1-NSTR
              T0 = TW(NS1,M1)
              IF(L1.GT.NSECP) T0= TLINER(NW1,M1)
              T1 =(T0+273.16)**4
              EPS1=EPS(T0,L1)
C
              KAFF(L1,NAF11)=L1
              B(L1)= -SIGMA*T1
              A(L1,NAF11)= -1./EPS1
C
              DO 500 K=1,NAF1
                     L2=KAFF(L1,K)
                     NS2=ISU(L2,1)
                     M2 =ISU(L2,2)
                     NW2=NS2-NSTR
                     T0 = TW(NS2,M2)
                     IF(L2.GT.NSECP) T0= TLINER(NW2,M2)
                     EPS2=EPS(T0,L2)
                     T2 =(T0+273.16)**4
C
                     B(L1)=B(L1)+SIGMA*T2*VFAC(L1,K)/DA
                     IF(L1.EQ.L2) GO TO 100
C
                     A(L1,K)=VFAC(L1,K)/DA*(1.0-EPS2)/EPS2
                     GO TO 200
  100                A(L1,NAF11)= A(L1,NAF11) +
     >                            VFAC(L1,K)/DA*(1.0-EPS2)/EPS2
                     A(L1,K)=0.0
  200                CONTINUE
C
  500                CONTINUE
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE MATBUS(A,B)
C -----------------------------------------------------------------------
C     BUILDS THE ARRAY OF COEFFINCIENTS FOR THE CONDUCTION WITHIN THE
C     SHROUD.
C
      DIMENSION A( 34, 34),B( 34)
C
      COMMON /SC17C/  SALFA( 18,2)
      COMMON /SC18C/  RE,RI,ALFW,ALFC
      COMMON /SC02R/  P,D,Z,ZWC,H,ENG
      COMMON /SC09R/  QSR( 18,2)
      COMMON /SC06L/  SHQ( 18,2)
      COMMON /MART5/  NSTR
      COMMON /GASD1/  NSTOT
      COMMON /SC13C/  GEO1( 42,3)
      COMMON /SC22C/  NTOT
      COMMON /SC24C/  NSEL
      COMMON /SHROUD/ TLINER( 18,2)
      COMMON /HEA6/   NPIN( 42),JPIN( 42,3)
      COMMON /SC26C/  TBR( 42,3),TBS( 18,2)
      COMMON /IND3/   NTYP( 42)
C
      REAL KMET
C
      NSW=NSTOT-NSTR
      RM=(RE+RI)/2.
      S=RE-RI
C
      DO 20 N=1,NTOT
             B(N)=0.0
             DO  10 L=1,NTOT
                    A(N,L)=0.0
   10               CONTINUE
   20        CONTINUE
C
      N=0
      DO 1000 NW=1,NSW
              NS=NW+NSTR
              NP=NPIN(NS)
              DO 900 M=1,NP
                     NSP=0
                     MP=0
                     NSD=0
                     MD=0
                     N=N+1
                     ALF=ALFW
                     IF(NTYP(NS).EQ.3) ALF=ALFC*6.0/GEO1(NS,1)
                     B(N)=SHQ(NW,M)+QSR(NW,M)+
     >                    ALF*H*RI*SALFA(NW,M)*TBS(NW,M)
                     IF(N.EQ.1 .AND. NSEL.NE.1) GO TO 300
                     IF(M.EQ.2) GO TO 100
                     NSP=NS-1
                     IF(N.EQ.1 ) NSP=NSTOT
                     MP=2
                     IF(NTYP(NSP).EQ.3) MP=1
                     GO TO 200
  100                CONTINUE
                     NSP=NS
                     MP=1
  200                CONTINUE
                     NWP=NSP-NSTR
```

```
                 TM=(TLINER(NWP,MP)+TLINER(NW,M))*0.5
                 ALP=ALFW
                 IF(NTYP(NSP).EQ.3) ALP=ALFC*6.0/GEO1(NSP,1)
                 AP=KMET(TM)*S*H/(RM*(ALF+ALP)*0.5)
                 N1=N-1
                 IF(N.EQ.1) N1=NTOT
                 A(N,N1)=-AP
                 GO TO 400
  300            AP=0.0
  400            CONTINUE
C
                 IF(N.EQ.NTOT .AND. NSEL.NE.1) GO TO 700
                 IF(M.EQ.1 .AND. NTYP(NS).EQ.2) GO TO 500
                 NSD=NS+1
                 IF(N.EQ.NTOT) NSD=NSTR+1
                 MD=1
                 GO TO 600
  500            CONTINUE
                 NSD=NS
                 MD=2
  600            CONTINUE
                 NWD=NSD-NSTR
                 TM=(TLINER(NWD,MD)+TLINER(NW,M))*0.5
                 ALD=ALFW
                 IF(NTYP(NSD).EQ.3) ALD=ALFC*6.0/GEO1(NSD,1)
                 AD=KMET(TM)*S*H/(RM*(ALD+ALF)*0.5)
                 ND=N+1
                 IF(N.EQ.NTOT) ND=1
                 A(N,ND)=-AD
                 GO TO 800
  700            CONTINUE
                 AD=0.0
  800            CONTINUE
                 A(N,N)=AD+AP+SALFA(NW,M)*ALF*H*RI
C
  900            CONTINUE
 1000       CONTINUE
C
      IF(N.EQ.NTOT) GO TO 9999
      WRITE(6,2000) NTOT,N
 2000 FORMAT ( '      SUB. MATBUS; NTOT = ',I4,'  N = ',I4,5X,
     >         'CALCULATION STOPS.')
      STOP
C
 9999 RETURN
      END
```

```
      SUBROUTINE MEZZI(NRODS,NSEL,NSTOT)
C -----------------------------------------------------------------------
C     DETERMINES THE ARRAYS GEO1 AND GEO
C
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /SC12C/ GEO( 42,3)
      COMMON /SC13C/ GEO1( 42,3)
      COMMON /SC24C/ NNSEL
C
      PIG=3.141593
      NNSEL=NSEL
C
      DO 200 NS=1,NSTOT
             NP=NPIN(NS)
             DO 100 M=1,NP
                    GEO1(NS,M)=6.
                    IF(NTYP(NS) .EQ. 2) GEO1(NS,M)=4.
  100               CONTINUE
  200        CONTINUE
C
      IF(NSEL .EQ. 1) GO TO 900
      IF(NRODS .NE. 12) GO TO 300
C
      GEO1(2,2)=12.
      GEO1(6,1)=12.
      GO TO 900
  300 IF(NSEL .GT. 2) GO TO 500
      DO 400 NS=1,NSTOT
             IF (NTYP(NS) .EQ. 3) GO TO 460
  400        CONTINUE
      WRITE(6,420)
  420 FORMAT(/5X,'SUB. MEZZI, ERROR IN LOOP 400',/5X,
     >       'CALCULATION STOPS.')
      STOP
  460 GEO1(NS,1)=12.
      GEO1(NSTOT,1)=12.
      GO TO 900
  500 DO 550 NS=1,NSTOT
      IF(NTYP(NS) .EQ. 3) GEO1(NS,1)=12.
  550 CONTINUE
  900 CONTINUE
C
      DO 1000 NS=1,NSTOT
              NP=NPIN(NS)
              DO 950 M=1,NP
                     GEO(NS,M)=PIG/GEO1(NS,M)
  950                CONTINUE
 1000         CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MODFQD(I,NI,NJ,A,EXF)
C-----------------------------------------------------------------------
C     MODFQD COMPUTES THE COEFFICIENTS A OF THE INTEGRAL PROFILES OF
C     POWER
C
      DIMENSION A(7,7)
      DO 10 J=1,NJ
      AJ=J
   10 A(I,J)=A(I,J)/(AJ*EXF**AJ)
      RETURN
      END
```

```
      SUBROUTINE NEWTON(R0,R1,R2)
C     -----------------------------------------------------------------
C     NEWTON FINDS R2 IN THE LAMINAR CALCULATIONS OF THE CENTRAL
C     SUBCHANNELS AND THE CENTRAL PORTIONS OF THE WALL SUBCHANNELS
C     BY MEANS OF THE NEWTON ITERATION METHOD
C
      R2P=2.*R0-R1
      A=-0.5/R0**2
      B=-ALOG(R1)+0.5*(R1/R0)**2
      C=2.*A
      DO 10 IT=1,20
      F=ALOG(R2P)+A*R2P**2+B
      DF=1./R2P+C*R2P
      R2=R2P-F/DF
      IF(ABS(R2/R2P-1).LE.1.E-04)GOTO 20
   10 R2P=R2
      WRITE(6,15)R2
   15 FORMAT(/5X,'STOP IN SUBROUTINE NEWTON ; R2=',E15.5)
      STOP
   20 RETURN
      END
```

```
      SUBROUTINE NORMA(TOT,L)
C -----------------------------------------------------------------------
C     NORMALIZATION OF THE COMPUTED VIEW FACTORS.
C
      COMMON /SC01R/ NSECT,NSECP
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC13R/ NAFF(132)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /SC16R/ ICOMP
C
      IF(ICOMP.GT.0) GO TO 200
C
C     ALL VIEW-FACTORS ARE COMPENSED (ICOMP=0)
      SUM=0.0
      NAF=NAFF(L)
      DO 100 K=1,NAF
           VFAC(L,K)=VFAC(L,K)/DAREA(L)+VFAC(L,K)/DAREA(L)*(1-TOT)/TOT
           SUM=SUM+VFAC(L,K)
           VFAC(L,K)=VFAC(L,K)*DAREA(L)
  100      CONTINUE
      GO TO 400
C
C     THE VIEW-FACTOR FII IS USED FOR THE COMPENSATION (ICOMP=1)
  200 CONTINUE
      NA1=NAFF(L)+1
      VFAC(L,NA1)=(1.0-TOT)*DAREA(L)
      SUM=0.0
      DO 300 K=1,NA1
             SUM=SUM+VFAC(L,K)/DAREA(L)
  300        CONTINUE
C
  400 CONTINUE
      IF ((SUM-1.) .GT. 0.01) WRITE(6,1001) SUM
C
      RETURN
 1001 FORMAT( 5X,'NORMA, ERROR: SUM=',E12.6)
      END
```

```
      SUBROUTINE NORMT(NSTOT,NSTR,TBT,ATOT,ASEC1,MFLOW)
C-----------------------------------------------------------------------
C     NORMT NORMALIZES THE CHANNEL TEMPERATURES TO THE TOTAL BULK
C     TEMPERATURE,THE SUBCHANNEL TEMPERATURES TO THE TEMPERATURE OF THE
C     CONTAINING CHANNELS. IT NORMALIZES ALSO THE VALUES OF THE
C     TEMPERATURES OF THE TWO PORTIONS OF THE WALL SUBCHANNELS TO THE
C     TEMPERATURE OF THE CONTAINING WALL SUBCHANNELS
C
      REAL MAV,MSCH,MAWC,MFLOW
      DIMENSION A( 42),ASCH(3)
      COMMON/GEN2/AZ( 42)/SUB1/ASCHZ( 42,3)/SUB2/TSCH( 42,3),MSCH( 42,3)
     1      /IND3/NTYP( 42)/HEA6/NPIN( 42),JPIN( 42,3)/MOB5/TAV( 42)
     2      /MOB6/MAV( 42)/WCSE7/MAWC( 18,2,2)/WCSE8/ASCHWC( 18,2,2)
     3      /WCSE9/TAVWC( 18,2,2)/LAMIN3/F1ATIP( 42),F1DTIP( 42)/LAMIN4/
     4      F2ATIP( 42,3),F2DTIP( 42,3)
      DEH=TBT*MFLOW*ASEC1/ATOT
      ASEC=0.
      DO 10 NS=1,NSTOT
      A(NS)=AZ(NS)*F1ATIP(NS)
      ASEC=ASEC+A(NS)
   10 DEH=DEH-TAV(NS)*MAV(NS)
      DEHA=DEH/ASEC
      DO 11 NS=1,NSTOT
   11 TAV(NS)=TAV(NS)+DEHA*A(NS)/MAV(NS)
      DO 5 NS=1,NSTOT
      NP=NPIN(NS)
      SHSCH=0.
      DO 1 M=1,NP
      ASCH(M)=ASCHZ(NS,M)*F2ATIP(NS,M)
    1 SHSCH=SHSCH+MSCH(NS,M)*TSCH(NS,M)
      DEH=MAV(NS)*TAV(NS)-SHSCH
      DO 4 M=1,NP
      RAPPA=ASCH(M)/A(NS)
      TSCH(NS,M)=TSCH(NS,M)+DEH*RAPPA/MSCH(NS,M)
      IF(NTYP(NS).NE.2)GOTO 4
      NSW=NS-NSTR
      SHWC=0.
      DO 2 JWC=1,2
    2 SHWC=SHWC+MAWC(NSW,M,JWC)*TAVWC(NSW,M,JWC)
      DEHWC=MSCH(NS,M)*TSCH(NS,M)-SHWC
      DO 3 JWC=1,2
      RAPPA=ASCHWC(NSW,M,JWC)/ASCH(M)
    3 TAVWC(NSW,M,JWC)=TAVWC(NSW,M,JWC)+DEHWC*RAPPA/MAWC(NSW,M,JWC)
    4 CONTINUE
    5 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PIVOT(A,B,N)
C ---------------------------------------------------------------------
C     INTERCHANGES ROWS IN ORDER TO GET THE PIVOTAL ELEMENT AS A(I,I)
C
      DIMENSION A( 34, 34),B( 34)
      COMMON /SC22C/ NTOT
C
C --- FINDS THE PIVOT ------------------------------------------------
C
      L=N
      I1=N+1
      DO 100 I=I1,NTOT
             IF(ABS(A(I,N)).GT.ABS(A(L,N))) L=I
  100        CONTINUE
      IF(L.EQ.N) GO TO 900
C
C --- IF NECESSARY INTERCHANGES ROWS --------------------------------
C
      DO 200 J=N,NTOT
             AC=A(N,J)
             A(N,J)=A(L,J)
             A(L,J)=AC
  200        CONTINUE
      BC=B(N)
      B(N)=B(L)
      B(L)=BC
C
  900 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE PIVOT1(A,B,N,NTOT,N1)
C ----------------------------------------------------------------------
C     INTERCHANGES ROWS IN ORDER TO GET THE PIVOTAL ELEMENT AS A(I,I)
C
      DIMENSION A(N1,NTOT),B(NTOT)
C
C --- FINDS THE PIVOT ----------------------------------------------
C
      L=N
      I1=N+1
      DO 100 I=I1,NTOT
             IF(ABS(A(I,N)).GT.ABS(A(L,N))) L=I
  100        CONTINUE
      IF(L.EQ.N) GO TO 900
C
C --- IF NECESSARY INTERCHANGES ROWS --------------------------------
C
      DO 200 J=N,NTOT
             AC=A(N,J)
             A(N,J)=A(L,J)
             A(L,J)=AC
  200        CONTINUE
      BC=B(N)
      B(N)=B(L)
      B(L)=BC
C
  900 CONTINUE
C
      RETURN
      END
```

```
      FUNCTION PSTAR(ZWC,P)
C ----------------------------------------------------------------------
C     PERIMETER OF A WALL TYPE SECTOR OF SHROUD
C
      PSTAR=2.*(P/2.-ZWC*1.732051+2.*ZWC)
C
      RETURN
      END
```

```
      SUBROUTINE QCOC (NCA,J,X,NSE)
C ----------------------------------------------------------------------
C     DETERMINES THE HEAT TRANSMITTED FROM EACH SECTOR TO THE GAS  IN
C     CASE OF POWER GENERATED WITHIN THE FUEL.
C
      DIMENSION X(NSE)
C
      COMMON /SC01C/ NCAN( 19),LIPS( 19,10)
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC03C/ NRODS
      COMMON /SC05C/ JZUR( 19, 42)
      COMMON /SC07C/ H1
      COMMON /SC11C/ FALFA( 42,3)
      COMMON /SC12C/ GEO( 42,3)
      COMMON /SC08R/ QPR( 42,3)
      COMMON /SC26C/ TBR( 42,3),TBS( 18,2)
      COMMON /HEA5 / QQ( 42,3)
      COMMON /QPAR1/ QDEV
C
      PIG = 3.141593
C
      IF (NCA .EQ. 1 .AND. NRODS .NE. 12 ) GO TO 2000
C
      SUM=0.0
      QTOT=0.0
      DO 1000 K=1,NCA
              NS=LIPS(J,K)
              M =JZUR(J,NS)
              Q = FALFA(NS,M)*(X(K)-TBR(NS,M))
              SUM=SUM+Q
              QU= QQ(NS,M)*H1*QDEV*GEO(NS,M)/PIG
              QTOT=QTOT+QU+QPR(NS,M)
C
              QJ(J,NS)=Q-QU
C
 1000         CONTINUE
C ---------------------------------------------------------------------
C     CONTROL ON THE COMPUTED HEAT
C
      IF(QTOT.LT.1.E-05) GO TO 9999
      DELTAQ=ABS((SUM/QTOT)-1.)
      IF (DELTAQ .GT. 0.01) WRITE(6,1100) J,DELTAQ
 1100 FORMAT( ' SUB. QCOC, HEAT BALANCE NOT RESPECTED FOR PIN ',I3,
     >        ' THE PERCENTUAL ERROR IS ',F10.5,'           ',
     >        ' CALCULATION PROCEEDING')
      GO TO 9999
C
C ---------------------------------------------------------------------
C     CASE OF NCA = 1 : QJ IS EQUAL TO QPR.
C
 2000 CONTINUE
C
      QJ(1,1)=QPR(1,1)
C
 9999 RETURN
      END
```

```
      SUBROUTINE QDEFI(QJR)
C -----------------------------------------------------------------------
C     DEFINITION OF THE ARRAYS QPR AND QSR
C      QPR=(CAL/SEC)    QSR=(CAL/SEC)
C
      COMMON /SC01R/ NSECT,NSECP
      COMMON /SC02R/ P,D,Z,ZWC,H,LENGTH
      COMMON /SC06R/ ISU(132,2)
      COMMON /SC08R/ QPR( 42,3)
      COMMON /SC09R/ QSR( 18,2)
      COMMON /SC10R/ QSTOT,QRTOT
      COMMON /SC13C/ GEO1( 42,3)
      COMMON /SC05L/ PERLT
      COMMON /DAT/   PIG
      COMMON /QPAR3/ PERL(3)
      COMMON /GEO0 / ACH(3)
      COMMON /SUB1/  ASCH( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /MART5/ NSTR
C
      DIMENSION QJR(132)
C
      REAL LENGTH
C
      AQTOT=0.0
      QTOT=0.0
      DO 1000 L=1,NSECT
              NS=ISU(L,1)
              M =ISU(L,2)
C
              IF (L .GT. NSECP) GO TO 500
C
              QPR(NS,M)=-QJR(L)*PIG*D*H/GEO1(NS,M)
              QTOT=QTOT+QPR(NS,M)
              AQTOT=AQTOT+ABS(QPR(NS,M))
                                                   GO TO  1000
C
  500         CONTINUE
              NW=NS-NSTR
              NTY=NTYP(NS)
              QSR(NW,M)= -QJR(L)*PERL(NTY)*ASCH(NS,M)/ACH(NTY)*H
              QTOT=QTOT+QSR(NW,M)
              AQTOT=AQTOT+ABS(QSR(NW,M))
C
 1000         CONTINUE
C
C     ----------------------------------------------------------------
C     CONTROL ON THE COMPUTED HEAT
C
      DELT=0.0
      QTOT1=QRTOT+QSTOT
      IF(QTOT1.GT. 1.E-06) DELT=ABS(QTOT/QTOT1)
      IF(DELT .LE. 0.01) GO TO 9999
C
      DELF=DELT*100.
      WRITE(6,9100) DELF
      ADEL=ABS(QTOT)
      DO 2000 L=1,NSECT
              NS=ISU(L,1)
              M =ISU(L,2)
              IF (L .GT. NSECP) GO TO 1500
```

```
               QPR(NS,M)= QPR(NS,M)+QPR(NS,M)*ADELT/AQTOT
               GO TO  2000
 1500        CONTINUE
             NW=NS-NSTR
             NTY=NTYP(NS)
             QSR(NW,M)= QSR(NW,M)+QSR(NW,M)*ADELT/AQTOT
 2000        CONTINUE
C
 9999 RETURN
 9100 FORMAT( ' SUB. QDEFI -',
     >        ' HEAT BALANCE AFTER RADIATION, DELT = ',F10.5,' %  ',
     >        ' CALCULATION PROCEEDING.')
      END
```

```
      SUBROUTINE QDEFIS
C ----------------------------------------------------------------------
C     HEAT TO THE GAS AFTER THE CONDUCTION IN THE SHROUD (CAL/SEC)
C
      COMMON /SHROUD/ TLINER( 18,2)
      COMMON /MART5/  NSTR
      COMMON /GASD1/  NSTOT
      COMMON /SC06L/  SHQ( 18,2)
      COMMON /SC09R/  QSR( 18,2)
      COMMON /SC17C/  SALFA( 18,2)
      COMMON /SC18C/  RE,RI,ALFW,ALFC
      COMMON /SC21C/  SHQC( 18,2)
      COMMON /SC02R/  P,D,Z,ZWC,H,ENG
      COMMON /SC26C/  TBR( 42,3),TBS( 18,2)
      COMMON /HEA6/   NPIN( 42),JPIN( 42,3)
      COMMON /IND3/   NTYP( 42)
C
      QU=0.0
      QD=0.0
      NW1=NSTR+1
      DO 1000 NS=NW1,NSTOT
              NW=NS-NSTR
              NP=NPIN(NS)
              DO 500 M=1,NP
                     ALF=ALFW
                     IF(NTYP(NS).EQ.3) ALF=ALFC
                     Q=SALFA(NW,M)*H*RI*ALF*(TLINER(NW,M)-TBS(NW,M))
                     QD=QD+Q
                     QU=SHQ(NW,M)+QSR(NW,M)+QU
                     SHQC(NW,M)=Q-SHQ(NW,M)
  500                CONTINUE
 1000         CONTINUE
C
      PC=ABS(QU/QD-1.)
      IF(ABS(QU/QD-1.).GT.0.01) WRITE(6,2000) PC
 2000 FORMAT ( ' SUB.QDEFIS; PRECISION (WORSE THAN 1%) =',E12.6,5X,
     >          ' CALCULATION PROCEEDING')
C
      RETURN
      END
```

```
      SUBROUTINE RADIA
C -----------------------------------------------------------------
C     MODEL FOR THE THERMAL EXCHANGES DUE TO RADIATION.
C
      DIMENSION A(132, 13),B(132),QJR(132)
C
C
C  01 ----------------------------------------------------------------
C     CONSTRUCTION OF THE COEFFICIENTS ARRAY.
C
      CALL MATBUI (A,B)
C
C  02 ----------------------------------------------------------------
C     SOLUTION OF THE SYSTEM.
C
      CALL SYSOL (A,B,QJR)
C
C  03 ----------------------------------------------------------------
C     REDEFINITION OF THE COMPUTED HEAT.
C
      CALL QDEFI(QJR)
C
C  04 ----------------------------------------------------------------
C
      RETURN
      END
```

```
      SUBROUTINE RECANG(I,AI,NS,K,IVIA,IRH,ALFA,AMA1,TI,PB,D,W,RH,DETOT
     *,PROV,DAI,DBI,AAI,ABI,G,SSSA,SSSB, AMTI,NTYP,H1,H,PR1,PR2,SQDPG,JJ
     *J,TE,SUR,TW1,AMAI,TAI,AMBI,TBI,III,TIE,TIAV,HPLUSB,HPLUSW,ANGT,EM1
     *,XC1,XC2,*,DEPA,CS)
C-----------------------------------------------------------------------
C     SUBROUTINE RECANG EVALUATES FRICTION FACTORS AND APPROXIMATE MASS
C     FLOW RATES AND TEMPERATURES FOR WALL-TYPE SUB-SUBCHANNELS.
C
      REAL LAMIA,LAMIB,KI, KAPPA,NUI,NUO
      COMMON/CORR1/SIGMAI( 42,3),PHII( 42,3)/COLAM1/COLAMB/COLAM2/COLAMA
     1      /CORR2/CHI( 18,2,2),PSI( 18,2,2)/GRID2/YY(100, 42,3)
     2      /ANG1/RA2(60)/HEA5/QQ( 42,3)/DAT/PIG/REC1/PVERT(90),PRAD(90)
     3      /SUB20/PROVI( 18,2)/GEN5/DE( 42)/SUB22/TW( 42,3)/MART/ITCORR
     4      /SUB21/TSCHA( 18,2),TSCHB( 18,2)
     5      /WSSCH1/DELTIE( 18,2,90),DTIEAV( 18,2)/WSSCH2/TIO( 18,2,90)
     6      /WSSCH/T1SSC1( 18,2),T2SSC1( 18,2),
     A             T1SSC2( 18,2),T2SSC2( 18,2)
     7      /WSSCH0/TBSSC1( 42,3),TWSSC1( 42,3),TBSSC2( 42,3),
     A              TWSSC2( 42,3)
     8      /GRAV/IGRAV/GAGR/DPSI/GAAG2/FCOPW2( 18,2)
      COMMON /QPAR1/QDEV/QPAR2/QLINM,QLDEV
      COMMON /HEA6 / NPIN( 42),JPIN( 42,3)
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC13C/ GEO1( 42,3)
      COMMON /SC06L/ SHQ( 18,2)
      COMMON /SC21C/ SHQC( 18,2)
      COMMON /SC09R/ QSR( 18,2)
      COMMON /GEO0 / ACH(3)
      COMMON /SUB1 / ASCH( 42,3)
      COMMON /QPAR3/ PERL(3)
C
      PERLS=PERL(NTYP)*ASCH(NS,JJJ)/ACH(NTYP)
C
      ICS=1
      IF(I.GT.1)TWI=TW1
      IF(ITCORR.EQ.1)PROVI(III,JJJ)=PROV
      PROVI(III,JJJ)=PROV
      DEPA=DETOT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     QROD=QQ(NS,JJJ)*QDEV
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      JP=JPIN(NS,JJJ)
      QROD=QQ(NS,JJJ)*QDEV+QJ(JP,NS)*GEO1(NS,JJJ)/H1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      Q=QROD*ALFA/(2.*PIG)*H1
      QA=QROD/SUR
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 07.11.1979 CCCCCCCCCC
C     QLIN=QLINM*H1*QLDEV
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 07.11.1979 CCCCCCCCCC
      QLIN=(SHQ(III,JJJ)+SHQC(III,JJJ))/PERLS
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 07.11.1979 CCCCCCCCCC
      AMABI=AMA1
C     ..................................................................
C     LOOP ITW1 STARTS (CALCULATION OF THE BULK TEMPERATURES OF THE
C     TWO ZONES DIVIDED BY THE TAU=0 LINE, TAI AND TBI)
C
      DO 2000 ITW1=1,10
C     ..................................................................
C     LOOP ITW STARTS (CALCULATION OF THE PIN TEMPERATURE TWI)
C
       DO 14 ITW=1,20
C     ..................................................................
```

```
C     LOOP ITTEMP STARTS (CALCULATION OT THE FRICTION FACTORS AND OF
C     THE MASS FLOW RATES FOR THE TWO ZONES DIVIDED BY THE TAU=0 LINE
C     AND OF THE BULK TEMPERATURE TI FOR THE WHOLE SUB-SUBCHANNEL)
C
      DO 7 ITTEMP=1,60
      IF(ITW1.GT.1)GOTO 1998
      IF(              ITCORR .GT.1 .AND. QQ(NS,JJJ).GT. 1.E-06)GOTO 25
      TAI=TI
      TBI=TI
      GOTO 26
   25 CONTINUE
      TAI=TSCHA(III,JJJ)
      TBI=TSCHB(III,JJJ)
      TI=TIAV
   26 CONTINUE
      IF(ITW.EQ.1 .AND. I.EQ.1)TWI=TBI
      IF(ITW.EQ.1)TWO=TWI
 1998 CONTINUE
      ETAA=ETA(PB,TAI)
      ETAB=ETA(PB,TBI)
      RHOA=RHO(PB,TAI)
      RHOB=RHO(PB,TBI)
      ETAIW=ETA(PB,TWI)
      RHOIW=RHO(PB,TWI)
      QPLUS=QA/(AMABI*CP(PB,TBI)*(TE+273.16))
C
      IF(IVIA.EQ.2 .OR. ITW1.GT.1)GOTO 1
C     .................................................................
C     CALCULATION OF THE POSITION OF THE TAU=0 LINE
C
      CALL TLINE(I,AI,ITTEMP,NS,K,ALFA,D,W,RH,DEPA  ,PROVI(III,JJJ),IRH,
     *DAI,DBI,AAI,ABI,RHPL,G,TWI,TE,QPLUS,ETAA,RHOA,ETAB,RHOB,ETAIW,
     *RHOIW,ANGT,EM1,XC1,XC2,TBI,&8500,CS)
C
    1 CONTINUE
      PAI=4.*AAI/DAI
      R0=0.5*SQRT(D**2+D*DBI)
      YDH=(R0-0.5*D)/RH
      IF(ITTEMP.EQ.1 .AND. ITW.EQ.1 .AND. ITW1.EQ.1)GOTO 30
C     .................................................................
C     AFTER THE FIRST ITERATION THE FRICTION FACTORS ARE EVALUATED
C     BY MEANS OF THE REYNOLDS NUMBERS AND OF THE FRICTION FACTORS
C     COMPUTED AT THE PRECEEDING ITERATION
C
      REAI=AMAI*DAI/(AAI*ETAA)
      REBI=AMBI*DBI/(ABI*ETAB)
      REIW=(ETAB*RHOIW)/(ETAIW*RHOB)*REBI
      IF(REAI.GT.0. .AND. REBI.GT.0. .AND. SQ8LIA.GT.0. .AND. SQ8LIB.GT.
     *0.)GOTO 700
      WRITE(6,699)NS, JJJ,I,REAI,SQ8LIA,REBI,SQ8LIB,ITCORR,ICS
  699 FORMAT(//5X,'SUB. RECANG',
     >      /5X,'NS=',I5,5X,'M=',I2,5X,'I=',I3/5X,'RE A=',E15.5,5X,'SQ
     *RT(8/LAMBDA) A=',E15.5/5X,'RE B=',E15.5,5X,'SQRT(8/LAMBDA) B=',E15
     *.5/5X,'ITCORR=',I5,5X,'ICS=',I2)
 8500 RETURN 1
C
  700 CONTINUE
      IF(IRH.EQ.2)GOTO 27
      SQ8LIB=2.5*ALOG(REBI/SQ8LIB)+5.5-G
      GA=6.0737
      GOTO 28
   27 HPLUSB=RH/DBI*REBI /SQ8LIB
      HPLUSW=HPLUSB*REIW/REBI
      RHPL=RHPLUS(HPLUSB,TWI,TE,QPLUS,HPLUSW,TBI,YDH)
      SQ8LIB=2.5*ALOG(DBI/RH)+RHPL-G
```

```
      GA=5.966
   28 IF(NTYP.EQ.3 .AND. IVIA.EQ.2)GOTO 29
      SQ8LIA=CS*(2.5*ALOG(REAI/SQ8LIA)-GA)+5.5*COLAMA
      GOTO 3
   29 SQ8LIA =SMFUN1(RHOA,ETAA,DETOT,PROV,I,2,REAI,DAI,SQ8LIA,RO,GA,CS)
    3 CS=CSFUN(IRH,REAI,SQ8LIA,SQ8LIB,GA)
      GOTO 6
C     ................................................................
C     FIRST ITERATION : THE FRICTION FACTORS ARE EVALUATED BY MEANS
C     OF THE EQUATION (LAMBDAI*RHOI**2/DI) = (LAMBDA*RHO*U**2/D) TOT.
C
   30 IF(IRH.EQ.2)GOTO 2
      SQ8LIB=2.5*ALOG(PROV/ETAB*SQRT((DBI/DETOT)**3*RHOB))+5.5-G
      GA=6.0737
      GOTO 4
    2 HPLUSB=RH/DETOT*PROV/ETAB*SQRT(DBI/DETOT*RHOB)
      HPLUSW=RH/DETOT*PROV/ETAIW*SQRT(DBI/DETOT*RHOIW)
      RHPL=RHPLUS(HPLUSB,TWI,TE,QPLUS,HPLUSW,TBI,YDH)
      SQ8LIB=2.5*ALOG(DBI/RH)+RHPL-G
      GA=5.966
    4 IF(NTYP.EQ.3 .AND. IVIA.EQ.2)GOTO 5
      SQ8LIA=CS*(2.5*ALOG(PROV/ETAA*SQRT((DAI/DETOT)**3*RHOA))-GA)+5.5
     **COLAMA
      GOTO 6
    5 SQ8LIA =SMFUN1(RHOA,ETAA,DETOT,PROV,I,1,REAI,DAI,SQ8LIA,RO,GA,CS)
C
C
    6 CONTINUE
       LAMIA=8./SQ8LIA**2
       LAMIB=8./SQ8LIB**2
      SSSA=AAI/SQRT(LAMIA*H/(2.*RHOA*DAI))
      SSSB=ABI/SQRT(LAMIB*H/(2.*RHOB*DBI))
      SQDPGB=SQRT(ABS(SQDPG**2*DPSI-IGRAV*RHOB*980.665*H)
     */FCOPW2(III,JJJ))
      SQDPGA=SQRT(ABS(SQDPG**2*DPSI-IGRAV*RHOA*980.665*H)
     */FCOPW2(III,JJJ))
      AMBI=SSSB*SQDPGB+ABI*SIGMAI(NS,JJJ)*CHI(III,JJJ,1)
      AMAI=SSSA*SQDPGA+AAI*SIGMAI(NS,JJJ)*CHI(III,JJJ,1)
      AMTI=AMAI+AMBI
      IF(                 ITCORR.GT.1 .AND. QQ(NS,JJJ).GT.1.E-06  .AND.
     *ITW1.EQ.1)GOTO 48
C
      DELTAT=(Q+QLIN*PAI+ PHII(NS,JJJ)*PSI(III,JJJ,1)*(AAI+ABI))/
     *(AMTI*CP(PB,TI))
      TI=TIE+0.5*DELTAT+DELTIE(III,JJJ,I)
      TIO(III,JJJ,I)=TI+0.5*DELTAT
C
   48 CONTINUE
      IF(ITTEMP.EQ.1)GOTO 50
      IF(ABS(AMAI/AMAI1-1.).LE.1.E-03 .AND. ABS(AMBI1/AMBI-1.).LE.1.E-03
     *)GOTO 9
   50 AMAI1=AMAI
      AMBI1=AMBI
      AMABI=AMBI/ABI
    7 CONTINUE
C     ................................................................
C     END OF LOOP ITTEMP: POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
      WRITE(6,8)I,NS,K,ITW,ITCORR
    8 FORMAT(    5X,'SUB. RECANG.',
     >           /5X,'CALCULATION STOPS: ITTEMP=10 FOR SUBCHANNEL',I4,2X,
     *'OF CHANNEL',I4,2X,'(AXIAL SECTION',I4,') ITW=',I2,5X,'ITCORR=',I5
     *)
      RETURN 1
C     ................................................................
```

```
C     CONVERGENCE  IS REACHED IN THE LOOP ITTEMP
C
    9 CONTINUE
      IF(ITW1.EQ.1)TW1=TWI
      IF(QQ(NS,JJJ).LE.1.E-06)GOTO 2002
      ATI=AAI+ABI
      DEI=ATI/(AAI/DAI+ABI/DBI)
      IF(IRH.EQ.1)GOTO 600
C     ..................................................................
C     CALCULATION OF THE PIN TEMPERATURE ONLY FOR HEATED ROUGHENED
C     SECTIONS
C
      IF(ABS(TWO).LT.3000. .AND. ABS(TWI).LT.3000.)GOTO 2005
      WRITE(6,2004)NS,JJJ,TWO,TWI
 2004 FORMAT(     5X,'STOP IN RECANG: NS=',I5,5X,'JJJ=',I5/5X,'TWO=',E15.
     *5,5X,'TWALL=',E15.5)
      RETURN 1
 2005 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 21.09.1979
C     IF(NTYP.EQ.3 .AND. ITVIA.EQ.2)GOTO 500
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(NTYP.EQ.3 .AND. IVIA.EQ.2)GOTO 500
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      R2=R0+0.25*DAI*XC2
      GOTO 501
  500 R2=RA2(I)
  501 CONTINUE
      R2MROH=(R2-R0)/RH
      R1=D*0.5
      R1DR2=R1/R2
      FACHE=TIS(R1,R2,IRH)
      KI=KAPPA(PB,TI)
      ETAI=ETA(PB,TI)
      RHOI=RHO(PB,TI)
      CPI=CP(PB,TI)
      PRI=ETAI*CPI/KI
      REI=AMTI*DEI/(ETAI*ATI)
      U1DU=AMBI*ATI*RHOI/(AMTI*ABI*RHOB)
      REWO=REIW*ETAIW*RHO(PB,TWO)/(RHOIW*ETA(PB,TWO))
      HPLUSO=HPLUSW*REWO/REIW
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      CALL SPANU(REI,PRI,NS,JJJ,Y)
C     CALL RNU(HPLUSW,TWI,LAMIB,REI,PRI,TI ,YDH,R1DR2,R2MROH,U1DU,REIW,
C    1  YY(K,NS,JJJ),NUI,GHPL)
C     FURTHER MODIFIED AT 09.10.1980
C     CALL RNU(HPLUSW,TWI,LAMIB,REI,PRI,TI ,YDH,R1DR2,R2MROH,U1DU,REIW,
C    1  Y,NUI,GHPL,2)
      CALL RNU(HPLUSW,TWI,LAMIB,REI,PRI,TI ,YDH,R1DR2,R2MROH,U1DU,REIW,
     1   Y ,NUI,GHPL,2,R0,R1,R2,0.0)
      CALL RNU(HPLUSO,TWO,LAMIB,REI,PRI,TI ,YDH,R1DR2,R2MROH,U1DU,REWO,
     |    1.,NUO,GHPL,2,R0,R1,R2,0.0)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      ALFAI=NUI*KI/DEI*FACHE
      TIW=TI+QA/ALFAI
      ALFAO=NUO*KI/DEI
      TWO=TI+QA/ALFAO
      IF(ABS(TWI/TIW-1.).LE.1.E-04)GOTO 16
   14 TWI=TIW
C     ..................................................................
C     END OF LOOP ITW : POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
      WRITE(6,15)I,JJJ,NS
   15 FORMAT(     5X,'SUB. RECANG',
     >          /5X,'CALCULATION STOPS:ITW =10 FOR SUB-SUBCH.',I3,2X,'(M
```

```
     *=',I2,2X,'NS=',I5,')')
      RETURN 1
C     .................................................................
C     CONVERGENCE  IS REACHED IN THE LOOP ITW
C
   16 CONTINUE
      IF(ITW1.GT.1)GOTO 1999
      TW1=TWI
      IF(ITCORR.EQ.1)RETURN
C     .................................................................
C     CALCULATION OF THE BULK TEMPERATURES OF THE TWO ZONES DIVIDED BY
C     THE TAU=0 LINE ONLY FOR HEATED ROUGHENED SECTIONS AT ITCORR>1
C
 1999 U1STAR=AMBI/(RHOB*ABI)*SQRT(LAMIB*0.125)
      FF=QA/(RHOI*CPI*U1STAR)
      R0DR2=R0/R2
      CALL DDONNE(TW0,TI,GHPL,R0DR2,R1DR2,YDH,R2MROH,FF,TAI,TIB,TE)
      IF(ABS(TBI/TIB-1.).LE.1.E-04)GOTO 2002
 2000 TBI=TIB
C     .................................................................
C     END OF LOOP ITW1: POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
       WRITE(6,2001)I,NS,JJJ,ITCORR,TBI,TAI,TI,TWI,TW0
 2001 FORMAT(/5X,'STOP IN RECANG (LOOP ITW1) I=',I3,' NS=',I5,' M=',I2,
     1 ' ITCORR=',I3/5X,'TBI=',E15.5,5X,'TAI='  ,E15.5,5X,'TI=',E15.5,5X,
     2 'TWI=',E15.5,5X,'TW0=',E15.5)
      RETURN 1
C     .................................................................
C     CONVERGENCE  IS REACHED IN THE LOOP ITW1
C
  600 TWI=TW(NS,JJJ)
C
 2002 CONTINUE
      TBSSC2(NS ,JJJ)=TI
      T1SSC2(III,JJJ)=TBI
      T2SSC2(III,JJJ)=TAI
      TWSSC2( NS,JJJ)=TWI
      IF(I.GT.1)RETURN
C
      TBSSC1( NS,JJJ)=TI
      T1SSC1(III,JJJ)=TBI
      T2SSC1(III,JJJ)=TAI
      TWSSC1( NS,JJJ)=TWI
      RETURN
      END
```

```
      SUBROUTINE RECCA1(K,NS,N,NSC45,IRH,PROV,PB,   RH,H1,ALFA,A,DE,MEC,
     *AT,DET,ATOT,DETOT,MFLOW,W,D,C, JJJ,NSTR,H,PR1,PR2,SQDPG,TE,SUR,
     *AMT,DDDD,ATSCH,CTU3,EM1,*,ALFACE)
C-----------------------------------------------------------------------
C     SUBROUTINE RECCA1 CALCULATES FRICTION FACTORS  AND APPROXIMATE
C     OUTLET MASS FLOW RATES AND TEMPERATURES FOR WALL CHANNELS AND SUBC
C
      REAL MEC,MFLOW,LAMB,LAMSCH,LAMWC,MSCH1,KAPPA,LAMLAM,MSCH,
     1    MWC1L,MWC2L
      DIMENSION A(46),DE(46),MEC(46)
      COMMON/WACO1/XMSCHB( 18,2),XMSCHA( 18,2)/DAT/PIG/CEN1/G(46)
     0      /REC1/      PVERT(90),PRAD(90)/REC2/E(90)/REC3/P(90)
     1      /SUB1/ASCH( 42,3)
     2      /SUB4/LAMB( 18,2)/SUB5/LAMSCH( 42,3)
     3      /SUB8/MSCH1( 42,3)/SUB23/HPLUSB( 42,3),HPLUSW( 42,3)
     4      ,QPLUS( 42,3),PRB ( 42,3),YODH( 42,3)/HEA5/QQ( 42,3)
     5      /SUB22/TW( 42,3)/WCSE1/DEWC( 18,2,2),PHWC( 18,2,2)
     6      /LAMIN0/I2TIP( 42,3)/LAMIN1/AKAPPA( 42) /LAMIN2/FATIP(3),
     7      FDTIP(3)/LAMIN3/F1ATIP( 42),F1DTIP( 42)
     A      /LAMIN4/F2ATIP( 42,3),
     8      F2DTIP( 42,3)/LAMIN5/RTIP(7)/LAMIN6/ANGLAM/LAMIN7/F1PTIP
     9      /WSSCH1/DELTIE( 18,2,90),DTIEAV( 18,2)/WSSCH2/TIO( 18,2,90)
      COMMON /INPAR/IPA
      COMMON /SUB2/TSCH( 42,3),MSCH( 42,3)/SUB3/ADAB( 18,2),DETB( 18,2)
      COMMON/LAMINK/BKAPPA(7,3)/QPAR1/QDEV/QPAR2/QLINM,QLDEV/HEA10/
     1      QSCH( 42,3)/WALLCO/WFCO1( 18,2),WFCO( 18,2)/WALLKA/AKAWC(2)
     2      /WCSE3/LAMWC( 18,2,2)
     A      /WCSE4/CTURB2( 18,2)/WCSE8/ASCHWC( 18,2,2)
     3      /WCSE5/TSCWC1( 18,2,2)/WCSE9/TAVWC( 18,2,2)/GEN2/ACHA( 42)
     4      /CORR1/SIGMAI( 42,3),PHII( 42,3)
     A      /CORR2/CHI( 18,2,2),PSI( 18,2,2)
     5      /WSSCH/T1SSC1( 18,2),T2SSC1( 18,2),
     A             T1SSC2( 18,2),T2SSC2( 18,2)
     6      /WSSCH0/TBSSC1( 42,3),TWSSC1( 42,3),
     A              TBSSC2( 42,3),TWSSC2( 42,3)
     7      /LAMIN9/I3TIP( 42,3)/SHROUD/TLINER( 18,2)/MART/ITCORR
     8      /GRAV/IGRAV/GAGR/DPSI/GAAG1/FCOPW1(3)/GAAG2/FCOPW2( 18,2)
      COMMON /HEA6 / NPIN( 42),JPIN( 42,3)
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC06L/ SHQ( 18,2)
      COMMON /SC21C/ SHQC( 18,2)
      COMMON /SC09R/ QSR( 18,2)
      COMMON /QPAR3/ PERL(3)
      COMMON /GEO0 / ACH(3)
      IF(JJJ.GT.1)GOTO 2998
      F1ATIP(NS)=0.
      F1PTIP=0.
 2998 CONTINUE
      III=NS-NSTR
      IF(ITCORR.EQ.1 .AND. K.EQ.1) FCOPW2(III,JJJ)=FCOPW1(2)
      DTIEAV(III,JJJ)=0.
      I2TIP(NS,JJJ)=I3TIP(NS,JJJ)
      IF(                  I2TIP(NS,JJJ).EQ.1)GOTO 2999
C     ..................................................................
C     ..................................................................
C     I3TIP#1: THE TURBULENT CALCULATION MUST BE PERFORMED
C
      TWIAV=0.
      CS=1.
```

```
      AMA1=MSCH1(NS,JJJ)/AT
      TETA=ALFA
      ANGT=0.
      AMT=0.
      AMA=0.
      TT=0.
      TTA=0.
      DDDDA=0.
      ATA=0.
      DDDDB=0.
      SRAMIB=0.
      SRAMIA=0.
      HPLUSB(NS,JJJ)=0.
      HPLUSW(NS,JJJ)=0.
      TI=TSCWC1(III,JJJ,1)
      SIGMA2=SIGMAI(NS,JJJ)*CHI(III,JJJ,2)
      PHI2=PHII(NS,JJJ)*PSI(III,JJJ,2)
      ASCHWC(III,JJJ,1)=0.
      IVIA=1
      EMAX=EM1
      XC1=0.
      XC2=1.
      IF(ITCORR.EQ.1)DEWC(III,JJJ,1)=DETOT
C     .................................................................
C     CALCULATION OF THE "WALL-TYPE SUB-SUBCHANNELS (I= SUB-SUBCHANNEL
C     INDEX)
C
      DO 3 I=1,N
      AI=I
C
    1 CONTINUE
      ANGT=ANGT+TETA
      CALL RECANG(I,AI,NS,K,IVIA,IRH,TETA,AMA1,TI,PB,D,W,RH,DETOT,PROV,
     *DAI,DBI,AAI,ABI,GG,SSSA,SSSB,AMTI,2,H1,H,PR1,PR2,SQDPG,JJJ,TE,SUR,
     *TWI,AMAI,TAI,AMBI,TBI,III,TSCWC1(III,JJJ,1),TAVWC(III,JJJ,1),
     *HPLUS1,HPLUS2,ANGT,EM1,XC1,XC2,&777,DEWC(III,JJJ,1),CS)
      IF(E(I).GE.EMAX .AND. IVIA.EQ.1)GOTO 5
C
      TWIAV=TWIAV+TWI*TETA
      AMT=AMT+AMTI
      AMA=AMA+AMAI
      RAMIA=AMTI*AAI/(AAI+ABI)
      RAMIB=AMTI*ABI/(AAI+ABI)
      TT=TT+AMTI*TI
      TTA=TTA+RAMIA*TAI
      SRAMIA=SRAMIA+RAMIA
      SRAMIB=SRAMIB+RAMIB
      DDDDA=DDDDA+SSSA
      DDDDB=DDDDB+SSSB
      DDDD=DDDDA+DDDDB
      ATA=ATA+AAI
      ASCHWC(III,JJJ,1)=ASCHWC(III,JJJ,1)+AAI+ABI
      DTIEAV(III,JJJ)=DTIEAV(III,JJJ)+AMTI*DELTIE(III,JJJ,I)
      IF(IRH.EQ.1)GOTO 30
      HPLUSB(NS,JJJ)=HPLUSB(NS,JJJ)+HPLUS1*ABI
      HPLUSW(NS,JJJ)=HPLUSW(NS,JJJ)+HPLUS2*ABI
   30 CONTINUE
      IF(IVIA.EQ.1)GOTO 3
      IF(ABS(EMAX*2./C-1.).LE.1.E-05)GOTO 10
C     .................................................................
C     POINT REACHED BY THE CALCULATION IF THE SHROUD PROFILE HAS
C     BLOCKING TRIANGLES
C
      IVIA=1
      EMAX=C*0.5
      XC1=1./SQRT(3.)
```

```
      XC2=2.*XC1
      TETA=ALFA
      E(I)=EM1
      P(I)=PP
    3 CONTINUE
C     ..............................................................
C     I HAS REACHED THE VALUE N, WHICH WOULD MEAN NO "CENTRAL-TYPE"
C     SUB-SUBCHANNELS
C
      WRITE(6,4)NS,JJJ,E(I),ITCORR    ,(I,PVERT(I),PRAD(I),I=1,N)
    4 FORMAT(1H1,5X,'SUB. RECCA1',
     >          /5X,'CALCULATION STOPS: NO CENTRAL SUBCHANNELS IN WALL C
     *HANNEL',I4/5X,'M=',I2, 5X,'E(I)=',E15.5,5X, ' ITCORR=',I3
     *        /(5X,'I=',I3,5X,'PVERT=',E15.5,5X,'PRAD=',E15.5))
      RETURN 1
C     ..............................................................
C     RECALCULATION OF THE SUB-SUBCHANNEL FOR WHICH IT WAS E(I)>EMAX,
C     IN ORDER TO FIT EMAX (I.E. E(I)=EMAX)
C
    5 CONTINUE
      IVIA=2
      II=I
      ANGT=ANGT-TETA
      DEE=EMAX-E(I-1)
      PP=P(I-1)-DEE*(P(I-1)-P(I))/(E(I)-E(I-1))
      BETA=ATAN(EMAX*2./(PP*D))
      TETA=BETA-ANGT
      PVERT(I)=PP*D*0.5
      PRAD(I)=PVERT(I)/COS(BETA)
      PAI=DEE*XC2
      WW=W-((EMAX+E(I-1))*0.5-EM1)*XC1
      DAI=4.*(WW-0.5*(D+PVERT(I)+PVERT(I-1)))/XC2
      DBI=2.*(P(I-1)*EMAX-PP*E(I-1))/TETA-D
      PBI=TETA *D*0.5
      AAI=DAI*PAI*0.25
      ABI=DBI*PBI*0.25
      EPS=SQRT(1.+DBI/D)
      GG=GSTAR(EPS)
      GOTO 1
C     ..............................................................
C     ALL THE "WALL-TYPE SUB-SUBCHANNELS HAVE BEEN COMPUTED: CALCULATION
C     OF AVERAGE SUB-SUBCHANNEL VARIABLES FOR THE WALL PORTION
C
   10 CONTINUE
      DTIEAV(III,JJJ)=DTIEAV(III,JJJ)/AMT
      TSCHAB=TT/AMT
      RHOTAB=RHO(PB,TSCHAB)
      PHWC(III,JJJ,1)=BETA*D*0.5
      PSHWC=(EMAX-EM1)*XC2+EM1
      PHWCTL=PHWC(III,JJJ,1)+PSHWC
      DEWC(III,JJJ,1)=4.*ASCHWC(III,JJJ,1)/PHWCTL
      LAMWC(III,JJJ,1)=((ASCHWC(III,JJJ,1)/DDDD)**2)*2.*DEWC(III,JJJ,1)*
     *      RHOTAB/H
      ATB=ASCHWC(III,JJJ,1)-ATA
      ADAB(III,JJJ)=ASCHWC(III,JJJ,1)/ATB
      DETB(III,JJJ)=4.*ATB/PHWC(III,JJJ,1)
      DDDDB=DDDD-DDDDA
      XMSCHA(III,JJJ)=AMA
      XMSCHB(III,JJJ)=AMT-XMSCHA(III,JJJ)
      TSCHB=(TT-TTA)/SRAMIB
      RHOTB=RHO(PB,TSCHB)
      LAMB(III,JJJ)=((ATB/DDDDB)**2)*2.*DETB(III,JJJ)*RHOTB/H
      AMTAB=AMT
      TTAB=TT
      DDDDAB=DDDD
C     ..............................................................
```

```
C     CALCULATION OF THE "CENTRAL-TYPE" SUB-SUBCHANNELS
C
      ALFC=ALFACE
      GAMMA=PIG*0.5-BETA
      AN1=GAMMA/ALFACE
      N1=AN1
      IF(N1.EQ.0)ALFC=GAMMA
      IF(N1.EQ.0)N1=1
      IF(N1.LE.NSC45)GOTO 12
      WRITE(6,11)NS,K,ITCORR
   11 FORMAT(1H1,5X,'SUB. RECCA1',
     >          /5X,'N1 GREATER THAN NSC45 FOR CHANNEL',I4,2X,'(AXIAL SE
     *CTION',I3,')'/5X,'ITCORR=',I3)
      RETURN 1
C
   12 CONTINUE
      L=II
      II1=II+1
      DO 1000 I=II1,N
 1000 TIO(III,JJJ,I)=TIO(III,JJJ,L)
      AN1=N1
      BETA1=ALFC*AN1
      IF(ABS(BETA1/GAMMA-1.).LT.1.E-06)GOTO 99
C     .................................................................
C     CALCULATION OF THE CENTRAL SUB-SUBCHANNEL DEFINED BY AN ANGLE
C     OF THE ROD SECTOR = ALFA1 ( IF ALFA1>0 )
C
      ALFA1=GAMMA-BETA1
      E1=C*0.5*TAN(BETA1)
      DELTAE=PVERT(II)-E1
      AA=C*DELTAE*0.25-ALFA1*D**2*0.125
      DD=8.*AA/(ALFA1*D)
      EPS=SQRT(1.+DD/D)
      GG=GSTAR(EPS)
      AM1=MFLOW*AA/ATOT
      L=II+1
      CALL CEWA(K,NS,IRH,PROV,PB,RH,AA,DD,GG,AM1,DETOT,H1,ALFA1,L,JJJ,H,
     *PR1,PR2,SQDPG,AMT,TT,DDDD,TE,SUR,2,III,HPLUSB(NS,JJJ),HPLUSW(NS,JJ
     *J),TSCWC1(III,JJJ,2),SIGMA2,PHI2,&777,D,TWI,TICEN,C)
      TWIAV=TWIAV+TWI*ALFA1
C     .................................................................
C     CALCULATION OF THE "CENTRAL-TYPE  SUB-SUBCHANNELS DEFINED BY AN
C     ANGLE OF THE ROD SECTOR = ALFC
C
   99 CONTINUE
      DO 13 J=1,N1
      I=N1-J+1
      IF(N1.EQ.1)GOTO 100
      AA=A(I)
      DD=DE(I)
      GG=G(I)
      AM1=MEC(I)
      GOTO 101
  100 AA=(C**2*TAN(ALFC)-D**2*ALFC)*0.125
      DD=8.*AA/(ALFC*D)
      EPSEPS=SQRT(1.+DD/D)
      GG=GSTAR(EPSEPS)
      AM1=AA*MEC(1)/A(1)
  101 LL=L+J
      CALL CEWA(K,NS,IRH,PROV,PB,RH,AA,DD,GG,AM1,DETOT,H1,ALFC,LL,JJJ,H,
     *PR1,PR2,SQDPG,AMT,TT,DDDD,TE,SUR,2,III,HPLUSB(NS,JJJ),HPLUSW(NS,JJ
     *J),TSCWC1(III,JJJ,2),SIGMA2,PHI2,&777,D,TWI,TICEN,C)
      TWIAV=TWIAV+TWI*ALFC
   13 CONTINUE
C     .................................................................
C     THE CALCULATION OF THE "CENTRAL-TYPE" SUB-SUBCHANNELS HAS BEEN
```

```
C     COMPLETED: CALCULATION OF AVERAGE SUB-SUBCHANNEL VARIABLES FOR THE
C     WHOLE CENTRAL PORTION AND FOR THE WHOLE WALL SUBCHANNEL
C
      TWIAV=TWIAV*2./PIG
      PHWC(III,JJJ,2)=GAMMA*D*0.5
      ASCHWC(III,JJJ,2)=AT-ASCHWC(III,JJJ,1)
      DEWC(III,JJJ,2)=4.*ASCHWC(III,JJJ,2)/PHWC(III,JJJ,2)
      TSCHC=(TT-TTAB)/(AMT-AMTAB)
      RHOTC=RHO(PB,TSCHC)
      DDDDC=DDDD-DDDDAB
      LAMWC(III,JJJ,2)=((ASCHWC(III,JJJ,2)/DDDDC)**2)*2.*DEWC(III,JJJ,2)
     *                 *RHOTC/H
       ATSCH=TT/AMT
      RHOT=RHO(PB,ATSCH)
      DO 14 JWC=1,2
   14 DDDD=DDDD+ASCHWC(III,JJJ,JWC)*SIGMAI(NS,JJJ)*(CHI(III,JJJ,JWC)-1.)
     */(SQRT(ABS(SQDPG**2*DPSI-IGRAV*RHOT*980.665*H)))
      LAMSCH(NS,JJJ)=((AT/DDDD)**2)*2.*DET*RHOT/H
      CTURB2(III,JJJ)=TURBWC(CTU3,PVERT(II),PRAD(II),D,W,C,GAMMA,ASCHWC
     *(III,JJJ,1),ASCHWC(III,JJJ,2),DEWC(III,JJJ,1),DEWC(III,JJJ,2),EM1)
      I2TIP(NS,JJJ)=0
      F2ATIP(NS,JJJ)=1.
      F2DTIP(NS,JJJ)=1.
      IF(I3TIP(NS,JJJ).EQ.2)GOTO 3000
      IF(ITCORR.GT.1)GOTO 2999
      MSCH(NS,JJJ)=AMT
      TSCH(NS,JJJ)=ATSCH
      TW(NS,JJJ)=TWIAV
C     ..................................................................
C     ..................................................................
C     FOR I3TIP=1 OR I3TIP=3
C
 2999 CONTINUE
      ZWC=(C*0.5-EM1)/SQRT(3.)
      PPPP=(W-0.5*D-ZWC)*ANGLAM
      OMEGA=ATAN(PPPP*2. /C)
      PHWC1L=(PIG*0.5-OMEGA)*RTIP(IPA)
      PHWC2L=OMEGA*RTIP(IPA)
      AWC2L=          C*0.25*PPPP-RTIP(IPA)**2*0.5*OMEGA
      AWC1L=ASCH(NS,JJJ)*FATIP(2)-AWC2L
      PHWCTL=PHWC1L+2.*ZWC+EM1
      DEWC1L=4.*AWC1L/PHWCTL
      DEWC2L=4.*AWC2L/PHWC2L
      MWC1L=MSCH(NS,JJJ)*AWC1L/(ASCH(NS,JJJ)*FATIP(2))
      MWC2L=MSCH(NS,JJJ)-MWC1L
      R1DR2L=1./SQRT(1.+ 2.*AWC1L/(PHWC1L*RTIP(IPA)))
      R21WA=RTIP(IPA)/R1DR2L
      R02WA=SQRT(RTIP(IPA)**2+2.*RTIP(IPA)*AWC2L/PHWC2L)
      PHWCTE=1.
      PHWC1E=1.
      IF(QQ(NS,JJJ).LE.1.E-06)GOTO 4444
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     QROD=QSCH(NS,JJJ)*QDEV
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      JP=JPIN(NS,JJJ)
      QROD=QSCH(NS,JJJ)*QDEV + QJ(JP,NS)/H1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  07.11.1979 CCCCCCCC
C     QLIN=QLINM*QLDEV*C*0.5
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  07.11.1979 CCCCCCCC
      PERLS=PERL(2)*ASCH(NS,JJJ)/ACH(2)
      QLIN=(SHQ(III,JJJ)+SHQC(III,JJJ))/(PERLS*H1) * C*0.5
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  07.11.1979 CCCCCCCC
      PHWCTE=(QROD+QLIN)*(PHWC1L+PHWC2L)/QROD
      QROD1=QROD*PHWC1L/(PHWC1L+PHWC2L)
```

```
      PHWC1E=(QROD1+QLIN)/QROD1*PHWC1L
 4444 FPROV=(DET*FDTIP(2))**2*AT*FATIP(2)/PHWCTE
      WFCO1(III,JJJ)=AKAWC(1)*PHWC1E*FPROV/(AWC1L*DEWC1L**2)
      WFCO(III,JJJ)=(WFCO1(III,JJJ)*PHWC1L+AKAWC(2)*PHWC2L**2*FPROV/
     /(AWC2L*DEWC2L**2))/((PHWC1L+PHWC2L)*BKAPPA(IPA,2))
       WFCO1(III,JJJ)=WFCO1(III,JJJ)/BKAPPA(IPA,2)
      RELA=RELAM(AT*FATIP(2),DET*FDTIP(2),PB,TSCH(NS,JJJ),TW(NS,JJJ),
     *           MSCH(NS,JJJ),TLINER(III,JJJ),2,R1DR2L,PHWCTL/(PHWCTL+
     +PHWC2L))
      LAMLAM=AKAPPA(NS)/RELA
      CALL ENTRFR(K,1,2,RTIP(IPA),RO2WA,R21WA,NS,III,JJJ,DEWC1L,AWC1L,
     *           MWC1L,PB,TSCH(NS,JJJ),LAMLAM)
      CALL ENTRFR(K,2,2,RTIP(IPA),RO2WA,R22WA,NS,III,JJJ,DEWC2L,AWC2L,
     *           MWC2L,PB,TSCH(NS,JJJ),LAMLAM)
      IF(                I2TIP(NS,JJJ).EQ.1)GOTO 2997
C     ...............................................................
C     I3TIP=3: SAGAPO DECIDES WHETHER THE FLOW IS LAMINAR OR TURBULENT
C
      IF(LAMSCH(NS,JJJ).GT.LAMLAM)GOTO 3000
C     ...............................................................
C     THE FLOW IS LAMINAR
C
 2997 CONTINUE
      LAMSCH(NS,JJJ)=LAMLAM
      LAMWC(III,JJJ,1)=LAMLAM
      LAMWC(III,JJJ,2)=LAMLAM
      DDDD=AT*FATIP(2)/SQRT(LAMLAM*H/(2.*DET*FDTIP(2)*
     *RHO(PB,TSCH(NS,JJJ))))
      AMT=MSCH(NS,JJJ)
      ATSCH=TSCH(NS,JJJ)
      I2TIP(NS,JJJ)=1
      F2ATIP(NS,JJJ)=FATIP(2)
      F2DTIP(NS,JJJ)=FDTIP(2)
      ASCHWC(III,JJJ,1)=AWC1L
      ASCHWC(III,JJJ,2)=AWC2L
      PHWC(III,JJJ,1)=(PIG*0.5-OMEGA)*D*0.5
      PHWC(III,JJJ,2)=OMEGA*D*0.5
      DEWC(III,JJJ,1)=DEWC1L
      DEWC(III,JJJ,2)=DEWC2L
      HPLUSB(NS,JJJ)=1.
      HPLUSW(NS,JJJ)=1.
      QPLUS(NS,JJJ)=1.
      PRB(NS,JJJ)=1.
      YODH(NS,JJJ)=1.
      TBSSC1( NS,JJJ)=TSCH(NS,JJJ)
      T1SSC1(III,JJJ)=TSCH(NS,JJJ)
      T2SSC1(III,JJJ)=TSCH(NS,JJJ)
      TBSSC2(NS ,JJJ)=TSCH(NS,JJJ)
      T1SSC2(III,JJJ)=TSCH(NS,JJJ)
      T2SSC2(III,JJJ)=TSCH(NS,JJJ)
      TWSSC1( NS,JJJ)=TW(NS,JJJ)
      TWSSC2( NS,JJJ)=TW(NS,JJJ)
      XMSCHA(III,JJJ)=MSCH(NS,JJJ)*ASCHWC(III,JJJ,1)/(ASCH(NS,JJJ)*
     *F2ATIP(NS,JJJ))*0.5
      XMSCHB(III,JJJ)=XMSCHA(III,JJJ)
      ADAB(III,JJJ)=2.
C     FOR LAMINAR AND TURBULENT FLOW
C
 3000 CONTINUE
      FCOPW2(III,JJJ)=FCOPW1(2)+PHWC(III,JJJ,2)/PHWCTL*(FCOPW1(2)-1.)
      F1ATIP(NS)=F1ATIP(NS)+ASCH(NS,JJJ)/ACHA(NS)*F2ATIP(NS,JJJ)
       F1PTIP=F1PTIP+ASCH(NS,JJJ)/ACHA(NS)*F2ATIP(NS,JJJ)/F2DTIP(NS,JJJ)
      F1DTIP(NS)=F1ATIP(NS)/F1PTIP
      IF(IRH.EQ.1 .OR.     I2TIP(NS,JJJ).EQ.1)RETURN
C     ...............................................................
C     ONLY FOR TURBULENT FLOW AND ROUGHENED RODS
```

```
C
      ATBC=ATB+ASCHWC(III,JJJ,2)
      HPLUSB(NS,JJJ)=HPLUSB(NS,JJJ)/ATBC
      HPLUSW(NS,JJJ)=HPLUSW(NS,JJJ)/ATBC
      AMTBC=AMT-SRAMIA
      TSCHBC=(TT-TTA)/AMTBC
      CPTBC=CP(PB,TSCHBC)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     QPLUS(NS,JJJ)=QQ(NS,JJJ)*ATBC/(SUR*AMTBC*CPTBC*(TE+273.16))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      QAD=QQ(NS,JJJ)
      JP=JPIN(NS,JJJ)
      IF (QDEV .GT. 0.0) QAD=QQ(NS,JJJ) + QJ(JP,NS)/QDEV*4./H1
      QPLUS(NS,JJJ)=QAD*ATBC/(SUR*AMTBC*CPTBC*(TE+273.16))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      PRB(NS,JJJ)=ETA(PB,ATSCH)*CP(PB,ATSCH)/KAPPA(PB,ATSCH)
      YODH(NS,JJJ)=0.5*(SQRT(D**2+16.*ATBC/PIG)-D)/RH
      RETURN
  777 RETURN 1
      END
```

```
      SUBROUTINE RECCA2  (NS,III,NP,INDSP,H,LENGTH,PR1,PR2,PBT,FRELI,FT,
     *ITCORR,PIG,D,DPAV,*,WSP,I1SPAC)
C-----------------------------------------------------------------------
C     SUBROUTINE SUBBAL EVALUATES MASS FLOW RATES AND TEMPERATURES FOR
C     THE TWO PARTS OF WALL SUBCHANNEL
C
      REAL LENGTH,LAMWC,MAWC,M1WC(2),M2WC(2),MSCWC1,MSCH,MSCH1,MAVCF(2),
     *     MAV,MAVJT
      DIMENSION WCFUD(2),WCFWC(2),WCF1WC(2),EP1WC(2),QWCL(2),T1WC(2),
     1          TAWC(2),T2WC(2),RHO1(2),RHOAV(2),RUASWC(2),AWC(2),
     2          TMOEX(2),TACF(2),UACF(2),ACF(2),WTWC2(2),WTWC3(2)
     3          ,XMEM(2),DELTAA(2),IPAWC(2),QLINWC(2),THEX(2),DPWC(2),
     4          UWC(2)
      COMMON /MOB8/  DP( 42)
      COMMON /SUBC2/ JCHC(3,2)
      COMMON /SUB1/  ASCH( 42,3)
      COMMON /SUB2/  TSCH( 42,3),MSCH( 42,3)
      COMMON /CORR1/ SIGMAI( 42,3),PHII( 42,3)
      COMMON /CORR2/ CHI( 18,2,2),PSI( 18,2,2)
     1      /GRID0/CSPAC( 42,3,4)/IJ1/NER( 42),NIS( 42,3)/IND3/NTYP( 42)
     2      /GEN2/A( 42)/MOB2/UAV( 42)/MOB5/TAV( 42)/MOB6/MAV( 42)
     4      /SUB8/MSCH1( 42,3)/SUB31/WCFNS(3),DPNS(3),WTNS1(3,
     5      3),WTNS2(3,2),UNS(3),RUASNS(3)/HEA10/QSCH( 42,3)
     6      /WCSE1/DEWC( 18,2,2),PHWC( 18,2,2)
     A      /WCSE2/MSCWC1( 18,2,2)/WCSE5/
     7      TSCWC1( 18,2,2)/ WCSE3/LAMWC( 18,2,2)/WCSE4/CTURB( 18,2)
     8      /WCSE6/ASCWC1( 18,2,2)/WCSE7/MAWC( 18,2,2)
     9      /WCSE8/ASCHWC( 18,2,2)/WCSE9/TAVWC( 18,2,2)
      COMMON /SUBC1/NCHC(3),JSCH(3,3)/GEO0/ACH(3)
     1      /GRID1/EPS( 42,3,5),DISTS(7)/GRID8/PGDP( 42,3,4)
     2      /SUB3/ADAB( 18,2),DDBB( 18,2)
     >      /WACO1/XMSCHB( 18,2),XMSCHA( 18,2)
     3      /QPAR1/QDEV/QPAR2/QLINM,QLDEV/QPAR3/PERL(3)
     4      /GRIDWC/EPSWC( 18,2,2,4),CSPWC( 18,2,2,4)/GRAV/IGRAV
     5      /GAAG2/FCOPW2( 18,2)
     6      /ENEOP/IENE/GRID2/YY(100, 42,3)/MIXS2/CY/SECIN/K
      COMMON /HEA6 / NPIN( 42),JPIN( 42,3)
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC07C/ H1
      COMMON /SC13C/ GEO1( 42,3)
      COMMON /SC06L/ SHQ ( 18,2)
      COMMON /SC21C/ SHQC( 18,2)
      COMMON /SC09R/ QSR ( 18,2)
      XX=1./980665.
      DO 70 I=1,NP
      FRELWC=FRELI
      NCHCI=NCHC(I)
C
C     IW IS THE OTHER SUBCHANNEL OF WALL CHANNEL NS; NCHCI IS THE NUMBER
C     OF CHANNELS CONNECTED TO SUBCHANNEL I
C
      IW=3-I
C
C     PORTION 1 IS CONNECTED TO AN EXTERNAL CHANNEL; PORTION 2 TO A
C     CENTRAL CHANNEL ( PORTION INDEX = IPAWC )
C
      DO 101 K1=1,NCHCI
      JCHCIK=JCHC(I,K1)
      J=NIS(NS,JCHCIK)
      IPAWC(K1)=3-NTYP(J)+NTYP(J)/3
  101 CONTINUE
```

```
      DO 3 JWC=1,2
      WCFUD(JWC)=WCFNS(I)*ASCHWC(III,I,JWC)/ASCH(NS,I)
      MIWC(JWC)=MSCWC1(III,I,JWC)
      AWC(JWC)=ASCHWC(III,I,JWC)
      DELTAA(JWC)=0.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     NEW
C     IN THE DEFITION OF QWCL QSCHT HAS REPLACED QSCH(NS,I)
      IPIN=JPIN(NS,I)
      QSCHT=QSCH(NS,I)*QDEV/(LENGTH*0.25*PIG*D)+QJ(IPIN,NS)/
     >      (PIG*D*H/GEO1(NS,I))
      QWCL(JWC)=QSCHT*PHWC(III,I,JWC)
C     QWCL(JWC)=QSCHT*PHWC(III,I,JWC)/(LENGTH*0.25*PIG*D)*QDEV
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 07.11.1979 CCCCC
      QLINWC(3-JWC)=(SHQ(III,I)+SHQC(III,I))/H
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      TIWC(JWC)=TSCWC1(III,I,JWC)
      RHO1(JWC)=RHO(PR1,TSCWC1(III,I,JWC))
    3 CONTINUE
C     ..................................................................
C     ITERATION ON THE RELAXATION FACTOR (LOOP ITFREL)
C
      DO 50 ITFREL=1,98
      IVIA=1
C     ..................................................................
C     CALCULATION OF THE PRESSURE LOSSES (LOOP ITGL)
C
      DO 49 ITGL=1,60
C
C-----EVALUATION OF THE CROSS FLOW SOLUTION
      CALL CRFL1(ITGL,DPWCAV,FRELWC,ASCH(NS,I),2,AWC,MIWC,DPWC,WCFWC,
     *    WCF1WC,EP1WC)
      DO 5 JWC=1,2
      WCFWC(JWC)=WCFWC(JWC)+WCFUD(JWC)
      M2WC(JWC)=MIWC(JWC)-H*WCFWC(JWC)
      MAWC(III,I,JWC)=(M2WC(JWC)+MIWC(JWC))*0.5
      TAWC(JWC)=TSCH(NS,I)
      RUASWC(JWC)=MAWC(III,I,JWC)*SQRT(LAMWC(III,I,JWC)*0.125)
    5 CONTINUE
C
      IF(ITGL.GT.1 .AND. IVIA.EQ.1 )GOTO 30
C     ..................................................................
C     CALCULATION OF THE AVERAGE GAS TEMPERATURES (LOOP ITERM)     .
C
      XPREC=1.E-04
      DO 25 ITERM=1,20
C
C  A) TURBOLENT EXCHANGE BETWEEN THE TWO PARTS OF SUBCHANNEL
C
      IF(TAWC(1).LE.0. .OR. TAWC(1).GT.3000. .OR. TAWC(2).LE.0. .OR.
     * TAWC(2).GT.3000.)GOTO 99
      YYWC=(YY(K,NS,I)-1.)*CY+1.
      WTWC1=TME(PBT,MAWC(III,I,1),MAWC(III,I,2),TAWC(1),TAWC(2),LAMWC(II
     *I,I,1),LAMWC(III,I,2),AWC(1),AWC(2),CTURB(III,I))*YYWC
      TA12=(MAWC(III,I,1)*TAWC(1)+MAWC(III,I,2)*TAWC(2))/MSCH(NS,I)
      THEX(1)=-(TAWC(1)-TAWC(2))*WTWC1*CP(PBT,TA12)
      THEX(2)=-THEX(1)
C
C  B) TURBOLENT EXCHANGE WITH CHANNELS
C
      DO 8 K1=1,NCHCI
      IWC=IPAWC(K1)
      JCHCIK=JCHC(I,K1)
      J=NIS(NS,JCHCIK)
      NTYPJ=NTYP(J)
```

```
      MAVJT=MAV(J)*ACH(NTYPJ)/A(J)
      WTWC2(IWC)=WTNS2(I,K1)
      TAIJ=(TAWC(IWC)*MAWC(III,I,IWC)+TAV(J)*MAVJT )/(MAWC(III,I,IWC)+
     *      MAVJT )
      THEX(IWC)=THEX(IWC)-(TAWC(IWC)-TAV(J))*WTWC2(IWC)*CP(PBT,TAIJ)
    8 CONTINUE
      IF(NP.EQ.1)GOTO 11
C
C  C) TURBOLENT EXCHANGE WITH THE OTHER SUBCHANNEL
C
      SRUAS=RUASWC(1)+RUASWC(2)+2.*RUASNS(IW)
      DO 10 JWC=1,2
      WTWC3(JWC)=WTNS1(1,2)*(RUASWC(JWC)+RUASNS(IW))/SRUAS
      TAIJ=(TAWC(JWC)*MAWC(III,I,JWC)+TSCH(NS,IW)*MSCH(NS,IW))/
     *    (MAWC(III,I,JWC)+MSCH(NS,IW))
      THEX(JWC)=THEX(JWC)-(TAWC(JWC)-TSCH(NS,IW))*WTWC3(JWC)*CP(PBT,TAIJ
     *)
   10 CONTINUE
   11 CONTINUE
C
C  D) CROSS FLOW EXCHANGE BETWEEN THE TWO PARTS OF SUBCHANNEL
C
      TACF(1)=0.
      MAVCF(1)=0.
      CALL CF1(TAWC(1),TAWC(2),MAWC(III,I,1),MAWC(III,I,2),DPWC(1),
     *DPWC(2),ITGL,TACF(1),MAVCF(1))
      TACF(2)=TACF(1)
      MAVCF(2)=MAVCF(1)
C
C  E) CROSS FLOW EXCHANGE WITH CHANNELS
C
      DO 16 K1=1,NCHCI
      IWC=IPAWC(K1)
      JCHCIK=JCHC(I,K1)
      J=NIS(NS,JCHCIK)
      NTYPJ=NTYP(J)
      MAVJT=MAV(J)*ACH(NTYPJ)/A(J)
      CALL CF1(TAWC(IWC),TAV(J),MAWC(III,I,IWC),MAVJT ,DPWC(IWC),DP(J),
     *ITGL,TACF(IWC),MAVCF(IWC))
   16 CONTINUE
      IF(NP.EQ.1)GOTO  18
C
C  F) CROSS FLOW EXCHANGE WITH THE OTHER SUBCHANNEL
C
      DO 17 JWC=1,2
      CALL CF1(TAWC(JWC),TSCH(NS,IW),MAWC(III,I,JWC),MSCH(NS,IW),
     *DPWC(JWC),DPNS(IW),ITGL,TACF(JWC),MAVCF(JWC))
   17 CONTINUE
   18 CONTINUE
      DO 20 JWC=1,2
      TACF(JWC)=TACF(JWC)/MAVCF(JWC)
      TAICF=( TAWC(JWC)*MAWC(III,I,JWC)+TACF(JWC)*MAVCF(JWC))/(MAWC(III,
     *I,JWC)+MAVCF(JWC))
      CFHEX=WCFWC(JWC)*(TAWC(JWC)-TACF(JWC))*CP(PBT,TAICF)
       XXMAV=MAWC(III,I,JWC)
      XXM2=M2WC(JWC)
      IF(IENE.EQ.2)XXMAV=MIWC(JWC)
      IF(IENE.EQ.2)XXM2=XXMAV
      T2WC(JWC)=TSCWC1(III,I,JWC)+H/(XXMAV*CP(PBT,TAWC(JWC)))*
     *           (QWCL(JWC)+QLINWC(JWC)+THEX(JWC)+CFHEX)
      IF(ABS(PHII(NS,I)).GT.1.E-20)GOTO 200
      PSI(III,I,JWC)=1.
      GOTO 201
  200 CONTINUE
      PSI(III,I,JWC)=(THEX(JWC)+CFHEX)*H/(AWC(JWC)*PHII(NS,I))
  201 CONTINUE
```

```
      TAVWC(III,I,JWC)=(XXM2*T2WC(JWC)+MIWC(JWC)*TSCWC1(III,I,JWC))
     *              *0.5  /XXMAV
   20 CONTINUE
      IF(ITGL.EQ.1)GOTO 30
      IF(ITERM.GT.10)XPREC=1.E-03
      IF(ITERM.GT.15)XPREC=1.E-02
      DO 21 JWC=1,2
      IF(ABS(TAWC(JWC)/TAVWC(III,I,JWC)-1.).GT.XPREC)GOTO 22
   21 CONTINUE
      GOTO 30
   22 CONTINUE
      DO 23 JWC=1,2
   23 TAWC(JWC)=TAVWC(III,I,JWC)
   25 CONTINUE
C     ..................................................................
C     END OF THE LOOP ITERM: POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
      WRITE(6,26)NS,I,(TAWC(JWC),JWC=1,2),ITCORR
   26 FORMAT(     5X,'STOP IN LOOP ITERM OF SUB. RECCA2. NS=',I5,2X,'I=',
     *I2,5X,'TEMPERATURES='/5X,2E15.7/5X,'ITCORR=',I5)
      RETURN 1
C     ..................................................................
C     CONVERGENCE HAS BEEN REACHED FOR THE ENERGY EQUATIONS; THE CALCU=
C     LATION OF THE PRESSURE DROPS STARTS
C
   30 CONTINUE
      DO 31 JWC=1,2
      RHOAV(JWC)=RHO(PBT,TAVWC(III,I,JWC))
      UWC(JWC)=MAWC(III,I,JWC)/(AWC(JWC)*RHOAV(JWC))
   31 CONTINUE
      DPWCAV=0.
      SMWC1=0.
C
C  A) TURBOLENT EXCHANGE BETWEEN THE TWO PARTS OF SUBCHANNEL
C
      TMOEX(1)=-(UWC(1)-UWC(2))*WTWC1
      TMOEX(2)=-TMOEX(1)
C
C  B) TURBOLENT EXCHANGE WITH CHANNELS
C
      DO 35 K1=1,NCHCI
      JCHCIK=JCHC(I,K1)
      J=NIS(NS,JCHCIK)
      IWC=IPAWC(K1)
      TMOEX(IWC)=TMOEX(IWC)-(UWC(IWC)-UAV(J))*WTWC2(IWC)
   35 CONTINUE
C
C  C) TURBOLENT EXCHANGE WITH THE OTHER SUBCHANNEL
C
      DO 37 JWC=1,2
      IF(NP.NE.1)TMOEX(JWC)=TMOEX(JWC)-(UWC(JWC)-UNS(IW))*WTWC3(JWC)
   37 TMOEX(JWC)=TMOEX(JWC)*FT*H/AWC(JWC)
      UACF(1)=0.
      ACF(1)=0.
C
C  D) CROSS FLOW EXCHANGE BETWEEN THE TWO PARTS OF SUBCHANNEL
C
      CALL CF1(UWC(1),UWC(2),AWC(1),AWC(2),DPWC(1),DPWC(2),    1,UACF(1),
     *         ACF(1))
      UACF(2)=UACF(1)
      ACF(2)=ACF(1)
C
C  E) CROSS FLOW EXCHANGE WITH CHANNELS
C
```

```
      DO 40 K1=1,NCHCI
      IWC=IPAWC(K1)
      JCHCIK=JCHC(I,K1)
      J=NIS(NS,JCHCIK)
      NTYPJ=NTYP(J)
      AJT=ACH(NTYPJ)
      CALL CF1(UWC(IWC),UAV(J),AWC(IWC),AJT ,DPWC(IWC),DP(J),1,
     *         UACF(IWC),ACF(IWC))
   40 CONTINUE
      DO 45 JWC=1,2
C
C  F) CROSS FLOW EXCHANGE WITH THE OTHER SUBCHANNEL
C
      IF(NP.NE.1)      CALL CF1(UWC(JWC),UNS(IW),AWC(JWC),ASCH(NS,IW),
     *     DPWC(JWC),DPNS(IW),1    ,UACF(JWC),ACF(JWC))
C
C
      UACF(JWC)=UACF(JWC)/ACF(JWC)
      CFMOEX=(2.*UWC(JWC)-UACF(JWC))*WCFWC(JWC)*H/AWC(JWC)
      XMEM(JWC)=LAMWC(III,I,JWC)*H/(2.*DEWC(III,I,JWC)*RHOAV(JWC))
      IF(JWC.EQ.1)XMEM(JWC)=XMEM(JWC)*FCOPW2(III,I)
      RE=MAWC(III,I,JWC)*DEWC(III,I,JWC)/(AWC(JWC)*ETA(PBT,TAVWC(III,I,
     1JWC)))
      IF(INDSP.EQ.2)XMEM(JWC)=XMEM(JWC)+(CSPWC(III,I,JWC,I1SPAC)+DSPDPF(
     *EPSWC(III,I,JWC,I1SPAC),DEWC(III,I,JWC),LAMWC(III,I,JWC),WSP,
     *PGDP(NS,I,I1SPAC),RE,2))/RHOAV(JWC)
      DPWC(JWC)=XX*(-(MAWC(III,I,JWC)/AWC(JWC))**2*(XMEM(JWC)-(RHO(PR2,
     *        T2WC(JWC))-RHO1(JWC))/RHOAV(JWC)**2-DELTAA(JWC)/(AWC(JWC)*
     *    RHOAV(JWC)))+TMOEX(JWC)+CFMOEX+IGRAV*980.665*RHOAV(JWC)*H)
      DPWCAV=DPWCAV+DPWC(JWC)*MIWC(JWC)
      SMWC1=SMWC1+MIWC(JWC)
   45 CONTINUE
      DPWCAV=DPWCAV/SMWC1
C     ..................................................................
C     TEST OF CONVERGENCE ON THE PRESSURE DROPS
C
      IF(ITGL.LT.4)GOTO 47
      DO 46 JWC=1,2
      IF(ABS(DPWC(JWC)/DPWCAV-1.).GT.1.E-02)GOTO 47
      IF(ABS(DPWC(JWC)/DPWCAV-1.).GT.1.E-03 .AND. ITGL.LT.40)GOTO 47
   46 CONTINUE
      IF(IVIA.EQ.2)GOTO 55
      IF(M2WC(1).LE.0. .OR. M2WC(2).LE.0.)GOTO 99
      IVIA=2
   47 CONTINUE
      DO 48 JWC=1,2
   48 WCFWC(JWC)=WCFWC(JWC)-WCFUD(JWC)
   49 CONTINUE
C     ..................................................................
C     END OF LOOP ITGL
C
   99 CONTINUE
      AIT=ITFREL
      FRELWC=1.-AIT*0.01
   50 CONTINUE
C     ..................................................................
C     END OF LOOP ITFREL: POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
      WRITE(6,51)ITCORR,NS,I,(DPWC(JWC),JWC=1,2),(MAWC(III,I,JWC),JWC=1,
     *           2),(TAVWC(III,I,JWC),JWC=1,2),(AWC(JWC),JWC=1,2)
   51 FORMAT(//  5X,'STOP IN LOOP ITGL OF RECCA2: ITCORR=',I5,5X,'NS=',
     1I5,5X,'I=',I2/5X,'PRESSURE LOSSES:',2E15.5/5X,'AVERAGE MASSES:',
     22E15.5/5X,'AVERAGE TEMPERATURES:',2E15.5/5X,'AREAS:',2E15.5)
      RETURN 1
C     ..................................................................
```

```
C     THE ENERGY EQUATIONS AND THE AXIAL MOMENTUM EQUATIONS HAVE
C     REACHED CONVERGENCE
C
   55 CONTINUE
      DO 56 JWC=1,2
      DPAVF=DPAV-IGRAV*RHOAV(JWC)*H*0.001
      BMWC=SQRT(ABS(DPAVF) /(XX*XMEM(JWC)))*AWC(JWC)
      CHI(III,I,JWC)=(MAWC(III,I,JWC)-BMWC)/(AWC(JWC)*SIGMAI(NS,I))
   56 CONTINUE
      EPSM=MAWC(III,I,1)-(XMSCHA(III,I)+XMSCHB(III,I))
      XMSCHA(III,I)=XMSCHA(III,I)+EPSM*(1.-1./ADAB(III,I))
      XMSCHB(III,I)=XMSCHB(III,I)+EPSM/ADAB(III,I)
   70 CONTINUE
      RETURN
      END
```

```
      FUNCTION RELAM(A,D,P,TB,TW,M,TLINER,ITYP,R1DR2L,PH1DPH)
C     -----------------------------------------------------------------
C     RELAM COMPUTES THE LAMINAR REYNOLDS NUMBERS FOR THE CALCULATION
C     OF THE SUBCHANNEL FRICTION FACTORS
C
      REAL M
      COMMON /GAAG1/ FCOPW1(3)
      COMMON/INPAR/IPA/LAMIN5/RTIP(7)/DAT/PIG/QPAR3/PERL(3)/MART/ITCORR
     1      /RETEM/TNY
      TL=TLINER
      IF(IPA/2*2.NE.IPA .OR. ITCORR.EQ.1)TW=TB
CCCCCCCCCCCCCCCCCCCCCCCCCCC  16.10.1979
      RENU =M*D/FCOPW1(ITYP) /(A*RHO(P,TB))
      PERLC=PERL(ITYP)*FCOPW1(ITYP)
CCCCCCCCCCCCCCCCCCCCCCCCCCC
C     RENU =M*D /(A*RHO(P,TB))
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      TNY=TW
      IF(ITYP.NE.1 .AND. IPA/2*2.EQ.IPA .AND. ITCORR.GT.1)
     *             TNY=TNU(TW,TL,ITYP,PERLC      ,PIG,RTIP(IPA))
CCCCCCCCCCCCCCCCCCCCCCCCCCC  16.10.1979
C    *             TNY=TNU(TW,TL,ITYP,PERL(ITYP),PIG,RTIP(IPA))
CCCCCCCCCCCCCCCCCCCCCCCCCCC
      RELAM=RENU *RHO(P,TNY)/ETA(P,TNY)
C
      RETURN
      END
```

```
      FUNCTION RHO(P,T)
C-----------------------------------------------------------------------
C     RHO EVALUATES THE DENSITY OF THE COOLANT (G/CCM)
C
      COMMON/GASD4/IGAS
      GOTO(10,20,30,40),IGAS
   10 CONTINUE
C     CASE OF HELIUM COOLANT
C
      TODT=273.16/(273.16+T)
      RHO=0.172823E-03*P*TODT-0.904002E-07*P**2*TODT**2.2
      RETURN
C
   20 CONTINUE
C     CASE OF CO2 COOLANT
C
      PP=P
      TT=T
      P=PP/1.0333
      T=TT+273.16
      TO=273.16
      IF(T-516.) 1, 1, 2
    1 CK=.0134
      GO TO 5
    2 IF(T-750.) 3, 4, 4
    3 CK=(650.-T)* 1.E-4
      GO TO 5
    4 CK=-.01
    5 TF=TO/T
      ROF=1.9635*P*TF*(1.+CK*P*(TF**5))
      RHO=ROF*0.001
      T=TT
      P=PP
      RETURN
C
   30 CONTINUE
C     CASE OF N2 COOLANT.
C
      TT=T
      T=TT+273.16
      RHO=1.2499*P*273.16/1.033/T*(1.-4.E-4*(P/1.033)*
     >    (1.-.05*ABS((600.-T)/273.16)**.61))*.001
      T=TT
C
      RETURN
   40 CONTINUE
      RHO=0.
      RETURN
      END
```

```
      FUNCTION RHPLUS(HPLUSB,TW,TE,QPLUS,HPLUSW,TB1,YDH)
C-----------------------------------------------------------------
C
C     RHPLUS EVALUATES THE FUNCTION R(H+)
C     IRHPL=1 : R(H+)=R(HW+)+CONST/(HW+)**CONST*(TW/TB1-1)**CONST+
C                    +CONST*ALOG(HR/(0.01*(R0-R1)))
C     IRHPL=2 : R(H+)=R(HB+) (FOR THE LAST UNHEATED ROUGH PART)
C
      COMMON/DAT2/B1,B2,B3,B4,B5,B6,B7,B8,B9,B10/TRANS/RHTU,RHSM
     1       /DAT6/IRHPL
      CORRTW=0.
      GOTO(1,2),IRHPL
    1 HPLUS=HPLUSW
      CTW=(TW+273.16)/(TB1+273.16)-1.
      IF(CTW.GT.0.)CORRTW=CTW**B10
      GOTO 3
    2 HPLUS=HPLUSB
    3 RHPL  =(B1+B2/HPLUS**B3)**B4+B5*ALOG(1./(YDH*B6))+B8/HPLUS**B9*
     * CORRTW
      RHTU=RHPL
CCCCCCCCCCCCCCCCCC 12.10.1979
C     RHSM=5.5+2.5*ALOG(HPLUSB)
CCCCCCCCCCCCCCCCCC
      RHSM=5.5+2.5*ALOG(HPLUS)
CCCCCCCCCCCCCCCCCC
C
C     IF R(H+) TURB. >RHSM THE FLOW IS "HYDRAULICALLY SMOOTH"
C
      IF(RHPL.GT.RHSM)RHPL=RHSM
      RHPLUS=RHPL
      RETURN
      END
```

```
      SUBROUTINE RNU(HPLUSW,TWI,LAMIB,REI,PRI,TBT,YDH,R1DR2,R2MROH,U1DU,
     *REW,YYI,NUI,GHPL,ITYP,R0,R1,R2,BK)
C-----------------------------------------------------------------
C     RNU EVALUATES NUSSELT NUMBER IN THE ROUGH PART
C
      REAL LAMIB, NUI
      GHPL=GHPLUS(HPLUSW,TWI,TBT,PRI,YDH,REW,R2MROH,R2,R1)
      FF=GHPL+2.5*ALOG(YDH+R2MROH)-(1.25+3.75*R1DR2)/(1.+R1DR2)
      STI=SQRT(LAMIB*0.125)*U1DU/FF
      NUI=STI*REI*PRI*YYI
      RETURN
      END
```

```
      SUBROUTINE RTRI(PBT,TBT,MASSI,DE1,AREAI,ADAB,LAM1,QA,FACHE,TE,
     * RH,I,II,M,JPIN,TW1,RU1DRU,ITYP,DEI,D,YYDH,*,F2ATIP,F2DTIP,QALIN)
C-----------------------------------------------------------------------
C     RTRI EVALUATES ROD TEMPERATURES FOR CENTRAL AND CORNER SUBCHANNELS
C     AND FOR THE TWO PARTS OF WALL SUBCHANNELS IN THE ROUGH PORTION.THE
C     BULK TEMPERATURES OF THE TWO REGIONS DEFINED BY THE TAU=0 LINE ARE
C     ALSO COMPUTED.
C
      REAL LAM1,MASSI,KI,KAPPA,NUI,NUO,NUTU
      COMMON/SUB21/TSCHA( 18,2),TSCHB( 18,2)/SHROUD/TLINER( 18,2)
     1                  /TRANS/RHTU,RHSM/LAMINO/I2TIP( 42,3)/ISUP/IQLIN
      COMMON /MART5/ NSTR
      COMMON /SC15C/ ALFA( 42,3)
      COMMON /SC10C/ ANU( 42,3)
      COMMON /SC32C/ GHPIU( 42,3)
      COMMON /SC01Z/ YH( 42,3)
      COMMON /SC34C/ ISUC
C     ..................................................................
C     TEMLAM IS CALLED IF THE FLOW IS LAMINAR; THE CALCULATION RETURNS
C     THEN AT THE END OF RTRI
C
      IF(I2TIP(I,M).EQ.1)CALL TEMLAM(&2000,PBT,TBT,MASSI,DEI,AREAI,QA,
     &                    QALIN,TE,I,II,M,TW1,ITYP,F2ATIP,F2DTIP,D)
C     ******************************************************************
C     THE FLOW IS TURBULENT: CALCULATION PERFORMED ASSUMING ROUGH FLOW
C     ******************************************************************
C
      R1=D*0.5
      R0=0.5*SQRT(D**2+DE1*D)
      R2=SQRT(D**2+ADAB*DE1*D)*0.5
C     ..................................................................
C     INLET EFFECT ON THE NUSSELT NUMBER OF THE RODS
C
      FACHE=TIS(R1,R2,2)
C
      YDH=(R0-R1)/RH
      R2MR0H=(R2-R0)/RH
      YYDH=YDH+R2MR0H
      R0DR2=R0/R2
      R1DR2=R1/R2
      KI=KAPPA(PBT,TBT)
      ETAI=ETA(PBT,TBT)
      RHOI=RHO(PBT,TBT)
      CPI=CP(PBT,TBT)
      REI=MASSI*DEI/(AREAI*ETAI)
      PRI=ETAI*CPI/KI
      UI=MASSI/(AREAI*RHOI)
      TWALL=TBT
      TW0=TBT
      TB1=TBT
C     ..................................................................
C     CALCULATION OF THE BULK TEMPERATURES OF THE TWO ZONES DIVIDED BY
C     THE TAU=0 LINE ( LOOP ITW )
C
      DO 7 ITW=1,20
      RHO1=RHO(PBT,TB1)
      U1DU=RU1DRU*RHOI/RHO1
      U1=U1DU*UI
      U1STAR=U1*SQRT(LAM1*0.125)
C     ..................................................................
C     CALCULATION OF THE SURFACE PIN TEMPERATURE AT INFINITE CONDUCTI=
```

```
C     VITY OF THE CANNING METAL AND AT (Q'' )SHROUD = 0 ( LOOP ITW1 )
C
      DO 30 ITW1=1,30
      IF(ABS(TWO).LT.3000..AND. ABS(TWALL).LT.3000.)GOTO 29
      WRITE(6,28)I,JPIN,TWO,TWALL
   28 FORMAT(    5X,'STOP IN RTRI: NS=',I5,5X,'PIN=',I5/5X,'TWO=',E15.5,
     15X,'TWALL=',E15.5)
      RETURN 1
   29 CONTINUE
      ETAW=ETA(PBT,TWALL)
      RHOW=RHO(PBT,TWALL)
      REW=U1*DE1*RHOW/ETAW
      REWO=REW*ETAW*RHO(PBT,TWO)/(RHOW*ETA(PBT,TWO))
      HPLUSW=RH*REW*SQRT(LAM1*0.125)/DE1
      HPLUSO=HPLUSW*REWO/REW
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      CALL SPANU(REI,PRI,I,M,YYI)
      BK=2.0
      IF(ITYP.EQ.1 .AND. I.GT.NSTR) BK=1.0
      CALL RNU(HPLUSW,TWALL,LAM1,REI,PRI,TBT,YDH,R1DR2,R2MROH,U1DU,REW,
     1     YYI,NUI,GHPL,ITYP,R0,R1,R2,BK)
      CALL RNU(HPLUSO,TWO   ,LAM1,REI,PRI,TBT,YDH,R1DR2,R2MROH,U1DU,REWO,
     1     1.,NUO,GHPO,ITYP,R0,R1,R2,BK)
C     18.03.1981
      IF(ITYP.NE.1 .OR. ISUC.EQ.0) GO TO 200
      CALL SUPCEN (RH,PBT,TWALL,LAM1,U1DU,YYI,ITYP,KI,DEI,FACHE,QA,TW1,
     >                GHPL,NUI,I,M,REI,ITW1)
  200 CONTINUE
C
      GHPIU(I,M)=GHPL
      ALFAI=NUI*KI/DEI*FACHE
      ALFAO=NUO*KI/DEI
      ALFA(I,M)=ALFAI
      ANU(I,M)=NUI
      YH(I,M)=YYI
      IF(ISUC.NE.1 .OR. ITYP.NE.1)TW1=TBT+QA/ALFAI
C     TW1=TBT+QA/ALFAI
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      TWO=TBT+QA/ALFAO
      IF(ABS(TW1/TWALL-1.).LE.1.E-04)GOTO 32
   30 TWALL=TW1
C     ...................................................................
C     END OF LOOP ITW1: POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
      WRITE(6,31)I,JPIN,TW1
   31 FORMAT(1H1,5X,'STOP IN RTRI (LOOP ITW1) NS=',I5,5X,'PIN=',I5,5X,
     *'TW1=',E15.5)
      RETURN 1
C     ...................................................................
C     CONVERGENCE HAS BEEN REACHED FOR THE PIN TEMPERATURE
C
   32 CONTINUE
      IF(ITYP.EQ.1)GOTO 9
C     ...................................................................
C     ONLY FOR THE CORNER CHANNELS AND FOR THE WALL PORTION OF THE WALL
C     SUBCHANNELS
C
      FF=QA/(RHOI*CPI*U1STAR)
      CALL DDONNE(TWO,TBT,GHPL,R0DR2,R1DR2,YDH,R2MROH,FF,TSCHA(II,M),
     1   TSCHB(II,M),TE)
      IF(ABS(TSCHB(II,M)/TB1-1.).LE.1.E-04)GOTO 9
      TB1=TSCHB(II,M)
    7 CONTINUE
C     ...................................................................
C     END OF LOOP ITW: POINT REACHED IN THE CASE OF CONVERGENCE
```

```
C     PROBLEMS
C
      WRITE(6,8)I,JPIN,TB1
    8 FORMAT(1H1,5X,'STOP IN RTRI (LOOP ITW) I=',I5,5X,'PIN=',I5,5X,'TB1
     $=',E15.5)
      RETURN 1
C     ..................................................................
C     CONVERGENCE HAS BEEN REACHED FOR THE BULK TEMPERATURES OF THE
C     TWO ZONES DIVIDED BY THE TAU=0 LINE; THE ASSUMPTION OF ROUGH FLOW
C     IF TESTED ( THIS POINT IS REACHED ALSO BY THE CALCULATION FOR THE
C     CENTRAL SUBCHANNELS AND THE CENTRAL PORTION OF THE WALL SUBCHANNEL
C
    9 CONTINUE
      ETA1=ETA(PBT,TB1)
      HPLUSB=HPLUSW*RHO1*ETAW/(ETA1*RHOW)
      RHPL=RHPLUS(HPLUSB,TWALL,TE,XYXYX,HPLUSW,TB1,YDH)
      IF(RHTU.LE.RHSM)GOTO 100
C     ******************************************************************
C     THE FLOW IF "HYDRAULICALLY" SMOOTH: THE CALCULATION IS REPEATED IN
C     SUBROUTINE RTSI. THE CALCULATION IS BASED STILL ON THE VOLUMETRIC
C     DIAMETER. THE CALCULATION RETURNS  IMMEDIATELY AFTER COMING BACK
C     FROM RTSI
C     ******************************************************************
      CALL RTSI(PBT,TBT,MASSI,DE1,AREAI,ADAB,LAM1,QA,FACHE,TE,RH,I,
     &II,M,JPIN,TW1,RU1DRU,ITYP,DEI,D,YYDH,&8500,F2ATIP,F2DTIP,QALIN)
      RETURN
C     ******************************************************************
C     POINT REACHED IN THE CASE OF ROUGH FLOW
C     ******************************************************************
  100 IF(ITYP.EQ.1)RETURN
C     ..................................................................
C     CALCULATION OF THE SHROUD TEMPERATURE  FOR THE CORNER CHANNELS AND
C     FOR THE WALL PORTION OF THE WALL SUBCHANNELS ( VALUE AT
C     (Q")SHROUD = 0 )
C
      TLINER(II,M)=TWO-FF*(2.5*ALOG((R2-R1)/RH)+GHPL)
      IF(TLINER(II,M).LE.TE)TLINER(II,M)=TE
C     ..................................................................
C     CORRECTION OF THE PREVIOUSLY COMPUTED PIN AND SHROUD TEMPERATURES
C     OF THE CORNER CHANNELS AND OF THE WALL PORTION OF THE WALL SUBCHA=
C     NNELS IN THE CASE OF HEATED SHROUD WALLS (SUPERPOSITION PRINCIPLE)
C
      DEIAN=2.*(R2-R1)
      TETA2=0.
      IF(QA.GT.1.E-06)TETA2=(TLINER(II,M)-TBT)*KI/(QA*DEIAN)
      NUI=NUI*DEIAN/DEI
CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ANU(I,M)=NUI
CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      REI=REI*DEIAN/DEI
      A1=0.45/(2.4+PRI)
      NUTU=TUBENU(REI,PRI)
      PEI=REI*PRI
      FTWA=22.*(0.27*R1DR2**2-1.)/(PEI**0.87*PRI**0.18)*R1DR2
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 22.05.1980
C     IF(IQLIN.EQ.1 .AND. ABS(QALIN)
C    1.GT.1.E-06)CALL TELIN(TW1,TLINER(II,M),TBT,TE,TETA2,FTWA,
C    1 QA,QALIN,NUI,NUTU,A1,KI,R1DR2,DEIAN,I,JPIN,YYI,FACHE,M)
      CALL TELIN(TW1,TLINER(II,M),TBT,TE,TETA2,FTWA,
     1 QA,QALIN,NUI,NUTU,A1,KI,R1DR2,DEIAN,I,JPIN,YYI,FACHE,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 22.05.1980
 2000 RETURN
 8500 RETURN 1
      END
```

```
      SUBROUTINE RTSI(PBT,TI,MASSI,DE1,AREAI,ADAB,LAM1,QA,FACHE,
     * TE,RH,I,II,M,JPIN,TWI,RU1DRU,ITYP,DEIR,D,XXXX,*,F2ATIP,F2DTIP,
     * QALIN)
C-----------------------------------------------------------------------
C     RTSI EVALUATES ROD TEMPERATURES FOR CENTRAL AND CORNER SUBCHANNELS
C     AND FOR THE TWO PARTS OF WALL SUBCHANNELS IN THE SMOOTH PART
C
      REAL NUTU,NUI0,NUI,KI,KAPPA,MASSI,LAM1
      COMMON/SUB21/TSCHA( 18,2),TSCHB( 18,2)/SHROUD/TLINER( 18,2)
     1                 /LAMINO/I2TIP( 42,3)/ISUP/IQLIN/ISMO/COTW
     2       /ISMO1/ITECO
C    1       /QSHR/QALIN/LAMINO/I2TIP( 42,3)/ISUP/IQLIN/ISMO/COTW
      COMMON /SC15C/ ALFA( 42,3)
      COMMON /SC10C/ ANU( 42,3)
      COMMON /SC01Z/ YH( 42,3)
C     ..................................................................
C     TEMLAM IS CALLED IF THE FLOW IS LAMINAR; THE CALCULATION RETURNS
C     THEN AT THE END OF RTSI
C
      IF(I2TIP(I,M).EQ.1)CALL TEMLAM(&2000,PBT,TI,MASSI,DEIR,AREAI,QA,
     &                  QALIN,TE,I,II,M,TWI,ITYP,F2ATIP,F2DTIP,D)
C     ******************************************************************
C     THE FLOW IS TURBULENT
C     ******************************************************************
      TG=TE
      IF(ITECO.EQ.2)TG=TI
       R1=D*0.5
      R0=0.5*SQRT(D**2+DE1*D)
      R2=SQRT(D**2+ADAB*DE1*D)*0.5
      DEI=2.*(R2-R1)
      R0DR2=R0/R2
      R1DR2=R1/R2
      R1DR0=R1/R0
C     ..................................................................
C     INLET EFFECT ON THE NUSSELT NUMBER OF THE RODS
C
      FACHE=TIS(R1,R2,1)
C
      TWI0=TWI
      KI=KAPPA(PBT,TI)
      ETAI=ETA(PBT,TI)
      RHOI=RHO(PBT,TI)
      CPI=CP(PBT,TI)
      REI=MASSI*DEI/(AREAI*ETAI)
      PRI=ETAI*CPI/KI
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      CALL SPANU(REI,PRI,I,M,YYI)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      A1=0.45/(2.4+PRI)
      A2=0.16*PRI**(-0.15)
      A3=1.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 21.09.1979
C     IF(R1DR2.LT.0.2)A3=1.+(7.5*(1./R1DR2-5.)/REI)**0.6
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF(R1DR2.LT.0.2)A3=1.+7.5*((1./R1DR2-5.)/REI)**0.6
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      NUTU=TUBENU(REI,PRI)
      FNU=(1.-A1)/R1DR2**A2  *A3*(TG+273.16)**COTW*NUTU
C     ..................................................................
C     CALCULATION OF THE SURFACE PIN TEMPERATURE AT (Q")SHROUD = 0
C     ( LOOP ITW )
```

```
C
      DO 5 ITW=1,10
      TWALL=TWI
      NUI=FNU/(TWI+273.16)**COTW*YYI*FACHE
      NUI0=FNU/(TWI0+273.16)**COTW
      ALFAI0=NUI0*KI/DEI
      ALFAI=NUI*KI/DEI
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ALFA(I,M)=ALFAI
      ANU(I,M)=NUI
      YH(I,M)=YYI
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      TWI=TI+QA/ALFAI
      TWI0=TI+QA/ALFAI0
      IF(ABS(TWALL/TWI-1.).LE.1.E-04)GOTO 7
    5 CONTINUE
C     .................................................................
C     END OF LOOP ITW: POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
      WRITE(6,6)I,JPIN,TWI,TWALL
    6 FORMAT(1H1,5X,'STOP IN RTSI (CHANNEL',I5,' , PIN',I5,') : TW=',
     *E15.7,5X,'TWALL=',E15.7)
      RETURN 1
C     .................................................................
C     CONVERGENCE HAS BEEN REACHED FOR THE PIN TEMPERATURE
C
    7 IF (ITYP.EQ.1)RETURN
C     .................................................................
C     CALCULATION OF THE SHROUD TEMPERATURE  FOR THE CORNER CHANNELS AND
C     FOR THE WALL PORTION OF THE WALL SUBCHANNELS ( VALUE AT
C     (Q")SHROUD = 0 )
C
      PEI=REI*PRI
      FTWA=22.*(0.27*R1DR2**2-1.)/(PEI**0.87*PRI**0.18)*R1DR2
      TLINER(II,M)=FTWA*QA*DEI/KI+TI
      IF(TLINER(II,M).LE.TE)TLINER(II,M)=TE
      TETA2=0.
      TSCHA(II,M)=TI
      TSCHB(II,M)=TI
      IF(QA.LE.1.E-06)GOTO 22
      TETA2=(TLINER(II,M)-TI)*KI/(QA*DEI)
      GTI=(1.5*R1DR2+0.5)/(R1DR2+1.)
      GT1=(1.5*R1DR0+0.5)/(R1DR0+1.)
      UI=MASSI/(AREAI*RHOI)
      F1=R0**2-R1**2
      F2=R2**2-R0**2
      FI=F1+F2
      TB1=TI
C     .................................................................
C     CALCULATION OF THE BULK TEMPERATURES OF THE TWO ZONES DIVIDED BY
C     THE TAU=0 LINE FOR THE CORNER CHANNELS AND FOR THE WALL PORTION OF
C     THE WALL SUBCHANNELS ( LOOP ITW1 )
C
      DO 20 ITW1=1,10
      RHO1=RHO(PBT,TB1)
      ETA1=ETA(PBT,TB1)
      U1DUAS=RU1DRU*RHOI/RHO1*SQRT(LAM1*0.125)
      U1AS=U1DUAS*UI
      FF=RHOI*CPI*U1AS/QA
      DD=ETA1/(RHO1*U1AS)
      AS=-TETA2*PEI*U1DUAS/GTI
      BS=(TWI0-TI)*FF-AS*(ALOG((R2-R1)/DD)-GTI)
      TSCHA(II,M)=FI/F2*TI-F1/F2*(TWI0-(AS*(ALOG((R0-R1)/DD)-GT1)+BS)/
     /FF)
```

```
      IF(TSCHA(II,M).LE.TE)TSCHA(II,M)=TE
      TSCHB(II,M)=FI/F1*TI-F2/F1*TSCHA(II,M)
      IF(ABS(TSCHB(II,M)/TB1-1.).LE.1.E-04)GOTO 22
      TB1=TSCHB(II,M)
   20 CONTINUE
C     ...................................................................
C     END OF LOOP ITW1: POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
      WRITE(6,21)I,JPIN,TB1
   21 FORMAT(1H1,5X,'STOP IN RTSI (LOOP ITW1)I=',I5,5X,'PIN=',I5,'TB1=',
     1E15.5)
      RETURN 1
C     ...................................................................
C     CONVERGENCE HAS BEEN REACHED FOR THE BULK TEMPERATURES OF THE
C     TWO ZONES DIVIDED BY THE TAU=0 LINE
C     CORRECTION OF THE PREVIOUSLY COMPUTED PIN AND SHROUD TEMPERATURES
C     OF THE CORNER CHANNELS AND OF THE WALL PORTION OF THE WALL SUBCHA=
C     NNELS IN THE CASE OF HEATED SHROUD WALLS (SUPERPOSITION PRINCIPLE)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 22.05.1980
   22 CALL TELIN(TWI,TLINER(II,M),TI,TE,TETA2,FTWA,QA
     1,QALIN,NUI,NUTU,A1,KI,R1DR2,DEI,I,JPIN,YYI,FACHE,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C  22 IF(IQLIN.EQ.1 .AND. ABS(QALIN)
C    O.GT.1.E-06)CALL TELIN(TWI,TLINER(II,M),TI,TE,TETA2,FTWA,QA
C    1,QALIN,NUI,NUTU,A1,KI,R1DR2,DEI,I,JPIN,YYI,FACHE,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC ORIGINALE
C  22 IF(IQLIN.EQ.1 .AND. ABS(QALIN)
C    O.GT.1.E-06)CALL TELIN(TWI,TLINER(II,M),TI,TE,TETA2,FTWA,QA
C    1,QALIN,NUI,NUTU,A1,KI,R1DR2,DEI,I,JPIN,YYI,FACHE)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
 2000 RETURN
      END
```

```
      SUBROUTINE SELAWA
C     -----------------------------------------------------------------
C     SELAWA COMPUTES THE KAPPA VALUES FOR THE SUBCHANNELS AND THE TWO
C     KAPPA VALUES FOR THE TWO PORTIONS OF THE WALL SUBCHANNELS IN THE
C     LAMINAR CALCULATIONS
      COMMON /LAMIN6/ANGLAM/WALLKA/AKAWC(2)/WAKAO/P,W,Z,ZWC,A,PW
     1      /LAMINK/BKAPPA(7,3)/INPAR/IPA
C
      BKAPPA(IPA,1)=GKAPPA(1.050075*P)
      BKAPPA(IPA,3)=FKAPPA(0.476156/Z)
      ALFA=ATAN(2.*(Z-ZWC)/P)
        BETA=BETAF(P,W,ZWC)
      ALFAB=ALFA*BETA
      ANGLAM=TAN(ALFAB)/TAN(ALFA)
      A2=P**2*TAN(ALFAB)*0.125-0.125*ALFAB
      PW2=ALFAB*0.5
      A1=A-A2
      PW1=PW-PW2
      R=SQRT((1.570796-ALFAB)/(4.*Z*P-P**2*TAN(ALFAB)-6.928204*ZWC**2))
      X=P*SQRT(TAN(ALFAB)/ALFAB)
      AKAWC(1)=FKAPPA(R)
      AKAWC(2)=GKAPPA(X)
  100 BKAPPA(IPA,2)=A**3/(PW**2*(A2**3/(PW2**2*AKAWC(2))+A1**3/(PW1**2*
     *AKAWC(1))))
       RETURN
      END
```

```
      SUBROUTINE SIMLA1(TE,TI,TWI,TLI,NUI,TETAI,I,JJJ,TBEQ1,TBEQ2,II)
C     ----------------------------------------------------------------
C     SIMLA1 CORRECTS THE NUSSELT NUMBERS AND THE DIMENSIONLESS TEMPERAT
C     URES OF THE UNHEATED WALLS IN THE CORNER AND WALL CHANNELS IN THE
C     LAMINAR CALCULATIONS IF THE KAPPA VALUES HAVE BEEN CORRECTED IN
C     SUBROUTINE KAPCOR
C
      REAL NUI
      COMMON/HEA6/NPIN( 42),JPIN( 42,3)/IND3/NTYP( 42)/QPAR3/PERL(3)
     1      /SUB1/ASCH( 42,3)/GEO0/ACH(3)/MART2/NS1,NS2/INPAR/IPA
     2      /LAMINK/BKAPPA(7,3)/LAMIN1/AKAPPA( 42)/WALLCO/WFCO1( 18,2),
     3      WFCO( 18,2)/SUB2/TB( 42,3),BMASS( 42,3)/SIMLAM/ISIMPL
      IF(I.GT.NS1 .OR. JJJ.GT.1)GOTO 20
      TBAVR=0.
      TBAVL=0.
      PERLT=0.
      SANG=0.
      AVRAKR=0.
      AVRAKL=0.
      DO 10 NS=NS1,NS2
      NP=NPIN(NS)
      ITYP=NTYP(NS)
      DO 10 M=1,NP
      PERLSC=PERL(ITYP)*ASCH(NS,M)/ACH(ITYP)
      ANG=60.*FLOAT(7-2*ITYP)*ASCH(NS,M)/ACH(ITYP)
      SANG=SANG+ANG
      PERLT=PERLT+PERLSC
      RAKA=BKAPPA(IPA,ITYP)/AKAPPA(NS)
      RAKR=RAKA*ANG*WFCO(II,JJJ)
      RAKL=RAKA*PERLSC*WFCO1(II,JJJ)
      AVRAKR=AVRAKR+RAKR
      AVRAKL=AVRAKL+RAKL
      TBAVR=TBAVR+TB(NS,M)*RAKR
   10 TBAVL=TBAVL+TB(NS,M)*RAKL
      TBAVR=TBAVR/AVRAKR
      TBAVL=TBAVL/AVRAKL
      AVRAKR=AVRAKR/SANG
      AVRAKL=AVRAKL/PERLT
      TBEQ1=TE+(TBAVR-TE)*AVRAKR
      TBEQ2=TE+(TBAVL-TE)*AVRAKL
C     ................................................................
C
      ENTRY SIMLA2(TI,TWI,TLI,NUI,TETAI,TBEQ1,TBEQ2)
   20 CO1=1.+(TBEQ1-TI)/(TWI-TI)
      CO2=1.+(TBEQ2-TI)/(TLI-TI)
      IF(ISIMPL.EQ.2)GOTO 1111
      CO1=1.
      CO2=1.
 1111 CONTINUE
      NUI=NUI/CO1
      TETAI=TETAI*CO2
      TWI=TI+(TWI-TI)*CO1
      TLI=TI+(TLI-TI)*CO2
      RETURN
      END
```

```
      FUNCTION SMFUN1(RHOI,ETAI,DETOT,PROV,I,KVIA ,REAI,DAI,SQ8LIA,RO,
     *G,CS)
C-----------------------------------------------------------------------
C     FUNCTION SMFUN1 EVALUATES SQRT(LAMBDA/8) FOR THE SMOOTH REGION OF
C     CORNER SUBCHANNELS (SECOND CALCULATION STEP) .
C
      COMMON/ANG1/R2(30),ALFA(30)/COLAM2/COLAMA
      BETA= R0/R2(I)
      G=(G*2.-8.1815+1.25*BETA)/(1.+BETA)
      IF(KVIA.EQ.1)GOTO 3
C     ..................................................................
C     AFTER THE FIRST ITERATION IN RECANG
C
      SMFUN1=(2.5*ALOG((R2(I)-R0)/DAI*REAI/SQ8LIA)-G)*CS+5.5*COLAMA
      RETURN
C     ..................................................................
C     AT THE FIRST ITERATION IN RECANG
C
    3 UAST2=SQRT((1.-BETA**2)/(1.-ALFA(I)))*PROV/(DETOT*SQRT(RHOI))
      SMFUN1=CS*(2.5*ALOG((R2(I)-R0)*RHOI/ETAI*UAST2)-G)+5.5*COLAMA
      RETURN
      END
```

```
      SUBROUTINE SPANU(R1,P1,NS,M,Y)
C ----------------------------------------------------------------------
C     ORGANIZES THE CALCULATION OF THE COEFFICIENT FOR THE EFFECT OF
C     THE SPACER ON THE NUSSELT NUMBER.
C
      COMMON /GEO5/   ATC,DETC,ATW,DETW,ATA,DETA,AAC,AAW,AAA
      COMMON /IROSMO/ IRH
      COMMON /SC07L/  WSP0,XM,NSPACT
      COMMON /SC08L/  AGRI( 42,3,7)
      COMMON /SC09L/  DIST(7)
      COMMON /SC10L/  RE,PR,EPS,DET1
      COMMON /SC01S/  IHAS
      COMMON /GEO2 /  ATOT,DET,ASEC
      COMMON /IND3/   NTYP( 42)
C
C     LOCAL OR AVERAGE APPLICATION OF THE CORRELATION
C
      DE=DETC
      IF(NTYP(NS).EQ.2) DE=DETW
      IF(NTYP(NS).EQ.3) DE=DETA
      IF(IHAS .LE.0   ) DE=DET
      RX=RE
      PX=PR
      IF(IHAS.LE.0) GO TO 10
      RX=R1
      PX=P1
   10 CONTINUE
C
      AL=WSP0/DE
      XA=XM/DE
      Y=1.0
      DO 1000 K=1,NSPACT
              X=XA-(DIST(K)-(WSP0/2.))/DE
              EPX=EPS
              IF(IHAS.GT. 0) EPX=AGRI(NS,M,K)
              IF( EPX.LT.0.001) GO TO 1000
              GO TO (100,200), IRH
C -----------------------------------------------------------------------
C             SMOOTH
  100         CALL SPANUG(X,YK,RX,PX,AL,EPX)
              GO TO 300
C -----------------------------------------------------------------------
C             ROUGH
  200         CALL SPANUR(X,YK,RX,PX,AL,EPX)
C -----------------------------------------------------------------------
  300         CONTINUE
              IF (YK.GT.Y) Y=YK
 1000         CONTINUE
      IF(Y.LT.1.0 .OR. Y.GT.3.0) WRITE(6,2000) K,NS,M,Y
C
      RETURN
 2000 FORMAT( /5X,' ERROR IN SPANU: SPACER ',I2,'  CHANNEL ',I3,
     >             ' SUBCH. ',I2,/5X,
     >             ' THE FACTOR FOR THE NUSSELT IS ',E12.6,/5X,
     >             ' CALCULATION PROCEEDING')
 2100 FORMAT( /5X,' SPANU: SPACER ',I2,'  CHANNEL ',I3,
     >             ' SUBCH. ',I2,/5X,
     >             ' THE BLOCKAGE FACTOR IS ',E12.6,
     >             ' , OUT OF THE LIMITS OF VALIDITY OF THE CORRELATION',
     >             ' ( 0.15 <= EPS <= 0.35 ) ',/5X,
     >             ' CALCULATION PROCEEDING ' )
      END
```

```
      SUBROUTINE SPANUG(X,Y,RE,PR,AL,E)
C ----------------------------------------------------------------------
C     M.D. HASSAN
C     IMPROVEMENT OF THE HEAT TRANSFER IN THE SPACERS REGION.
C     CASE OF SMOOTH RODS.
C
C     RE   =REYNOLDSNUMBER
C            (RELATIVE TO THE HYDRAULIC DIAMETER OF THE CENTRAL CHANNEL)
C     PR   =PRANDTLNUMBER
C     AL   =DIMENSIONLOSS AXIAL LENGTH OF THE SPACER.
C            (RELATIVE TO THE HYDRAULIC DIAMETER OF THE CENTRAL CHANNEL)
C     E    =BLOCKAGE AREA (.15<E<.35)
C
C     X    =DIMENSIONLESS AXIAL DISTANCE OF THE CONSIDERED POINT TO THE
C            BEGINN OF THE SPACER.
C            (RELATIVE TO THE HYDRAULIC DIAMETER OF THE CENTRAL CHANNEL)
C     Y    =NUSSELTNUMBER RATIO     (NUX/NUO)
C            NUX = NUSSELT IN SPACER REGION
C            NUO = NUSSELT IN THE UNDISTURBED REGION
C
C ----------------------------------------------------------------------
C     EVALUATION OF THE MAXIMAL NUSSELT RATIO
      IF(RE.LT.2.E3) AL=AL/2
      RNUMX=1.+.174*E**2*RE**.5
      IF(RE.LT.2.2E3) GO TO 11
      RNUL=RNUMX
      RNUMX=1.+E**2.4*(6.38+4.55E3/RE**.8)
      IF(RNUL.LT.RNUMX) RNUMX=RNUL
   11 CONTINUE
C ----------------------------------------------------------------------
C     EVALUATION OF THE STEIGUNG <M>
      CMAX=4.
      C=1.855E-3*RE
      IF(C.GE.CMAX) C=CMAX
      IF(RE.LT.3.E3) GO TO 12
      C=30.34*RE**(-.253)
      IF(C.GE.CMAX) C=CMAX
   12 CM=-C*E**2
C ----------------------------------------------------------------------
C     EVALUATION OF THE CONSTANT <K>
      DMIN=.895
      D=4.42-1.05*ALOG10(RE)
      IF(D.LE.DMIN) D=DMIN
      IF(RE.LT.3.E3) GO TO 13
      D=.426+.113*ALOG10(RE)
      IF(D.LE.DMIN) D=DMIN
   13 CK=D-2.25*E
      XE=AL
C ----------------------------------------------------------------------
C
      YA=1+((RNUMX-1)/(XE+1))*(X+1)
      IF(YA.LT.1.) YA=1.
      YB=RNUMX
      Y=YA
      IF(YB.LT.YA) Y=YB
      IF(X.LE.0.) GO TO 20
      YC=CK*(X/RE/PR)**CM
      IF(YC.LT.1.) YC=1.
      IF(YC.LT.YB) Y=YC
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SPANUR(X,Y,RE,PR,AL,E)
C ---------------------------------------------------------------------
C     EIN FORTRAN-IV-UNTERPROGRAMM ZUR EMPIRISCHEN BERECHNUNG
C     DES WAERMEUEBERGANGS IM ABSTANDSHALTERBEREICH
C
C     INR/HASSAN
C
C     02.12.1979
C
C     EINGABEDATEN
C
C     RE    =REYNOLDSZAHL,BERECHNET MIT DEM HYDRAULISCHEN
C            DURCHMESSER DES GESAMTKANALS
C     PR    =PRANDTLZAHL
C     AL    =DIMENSIONLOSE ABSTANDSHALTERLAENGE,BEZOGEN AUF
C            DEN HYDRAULISCHEN DURCHMESSER DES GESAMTKANALS
C     E     =QUERSCHNITTSVERSPERRUNG (.15<E<.35)
C
C     ERGEBNISSE
C     X     =DIMENSIONLOSER AXIALER ABSTAND VON DER VORDERSEITE
C            DES ABSTANDSHALTERS,BEZOGEN AUF DEN HYDRAULISCHEN
C            DURCHMESSER DES GESAMTKANALS
C     Y     =NUSSELTZAHL-VERHAELTNIS (NUX/NU0)
C *** NUX   =DURCH DEN ABSTANDSHALTER GESTOERTE NU-ZAHL(LOKAL)
C *** NU0   =UNGESTOERTE NU-ZAHL
C
C *** BESTIMMUNG DES MAX.NU-ZAHL-VERHAELTNISSES *******
      RNUMX=1.+.174*E**2*RE**.5
      IF(RE.LT.2.2E3) GO TO 31
      RNUL=RNUMX
      RNUMX=1.+E**2*(3.3+7.27E4/RE**1.2)
      IF(RNUL.LE.RNUMX) RNUMX=RNUL
   31 CONTINUE
C
C *** BESTIMMUNG DER STEIGUNG <M> *********
      CMAX=4.
      C=1.855E-3*RE
      IF(C.GE.CMAX) C=CMAX
      IF(RE.LT.3.E3) GO TO 32
      C=1.+4.9E4/RE**1.2
      IF(C.GE.CMAX) C=CMAX
   32 CM=-C*E**2
C *** BESTIMMUNG DER KONSTANTEN <K> **********
      DMIN=.885
      D=4.42-1.05*ALOG10(RE)
      IF(D.LE.DMIN) D=DMIN
      IF(RE.LT.3.E3) GOTO 33
      D=-.344+.35*ALOG10(RE)
      IF(D.LE.DMIN) D=DMIN
      IF(RE.LT.8E3) GO TO 33
      DK=D
      D=-1.8478+1.2466*ALOG10(RE)-.1298*ALOG10(RE)**2
      IF(DK.LE.D) D=DK
   33 CK=D-2.25*E
C
      XE=AL/2
C ************************
      YA=1+((RNUMX-1)/(XE+1))*(X+1)
      IF(YA.LT.1.) YA=1.
      YB=RNUMX
      Y=YA
```

```
      IF(YB.LT.YA) Y=YB
      IF(X.LE.0.) GO TO 40
      YC=CK*(X/RE/PR)**CM
      IF(YC.LT.1.) YC=1.
      IF(YC.LT.YB) Y=YC
   40 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SUBBAL(NSTOT,NSTR, INDSP,H,LENGTH,D,PIG,PR1,PR2,PBT,FRE
     *L,FT,ITCORR,DPAV,*,WSP,I1SPAC)
C-----------------------------------------------------------------------
C     SUBROUTINE SUBBAL EVALUATES THE SUBCHANNEL MASS FLOW RATES AND
C     BULK TEMPERATURES
C
      REAL LAMSCH,MI,MAV,MSCH1,MSCH,MAVCF,LENGTH,MINS(3),M2NS(3),
     1     MAV1,MAV2 ,MAWC,KAPPA
      DIMENSION RHO1(3),TINS(3),WCFUD(3),WCF1NS(3),EP1NS(3),TANS(3),
     1          T2NS(3),RHOAV(3),ANS(3),XMEM(3),DE(3) ,A( 42)
      COMMON/CORR/SIGMA( 42),PHI( 42)/CORR1/SIGMAI( 42,3),PHII( 42,3)
     1     /GRID0/CSPAC( 42,3,4)/IJ1/NER( 42),NIS( 42,3)/IND3/NTYP( 42)
     2     /GEN2/AZ( 42)/GEN3/MI( 42)/GEN5/DEZ( 42)/MOB2/UAV( 42)
     3     /MOB4/WCF( 42)/MOB5/TAV( 42)/MOB6/MAV( 42)/MOB8/DP( 42)
     4     /SUBC1/NCHC(3),JSCH(3,3)/SUBC2/JCHC(3,2)/SUB1/ASCH( 42,3)
     5     /SUB2/TSCH( 42,3),MSCH( 42,3)/SUB5/LAMSCH( 42,3)
     6     /SUB6/TSCH1( 42,3)/SUB8/MSCH1( 42,3)/HEA10/QSCH( 42,3)
     7     /SUB31/WCFNS(3),DPNS(3),WTNS1(3,3),WTNS2(3,2),UNS(3),RUASNS(
     8     3)/MOB24/WT( 42,3)/MOB26/RUAS( 42)/TUR2/CTURB1(2)
     9     /HEA6/NPIN( 42),JPIN( 42,3)/GEO0/ACH(3)
      COMMON/GRID1/EPS( 42,3,5),DISTSP(7)/GRID8/PGDP( 42,3,4)
     1     /SUB3/ADAB( 18,2),DDTBB( 18,2)
     >     /WACO1/XMSCHB( 18,2),XMSCHA( 18,2)
     2     /QPAR1/QDEV/QPAR2/QLINM,QLDEV/QPAR3/PERL(3)
     3     /LAMIN0/I2TIP( 42,3) /LAMIN3/F1ATIP( 42),F1DTIP( 42)
     4     /LAMIN4/F2ATIP( 42,3),F2DTIP( 42,3)/WCSE7/MAWC ( 18,2,2)
     5     /WCSE9/TAVWC( 18,2,2)/CORR2/CHI( 18,2,2),PSI( 18,2,2)
     6     /WCSE8/ASCHWC( 18,2,2)/COND1/CCOND( 42,3)/COND2/CCOND1(2)
     7     /GRAV/IGRAV/SUBDI/IDIV1,IDIV2/GAAG1/FCOPW1(3)
     8     /ENEOP/IENE/GRID2/YY(100, 42,3)/MIXS2/CY/SECIN/K
      COMMON /SC06L/ SHQ( 18,2)
      COMMON /SC21C/ SHQC( 18,2)
      COMMON /SC09R/ QSR( 18,2)
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC07C/ H1
C
      XX=1./980665.
C     ..................................................................
C     CORRECTION OF THE CHANNEL FLOW AREAS TO TAKE INTO ACCOUNT THAT
C     THE SUBCHANNEL GEOMETRIC PARAMETERS MUST BE BASED ON THE TIP
C     DIAMETER OF THE RODS IN THE CASE OF LAMINAR FLOW
C
      DO 1000 NS=1,NSTOT
 1000 A(NS)=AZ(NS)*F1ATIP(NS)
C     ..................................................................
C     LOOP "NS" STARTS ( NS = CHANNEL INDEX )
C
C
      DO 80 NS=1,NSTOT
      III=NS-NSTR
      FRELI=FREL
      NP=NPIN(NS)
      ITYP=NTYP(NS)
C
      NI=NER(NS)
      NP1=NP-1
      NSCH=4-ITYP
      SCH=NSCH
      AREASC=ACH(ITYP)/SCH
C     ..................................................................
C     CONNECTIONS BETWEEN THE SUBCHANNELS OF CHANNEL "NS" AND THE
```

```
C     CHANNELS ADJACENT TO "NS"
C
      CALL SUBCON(NS,NP,NP1,NI)
      IF(NPIN(NS).EQ.1)GOTO 65
      IF(ITYP.EQ.1 .AND. IDIV1.EQ.IDIV1/2*2)GOTO 65
      IF(ITYP.EQ.2 .AND. IDIV1.GT.2)GOTO 65
C
      DO 1 I=1,NP
      RHO1(I)=RHO(PR1,TSCH1(NS,I))
      MINS(I)=MSCH1(NS,I)
      ANS(I)=ASCH(NS,I)*F2ATIP(NS,I)
      DE(I)=DEZ(NS)*F2DTIP(NS,I)
      TINS(I)=TSCH1(NS,I)
    1 WCFUD(I)=WCF(NS)*ANS(I)/A(NS)
C     ..................................................................
C     ITERATION ON THE RELAXATION FACTOR (LOOP ITFREL)
C
      DO 48 ITFREL=1,98
      IVIA=1
C     ..................................................................
C     CALCULATION OF THE PRESSURE LOSSES (LOOP ITGL)
C
      DO 47 ITGL=1,60
C
C     EVALUATION OF THE CROSS-FLOW SOLUTIONS
C
      CALL CRFL1(ITGL,DPNSAV,FRELI,A(NS),NP,ANS,MINS,DPNS,WCFNS,WCF1NS,
     *  EP1NS)
      DO 2 I=1,NP
      WCFNS(I)=WCFNS(I)+WCFUD(I)
      M2NS(I)=MINS(I)-H*WCFNS(I)
      MSCH(NS,I)=(M2NS(I)+MINS(I))*0.5
      TANS(I)=TAV(NS)
    2 RUASNS(I)=MSCH(NS,I)*SQRT(LAMSCH(NS,I)*0.125)/ASCH(NS,I)*AREASC
      IF(ITGL.GT.1 .AND. IVIA.EQ.1)GOTO 25
C     ..................................................................
C     CALCULATION OF THE BULK TEMPERATURES ( LOOP ITERM )
C
      XPREC=1.E-04
      DO 20 ITERM=1,20
C
C     A) TURBOLENT EXCHANGE SUBCHANNEL-SUBCHANNEL
C
      DO 4 I=1,NP1
      MAV1=MSCH(NS,I)*AREASC/ASCH(NS,I)
      I1=I+1
      DO 3 II=I1,NP
      MAV2=MSCH(NS,II)*AREASC/ASCH(NS,II)
      IF(TANS(I).LE.0. .OR. TANS(I).GT.3000. .OR. TANS(II).LE.0. .OR.
     *TANS(II).GT.3000.)GOTO 302
       YYIII=((YY(K,NS,I)+YY(K,NS,II))*0.5-1.)*CY+1.
      WTNS1(I,II)=TME(PBT,MAV1,MAV2,TANS(I),TANS(II),LAMSCH(NS,I),
     *LAMSCH(NS,II),AREASC,AREASC,CTURB1(ITYP))*YYIII
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 10.9.1980
C     IF(I2TIP(NS,I).EQ.1 .OR. I2TIP(NS,II).EQ.1)WTNS1(I,II)=0.
C   3 WTNS1(II,I)=WTNS1(I,II)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 10.9.1980
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 10.9.1980
      IF(ABS(YY(K,NS,I)*YY(K,NS,II)-1.) .GT. 1.E-03) GO TO 3
      IF(I2TIP(NS,I).EQ.1 .OR. I2TIP(NS,II).EQ.1)WTNS1(I,II)=0.
    3 WTNS1(II,I)=WTNS1(I,II)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 10.9.1980
    4 CONTINUE
      DO 16 I=1,NP
      THEX=0.
      CONHE=0.
```

```
      MAV1=MSCH(NS,I)*AREASC/ASCH(NS,I)
      DO 5 II=1,NP
      IF(I.EQ.II)GOTO 5
      MAV2=MSCH(NS,II)*AREASC/ASCH(NS,II)
      TAIII=(MAV1*TANS(I)+MAV2*TANS(II))/(MAV1+MAV2)
      CONHE=CONHE-(TANS(I)-TANS(II))*CCOND1(ITYP)*(KAPPA(PBT,TANS(I))+
     +KAPPA(PBT,TANS(II)))
      THEX=THEX-(TANS(I)-TANS(II))*WTNS1(I,II)*CP(PBT,TAIII)
    5 CONTINUE
C
C     B) TURBOLENT EXCHANGE SUBCHANNEL-CHANNEL
C
      NCHCI=NCHC(I)
      IF(NCHCI.EQ.0)GOTO 7
      DO 6 K1=1,NCHCI
      M=JCHC(I,K1)
      I1=JSCH(I,M)
      J=NIS(NS,M)
      NTYPJ=NTYP(J)
      MAV2=MAV(J)*ACH( NTYPJ)/AZ(J)
      WTNS2(I,K1)=WT(NS,M)
      IF(I2TIP(NS,I).EQ.1)WTNS2(I,K1)=0.
      IF(I1.NE.0)WTNS2(I,K1)=WTNS2(I,K1)*(RUASNS(I)+RUAS(J))/(RUASNS(I)+
     *          RUASNS(I1)+2.*RUAS(J))
      TAIJ=(TANS(I)*MAV1+TAV(J)*MAV2)/(MAV1+MAV2)
      CONHEP=(TANS(I)-TAV(J))*CCOND(NS,M)*(KAPPA(PBT,TANS(I))+KAPPA(PBT,
     *TAV(J)))
      IF((ACH(ITYP)/AZ(NS).LE.1.1 .OR. ACH(NTYPJ)/AZ(J).LE.1.1) .AND.
     *(NTYP(NS).EQ.1 .OR. NTYP(J).EQ.1))CONHEP=CONHEP*0.5
      CONHE=CONHE-CONHEP
    6 THEX=THEX-(TANS(I)-TAV(J))*WTNS2(I,K1)*CP(PBT,TAIJ)
C
C     C) CROSS FLOW HEAT EXCHANGE SUBCHANNEL-SUBCHANNEL
C
    7 CONTINUE
      TACF=0.
      MAVCF=0.
      DO 8 II=1,NP
      IF(I.EQ.II)GOTO 8
      MAV2=MSCH(NS,II)*AREASC/ASCH(NS,II)
      CALL CF1(TANS(I),TANS(II),MAV1,MAV2,DPNS(I),DPNS(II),
     *    ITGL,TACF,MAVCF)
C
C     D) CROSS FLOW HEAT EXCHANGE  SUBCHANNEL-CHANNEL
C
    8 CONTINUE
      IF(NCHCI.EQ.0)GOTO 12
      DO 11 K1=1,NCHCI
      M=JCHC(I,K1)
      J=NIS(NS,M)
      NTYPJ=NTYP(J)
      MAV2=MAV(J)*ACH(NTYPJ)/AZ(NS)
      CALL CF1(TANS(I),TAV(J),MAV1,MAV2,DPNS(I),DP(J),ITGL,TACF,MAVCF)
   11 CONTINUE
   12 CONTINUE
C
C
      TACF=TACF/MAVCF
      TAICF=(TANS(I)*MAV1+TACF*MAVCF)/(MAV1+MAVCF)
      CFHEX=WCFNS(I)*(TANS(I)-TACF)*CP(PBT,TAICF)
      PHII(NS,I)=(THEX+CFHEX+CONHE)*H/ASCH(NS,I)
      XXMAV=MSCH(NS,I)
      XXM2=M2NS(I)
      IF(IENE.EQ.2)XXMAV=MINS(I)
      IF(IENE.EQ.2)XXM2=XXMAV
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
C
C     T2NS(I)=TSCH1(NS,I)+H/(XXMAV*CP(PBT,TANS(I)))*((QSCH(NS,I)*QDEV
C    +  + QLINM*PERL(ITYP)*0.5*QLDEV)/LENGTH+THEX+CFHEX+CONHE)
C
      IPIN=JPIN(NS,I)
      QPARE=0.0
      IF (ITYP.NE.1)  QPARE= SHQ(III,I)+SHQC(III,I)
      QSCHT=QSCH(NS,I)*QDEV*H1+QJ(IPIN,NS) + QPARE
      T2NS(I)=TSCH1(NS,I)+ ( QSCHT+(THEX+CFHEX+CONHE)*H ) /
     >                         (XXMAV*CP(PBT,TANS(I)))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      TSCH(NS,I)=(XXM2*T2NS(I)+MSCH1(NS,I)*TSCH1(NS,I))*0.5/
     *    XXMAV
   16 CONTINUE
      IF(ITGL.EQ.1)GOTO 25
C
C     TEST OF CONVERGENCE FOR THE GAS TEMPERATURES
C
      IF(ITERM.GT.10)XPREC=1.E-03
      IF(ITERM.GT.15)XPREC=1.E-02
      DO 17 I=1,NP
      IF(ABS(TANS(I)/TSCH(NS,I)-1.).GT.XPREC)GOTO  18
   17 CONTINUE
      GOTO 25
   18 CONTINUE
      DO 19 I=1,NP
   19 TANS(I)=TSCH(NS,I)
   20 CONTINUE
C     ..................................................................
C     END OF LOOP ITERM: POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
      WRITE(6,21)NS,(TANS(I),I=1,NP),ITCORR
   21 FORMAT(    5X,'SUB. SUBBAL',
     >            /5X,'SUBCHANNEL CALCULATION STOPS IN LOOP ITERM  OF CHAN
     *NEL',I6,5X,'TEMPERATURES='/5X,3E15.7/5X,'ITCORR=',I5)
      RETURN 1
C     ..................................................................
C     CONVERGENCE HAS BEEN REACHED FOR THE ENERGY EQUATIONS; THE CALCU=
C     LATION OF THE PRESSURE DROPS STARTS
C
   25 CONTINUE
      DO 26 I=1,NP
      RHOAV(I)=RHO(PBT,TSCH(NS,I))
      UNS(I)=MSCH(NS,I)/(ANS(I)    *RHOAV(I))
   26 CONTINUE
      DPNSAV=0.
      SMSCH1=0.
      DO 40 I=1,NP
      TMOEX=0.
C
C     TURBULENT EXCHANGE SUBCHANNEL-SUBCHANNEL
C
      DO 27 II=1,NP
      IF(I.EQ.II)GOTO 27
      TMOEX=TMOEX-(UNS(I)-UNS(II))*WTNS1(I,II)
   27 CONTINUE
C
C     TURBULENT EXCHANGE SUBCHANNEL-CHANNEL
C
      NCHCI=NCHC(I)
      IF(NCHCI.EQ.0)GOTO 29
      DO 28 K1=1,NCHCI
      M=JCHC(I,K1)
      J=NIS(NS,M)
```

```
   28 TMOEX=TMOEX-(UNS(I)-UAV(J))*WTNS2(I,K1)
   29 TMOEX=TMOEX*FT*H/ANS(I)
C
      UACF=0.
      ACF=0.
      AREAI =AREASC*F2ATIP(NS,I)
C
C     CROSS-FLOW EXCHANGE SUBCHANNEL-SUBCHANNEL
C
      DO 30 II=1,NP
      IF(I.EQ.II)GOTO 30
      AREAII=AREASC*F2ATIP(NS,II)
      CALL CF1(UNS(I),UNS(II),AREAI ,AREAII,DPNS(I),DPNS(II),
     *         1,UACF,ACF)
   30 CONTINUE
C
C     CROSS-FLOW EXCHANGE SUBCHANNEL-CHANNEL
C
      IF(NCHCI.EQ.0)GOTO 36
      DO 35 K1=1,NCHCI
      M=JCHC(I,K1)
      J=NIS(NS,M)
      NTYPJ=NTYP(J)
      AREAJ=ACH(NTYPJ)*F1ATIP(J)
      CALL CF1(UNS(I),UAV(J),AREAI ,AREAJ,DPNS(I),DP(J),1,
     *    UACF,ACF)
   35 CONTINUE
C
C
   36 UCF=UACF/ACF
      CFMOEX=(2.*UNS(I)-UCF)*WCFNS(I)/ANS(I)*H
      XMEM(I)=LAMSCH(NS,I)*H/(2.*DE(I) *RHOAV(I))*FCOPW1(ITYP)
      RE=MSCH(NS,I)*DE(I)/(ANS(I)     *ETA(PBT,TSCH(NS,I)))
      IF(INDSP.EQ.2)XMEM(I)=XMEM(I)+(CSPAC(NS,I,I1SPAC)+DSPDPF(EPS(NS,I,
     *I1SPAC),DE(I) ,LAMSCH(NS,I),WSP,PGDP(NS,I,I1SPAC),RE,ITYP))/
     /RHOAV(I)
      DPNS(I)=XX*(-(MSCH(NS,I)/ANS(I)     )**2*(XMEM(I)-(RHO(PR2,T2NS(I))
     *     -RHO1(I))/RHOAV( I)**2)+TMOEX+CFMOEX+IGRAV*RHOAV(I)*980.665*
     *     H)
      DPNSAV=DPNSAV+DPNS(I)*MSCH1(NS,I)
      SMSCH1=SMSCH1+MSCH1(NS,I)
   40 CONTINUE
      DPNSAV=DPNSAV/SMSCH1
      IF(ITGL.LT.4)GOTO 45
C     ..............................................................
C     TEST FOR THE CONVERGENCE OF THE PRESSURE DROPS
C
      DO 41 I=1,NP
      IF(ABS(DPNS(I)/DPNSAV-1.).GT.1.E-02)GOTO 45
      IF(ABS(DPNS(I)/DPNSAV-1.).GT.1.E-03 .AND. ITGL.LT.40)GOTO 45
   41 CONTINUE
      IF(IVIA.EQ.2)GOTO 50
      DO 301 I=1,NP
      IF(M2NS(I).LE.0.)GOTO 302
  301 CONTINUE
      IVIA=2
   45 CONTINUE
      DO 46 I=1,NP
   46 WCFNS(I)=WCFNS(I)-WCFUD(I)
   47 CONTINUE
C     ..............................................................
C     END OF LOOP ITGL : POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
  302 CONTINUE
      AIT=ITFREL
      FRELI=1.-AIT*0.01
```

```
   48 CONTINUE
C     ...................................................................
C     END OF LOOP ITFREL: POINT REACHED IN THE CASE OF CONVERGENCE
C     PROBLEMS
C
      WRITE(6,49)ITCORR,NS,(DPNS(I),I=1,NP),(MSCH(NS,I),I=1,NP),
     *     (TSCH(NS,I),I=1,NP)
   49 FORMAT(//  5X,'SUB. SUBBAL',
     >          /5X,'SUBCHANNEL CALCULATION STOPS  IN LOOP ITGL: ITCORR=
     1',I5,5X,'NS=',I5/5X,'PRESSURE LOSSES + AVERAGE MASSES + AVERAGE TE
     2MPERATURES : '/(8E15.5))
  777 RETURN 1
C     ...................................................................
C     CONVERGENCE HAS BEEN REACHED FOR THE ENERGY EQUATIONS AND FOR THE
C     AXIAL MOMENTUM EQUATIONS
C
   50 CONTINUE
      DO 60 I=1,NP
      DPAVF=DPAV-IGRAV*RHOAV(I)*H*0.001
      BMI=SQRT(ABS(DPAVF) /(XX*XMEM(I)))*ANS(I)
      SIGMAI(NS,I)=(MSCH(NS,I)-BMI)/ASCH(NS,I)
   60 CONTINUE
      GOTO 70
C     *******************************************************************
C     FOR THE CHANNELS WITH ONLY ONE SUBCHANNEL
C
   65 CONTINUE
      DO 66 I=1,NP
      IF(NTYP(NS).NE.2)GOTO 7007
      WCFNS(I)=WCF(NS)
      M1=JCHC(I,1)
      M2=JCHC(I,2)
      WTNS2(I,1)=WT(NS,M1)
      WTNS2(I,2)=WT(NS,M2)
      RUASNS(I)=RUAS(NS)
      UNS(I)=UAV(NS)
 7007 CONTINUE
      MSCH(NS,I)=MAV(NS)*ASCH(NS,I)/AZ(NS)
      TSCH(NS,I)=TAV(NS)
      SIGMAI(NS,I)=SIGMA(NS)
      PHII(NS,I)=PHI(NS)
      IF(NTYP(NS).NE.3)GOTO 66
      EPSM=MSCH(NS,I)-(XMSCHA(III,I)+XMSCHB(III,I))
      XMSCHA(III,I)=XMSCHA(III,I)+EPSM*(1.-1./ADAB(III,I))
      XMSCHB(III,I)=MSCH(NS,I)-XMSCHA(III,I)
   66 CONTINUE
C
   70 CONTINUE
      IF(NTYP(NS).NE.2)     GOTO 80
C     *******************************************************************
C     ONLY FOR THE WALL SUBCHANNELS
C
      I2TTIP=0
      DO 4001 I=1,NP
      I2TTIP=I2TTIP+I2TIP(NS,I)
      DO 4000 JWC=1,2
      CHI(III,I,JWC)=1.
      PSI(III,I,JWC)=1.
      TAVWC(III,I,JWC)=TSCH(NS,I)
 4000 MAWC(III,I,JWC)=MSCH(NS,I)*ASCHWC(III,I,JWC)/ANS(I)
      IF(IDIV2.EQ.1)GOTO 4001
      EPSM=MAWC(III,I,1)-(XMSCHA(III,I)+XMSCHB(III,I))
      XMSCHA(III,I)=XMSCHA(III,I)+EPSM*(1.-1./ADAB(III,I))
      XMSCHB(III,I)=MAWC(III,I,1)-XMSCHA(III,I)
 4001 CONTINUE
```

```
C     ..................................................................
C     RECCA2 IS CALLED ONLY IF THE FLOW IS TURBULENT IN THE WHOLE WALL
C     CHANNEL IN CASE OF IDIV2=1
C
      IF(IDIV2.EQ.2 .OR. I2TTIP.NE.0)GOTO 80
      IF(ITYP.EQ.2 .AND. IDIV1.GT.2)GOTO 80
      CALL        RECCA2  (NS,III,NP,INDSP,H,LENGTH,PR1,PR2,PBT,FRELI,FT,
     *ITCORR,PIG,D,DPAV,&777,WSP,I1SPAC)
C
C
   80 CONTINUE
C     ..................................................................
C     END OF LOOP "NS" : THE CALCULATIONS HAVE BEEN PERFORMED FOR ALL
C     SUBCHANNELS OF ALL CHANNELS
C
      RETURN
      END
```

```
      SUBROUTINE SUBCON(NS,NP,NP1,NI)
C-----------------------------------------------------------------------
C     SUBROUTINE SUBCON EVALUATES THE NUMBER OF CHANNELS CONNECTED TO
C     EACH SUBCHANNEL I OF CHANNEL NS ( NCHC(I) ), IDENTIFIES THESE
C     CHANNELS BY MEANS OF JCHC(I,K), IDENTIFIES WHICH SUBCHANNEL II OF
C     THE SAME CHANNEL NS IS CONNECTED TO THE SAME CHANNEL ( BY MEANS OF
C     JSCH(I,M) ).
C
      COMMON/HEA6/NPIN( 42),JPIN( 42,3)/IJ1/NER( 42),NIS( 42,3)
     1      /SUBC1/NCHC(3),JSCH(3,3)/SUBC2/JCHC(3,2)/IND3/NTYP( 42)
      DO 4 I=1,NP
      NCHC(I)=0
      DO 3 M=1,NI
      J=NIS(NS,M)
      NPJ=NPIN(J)
      DO 1 IJ=1,NPJ
      IF(JPIN(J,IJ).EQ.JPIN(NS,I))GOTO 2
    1 CONTINUE
      GOTO 3
    2 NCHC(I)=NCHC(I)+1
      NCHCI=NCHC(I)
      JCHC(I,NCHCI)=M
      JSCH(I,M)=0
    3 CONTINUE
    4 CONTINUE
      IF(NP .EQ.1)RETURN
C
      DO 9 I=1,NP1
      IF(NCHC(I).EQ.0)GOTO 9
      NCHCI=NCHC(I)
      DO 8 K1=1,NCHCI
      I1=I+1
      DO 6 II=I1,NP
      IF(NCHC(II).EQ.0)GOTO 6
      NCHCII=NCHC(II)
      DO 5 K2=1,NCHCII
      IF(JCHC(I,K1).EQ.JCHC(II,K2))GOTO 7
    5 CONTINUE
    6 CONTINUE
      GOTO 8
    7 JCHCIK=JCHC(I,K1)
      JSCH(I,JCHCIK)=II
      JSCH(II,JCHCIK)=I
    8 CONTINUE
    9 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SUBDH(N,K,K1,NSTOT)
C-----------------------------------------------------------------------
C     SUBDH HALVES THE K.TH AXIAL SECTION IF CONVERGENCE PROBLEMS
C     OCCURRED IN IT
C
      COMMON/GRID2/YY(100, 42,3)/GRID3/X(100)/HEA6/NPIN( 42),JPIN( 42,3)
C     ..................................................................
C     THE MAXIMUM VALUE OF THE AXIAL INDICES IS 100
C
      IF(N.LT.100)GOTO 2
      WRITE(6,3)
    3 FORMAT(1H1,5X,'SUB. SUBDH',
     >          /5X,'NUMBER OF AXIAL SECTIONS BECOMES TOO BIG')
      STOP
C     ..................................................................
C
    2 CONTINUE
      NI=N-K
      N=N+1
      DO 10 I=1,NI
      II=N-I+1
      X(II+1)=X(II)
      X(II)=X(II-1)
      DO 1 NS=1,NSTOT
      NP=NPIN(NS)
      DO 1 M=1,NP
    1 YY(II,NS,M)=YY(II-1,NS,M)
   10 CONTINUE
      X(K+1)=(X(K)+X(K+2))*0.5
      DO 20 NS=1,NSTOT
      NP=NPIN(NS)
      DO 20 M=1,NP
   20 YY(K+1,NS,M)=YY(K,NS,M)
      K1=K
      WRITE(6,30)
   30 FORMAT(/130('*')//)
      RETURN
      END
```

```
      SUBROUTINE SUFCON(A,IND)
C ---------------------------------------------------------------------
C     TESTS THE SUFFICIENT CONVERGENCE CONDITION FOR THE USE OF THE
C     GAUSS-SEIDEL METHOD.
C
      DIMENSION A(132, 13)
      COMMON /SC01R/ NSECT,NSECP
      COMMON /SC13R/ NAFF(132)
C
      IND  = 0
      IDIM= 13
      IND1 =IDIM
      IND2 = 0
C
      DO 1000 L=1,NSECT
              SUM=0.0
              NAF=NAFF(L)
              NAF1=NAF+1
              DO 500 K=1,NAF
                     SUM=SUM+ABS(A(L,K))
  500                CONTINUE
              IF(ABS(A(L,NAF1)) .LT. SUM) IND1=IND1-1
              IF(ABS(A(L,NAF1)) .GT. SUM) IND2=IND2+1
 1000         CONTINUE
C
      IF(IND1.LT.IDIM.OR.IND2.LE.0) IND=1
C
      RETURN
      END
```

```
      SUBROUTINE SUPCEN(RH,PBT,TWALL,LAM1,U1DU,YYI,ITYP,KI,DEI,FACHE,
     >                  QA,TW1,G1A,NU11,I,M,REI,ITW1)
C ----------------------------------------------------------------------
C     THE TEMPERATURES OF THE SECTORS FACING A CENTRAL CHANNEL (OR THE
C     CENTRAL PART OF A WALL SUBCHANNEL) ARE MODIFIED BY MEANS OF THE
C     SUPERPOSITION PRINCIPLE.
C
      COMMON /SUB23/ HPLUSB( 42,3),HPLUSW( 42,3),QPLUS( 42,3),
     >               PRB( 42,3),YODH( 42,3)
      COMMON /SUB22/ TW( 42,3)
      COMMON /SUB2/  TSCH( 42,3),MSCH( 42,3)
      COMMON /HEA5/  QQ( 42,3)
      COMMON /IJ1/   NER( 42),NIS( 42,3)
      COMMON /DAT/   PIG
      COMMON /MART5/ NSTR
      COMMON /QPAR1/ QDEV
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC02R/ P,D,Z,ZWC,H,LENGTH
      COMMON /SC13C/ GEO1( 42,3)
      COMMON /HEA6 / NPIN( 42),JPIN( 42,3)
      COMMON /WCSE9/ TAVWC( 18,2,2)
      COMMON /WCSE7/ MAWC( 18,2,2)
      COMMON /WCSE12/TWWC( 18,2,2)
C
      REAL LENGTH,KAPPA,LAM1,KI,NU11,MAWC,MSCH
C
      R1=D*0.5
      S3=1.732051
      IF(I.GT.NSTR) GOTO 1000
C     -----------------------------------------------------------------
C     CENTRAL SUBCHANNELS
C  1- DETERMINATION OF THE CHANNEL BULK TEMPERATURE
      NR=NER(I)
      SPM=0.0
      T=0.0
      DO 100 M1=1,NR
             SPM=SPM+MSCH(I,M1)
             T=T+TSCH(I,M1)*MSCH(I,M1)
  100        CONTINUE
      TB=T/SPM
C
C  2- DETERMINATION OF NU11
      R2A=SQRT(P**2*S3*3./(2.*PIG)-R1**2*2.)
      BETA=PIG/6.-ATAN((P*0.5-R1)/(P*S3*0.5))
      R2B=SQRT((P*0.5*S3*R1-PIG*R1**2/3.)/BETA)
      YDA=(R2A-R1)/RH
      YDB=(R2B-R1)/RH
      R1DR2A=R1/R2A
      R1DR2B=R1/R2B
      CALL RNU(HPLUSW(I,M),TWALL,LAM1,REI,PRB(I,M),TB,YDA,R1DR2A,0.0,
     >         U1DU,0.0,YYI,NU11,G1A,ITYP,R2A,R1,R2B,BK)
      AL=NU11*KI/DEI*FACHE
      DT=QA/AL
C
C  3- DETERMINATION OF TETA(I,J)
      DO 200 M1=1,NR
             IF(M1.EQ.M) GOTO 200
             G1B=GHPLUS(HPLUSW(I,M1),TW(I,M1),TB,PRB(I,M1),YDB,0.0,0.0,
     >                  R2B,R1)
             BB=2.5*ALOG((R2B-R1)/RH)
             BA=2.5*ALOG((R2A-R1)/RH)
```

```
              BC=(1.25+3.75*R1/R2A)/(1.0+R1/R2A)
              B=1.0-((BB+G1B)/(BA+G1A-BC))
              TETA=B/NU11
              JP=JPIN(I,M1)
              QF=QQ(I,M1)*QDEV/(LENGTH*PIG*D)
              QF=QF+QJ(JP,I)/(PIG*D*H)*GEO1(I,M1)
              DT=DT+ TETA*QF*DEI/KAPPA(PBT,TB)
  200         CONTINUE
      TW1=DT+TB
      GOTO 9999
C
C     ---------------------------------------------------------------
C     CENTRAL PART OF WALL SUBCHANNELS
 1000 CONTINUE
C  1- DETERMINATION OF THE "CHANNEL" BULK TEMPERATURE
      M1=2
      IF(M.EQ.2) M1=1
      NW=I-NSTR
      TB = (TAVWC(NW,M,2)*MAWC(NW,M,2) + TAVWC(NW,M1,2)*MAWC(NW,M1,2)) /
     >     (MAWC(NW,M,2) + MAWC(NW,M1,2))
C  1- DETERMINATION OF NU11
      ALFA=0.445
      R2A=SQRT(P**2*0.5*TAN(ALFA)/ALFA-R1**2)
      R2B=R2A
      YDA=(R2A-R1)/RH
      R1DR2A=R1/R2A
      CALL RNU(HPLUSW(I,M),TWALL,LAM1,REI,PRB(I,M),TB,YDA,R1DR2A,0.0,
     >          U1DU,0.0,YYI,NU11,G1A,ITYP,R2A,R1,R2B,BK)
      AL=NU11*KI/DEI*FACHE
      DT=QA/AL
C
C  3- DETERMINATION OF TETA(I,J)
CCCCCCCCCCCCCCCCCCCCCCC
      TWX=TWWC(NW,M,2)
      IF(TWX.LT.0.5) TWX=TW(I,M1)
CCCCCCCCCCCCCCCCCCCCCCC
      G1B=GHPLUS(HPLUSW(I,M1),TWX,TB,PRB(I,M1),YDA,0.0,0.0,R2A,R1)
      BB=2.5*ALOG((R2B-R1)/RH)
      BA=2.5*ALOG((R2A-R1)/RH)
      BC=(1.25+3.75*R1/R2A)/(1.0+R1/R2A)
      B=1.0-((BB+G1B)/(BA+G1A-BC))
      TETA=B/NU11
      JP=JPIN(I,M1)
      QF=QQ(I,M1)*QDEV/(LENGTH*PIG*D)
      QF=QF+QJ(JP,I)/(PIG*D*H)*GEO1(I,M1)
      DT=DT+ TETA*QF*DEI/KAPPA(PBT,TB)
      TW1=DT+TB
C
C     ---------------------------------------------------------------
 9999 RETURN
      END
```

```
      SUBROUTINE SYSOL (A,B,X)
C ---------------------------------------------------------------------
C     SOLVES THE SYSTEM OF LINEAR EQUATIONS.
C
      DIMENSION A(132, 13),B(132),X(132)
      COMMON /SC01R/ NSECT,NSECP
C
C     ----------------------------------------------------------------
C     CONTROLS ON THE CONVERGENCE POSSIBILITY.
C
      CALL SUFCON(A,INDEX)
      IF (INDEX .EQ. 0) GO TO 1000
C
C     ----------------------------------------------------------------
C
C
      WRITE (6,500)
  500 FORMAT ( ' SUB. SYSOL- THE SUFFICIENT CONDITION FOR ',
     >         'THE GAUSS-SEIDEL METHOD IS NOT RESPECTED.',
     >         ' THE METHOD IS USED ANYWAY.')
C
 1000 CONTINUE
C
C     ----------------------------------------------------------------
C     GAUSS-SEIDEL METHOD
C
      PERC=0.001
      ITMAX=NSECT*3
      CALL GAUDEL(A,B,X,PERC,ITMAX)
C
C     ----------------------------------------------------------------
C
      RETURN
      END
```

```
      SUBROUTINE TARRAY
C ----------------------------------------------------------------------
C     DETERMINES THE ARRAYS ISU,IGI,ISS
C
      COMMON /SC01C/  NCAN( 19),LIPS( 19,10)
      COMMON /SC03C/  NRODS
      COMMON /SC05C/  JZUR( 19, 42)
      COMMON /MART5/  NSTR
      COMMON /GASD1/  NSTOT
      COMMON /HEA6/   NPIN( 42),JPIN( 42,3)
      COMMON /SC01R/  NSECT,NSECP
      COMMON /SC06R/  ISU(132,2)
      COMMON /SC11R/  ISS( 18,2)
      COMMON /SC12R/  IGI( 42,3)
C
      L=0
      DO 1000 JP=1,NRODS
              NCA=NCAN(JP)
              IF (NCA .LE. 0) GO TO 1000
              DO 800 K=1,NCA
                     NS=LIPS(JP,K)
                     M=JZUR(JP,NS)
                     L=L+1
                     ISU(L,1)=NS
                     ISU(L,2)=M
                     IGI(NS,M)=L
  800                CONTINUE
 1000        CONTINUE
C
      NSTR1=NSTR+1
      DO 1800 NS=NSTR1,NSTOT
              NP=NPIN(NS)
              DO 1600 M=1,NP
                      L=L+1
                      NSW=NS-NSTR
                      ISU(L,1)=NS
                      ISU(L,2)=M
                      ISS(NSW,M)=L
 1600                 CONTINUE
 1800        CONTINUE
C
      IF ( L .EQ. NSECT) GO TO 3000
C
           WRITE(6,2000) L,NSECT
 2000      FORMAT( 5X,'SUB. TARRAY',/5X,
     >                'L     = ',I4,/5X,
     >                'NSECT = ',I4,/5X,
     >                'CALCULATION STOPS.')
           STOP
C
 3000 RETURN
      END
```

```
      SUBROUTINE TAU(I,AI,P,ALFA,D,W,RH,DET,PROV,IRH,DAI,DBI,PAI,F,
     *RHPL,TWI,TE,ITTEMP,QPLUS,ETAA,RHOA,ETAB,RHOB,ETAIW,RHOIW,BETA,EM1,
     *XC1,XC2,T1,*,CS)
C-----------------------------------------------------------------------
C     SUBROUTINE TAU EVALUATES POINTS OF THE LINE F=F(P) (F=0 FOR POINTS
C     ON THE TAU=0 LINE).
C
       COMMON /COLAM2/COLAMA/MART/ITCORR/REC1/PVERT(90)
     1       ,PRAD(90)/REC2/E(90)
C
      PVERT(I)=P*D*0.5
      PRAD(I)=PVERT(I)/COS(BETA)
      E(I)=PVERT(I) *TAN(BETA)
      IF(I.GT.1)GOTO 1
C     ..................................................................
C     FIRST SUB-SUBCHANNEL
C
      ZAI=W-0.5*D-PVERT(1)
      DELTAE=E(1)
      DBI=2.*E(1)*P/ALFA-D
      GOTO 2
C     ..................................................................
C     FOR THE I.TH SUB-SUBCHANNEL, IF I>1
C
    1 CONTINUE
      WW=W-((E(I)+E(I-1))*0.5-EM1)*XC1
      ZAI=WW-0.5*(D+PVERT(I)+PVERT(I-1))
      DELTAE=E(I)-E(I-1)
      DBI=2.*(PVERT(I-1)*TAN(BETA)-E(I-1))*P/ALFA-D
C     ..................................................................
C     FOR ALL SUB-SUBCHANNELS
C
    2 DAI=4.*ZAI/XC2
      PAI=DELTAE*XC2
      ZBI=0.5*(SQRT(D**2+D*DBI)-D)
C
      IF(DAI.GT.0. .AND. DBI.GT.0.)GOTO 100
      WRITE(6,22)I,DAI,DBI,P,E(I),E(I-1),PVERT(I),PVERT(I-1),ITCORR
   22 FORMAT(5X,'STOP IN TAU : I=',I5,5X,'DAI=',E15.5,5X,'DBI=',E15.5
     1/5X,'P=',E15.5,5X,'E(I)=',E15.5,5X,'E(I-1)=',E15.5/5X,'PVERT(I)=',
     2E15.5,5X,'PVERT(I-1)=',E15.5,5X,'ITCORR=',I5)
      RETURN 1
C
  100 CONTINUE
      F0=2.5*ALOG(ZAI*PROV*SQRT(RHOA *DAI/DET**3)/ETAA )*CS+5.5*COLAMA
      IF(IRH.EQ.2)GOTO 3
C
C     IN THE CASE OF SMOOTH RODS
C
      F01=SQRT(DBI*RHOA/(DAI*RHOB))*(2.5*ALOG(ZBI*PROV*SQRT(RHOB*DBI/DET
     ***3)/ETAB)+5.5)
      GOTO 4
C
C     IN THE CASE OF ROUGHENED RODS
C
    3 HPLUSB=RH/DET*PROV/ETAB*SQRT(DBI/DET*RHOB)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 21.09.1979
C     HPLUSW=RH*DET  *PROV/ETAIW*SQRT(DBI/DET*RHOIW)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      HPLUSW=RH/DET  *PROV/ETAIW*SQRT(DBI/DET*RHOIW)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
      YDH=(SQRT(D**2+D*DBI)-D)*0.5/RH
      RHPL=RHPLUS(HPLUSB,TWI,TE,QPLUS,HPLUSW,T1,YDH)
      F01=SQRT(DBI*RHOA/(DAI*RHOB))*(2.5*ALOG(ZBI/RH)+RHPL)
C
C
    4 F=F0-F01
      RETURN
      END
```

```
      FUNCTION TBFUN(NSTR,NSTOT)
C-----------------------------------------------------------------------
C     TBFUN EVALUATES THE MEAN LINER TEMPERATURE IN THE AXIAL SECTION
C
      COMMON/SHROUD/TLINER( 18,2)/QPAR3/PERL(3)/IND3/NTYP( 42)
     1       /HEA6/NPIN( 42),JPIN( 42,3)/SUB1/ASCH( 42,3)/GEO0/ACH(3)
C
      NSTR1=NSTR+1
      TBPIPA=0.
      PERLT=0.
      DO 10 NS=NSTR1,NSTOT
      NTYPNS=NTYP(NS)
      NP=NPIN(NS)
      DO 10 M=1,NP
      PERLSC=PERL(NTYPNS)*ASCH(NS,M)/ACH(NTYPNS)
      PERLT=PERLT+PERLSC
   10 TBPIPA=TBPIPA+TLINER(NS-NSTR,M)*PERLSC
      TBFUN=TBPIPA/PERLT
      RETURN
      END
```

```
      SUBROUTINE TBRTBS
C ----------------------------------------------------------------------
C     DETERMINES THE ARRAYS
C     TBR =REFERENCE GAS TEMPERATURE IN THE ZONE ADJACENT TO THE ROD
C     TBS =REFERENCE GAS TEMPERATURE IN THE ZONE ADJACENT TO THE SHROUD
C     TBR AND TBS ARE USED FOR THE CONDUCTION CALCULATIONS.
C
      COMMON /SC26C/  TBR( 42,3),TBS( 18,2)
      COMMON /SUB2 /  TSCH( 42,3),MSCH( 42,3)
      COMMON /SC15C/  ALFA( 42,3)
      COMMON /SC17C/  SALFA( 18,2)
      COMMON /SUB22/  TW( 42,3)
      COMMON /SHROUD/ TLINER( 18,2)
      COMMON /HEA5/   QQ( 42,3)
      COMMON /QPAR1/  QDEV
      COMMON /SC02R/  P,DU,Z,ZWC,H,LENGTH
      COMMON /SC13C/  GEO1( 42,3)
      COMMON /SC06L/  SHQ( 18,2)
      COMMON /SC21C/  SHQC( 18,2)
      COMMON /DAT/    PIG
      COMMON /SC02C/  QJ( 19, 42)
      COMMON /SC18C/  RE,RI,ALFW,ALFC
      COMMON /MART5/  NSTR
      COMMON /GASD1/  NSTOT
      COMMON /HEA6 /  NPIN( 42),JPIN( 42,3)
      COMMON /IND3/   NTYP( 42)
      COMMON /LAMIN5/ RTIP(7)
      COMMON /INPAR/  IPA
      COMMON /CEV04/  LAMOP3
      COMMON /SC34C/  ISUC
C
      REAL LENGTH
C
      D=DU
      IF(LAMOP3.EQ.1) D=RTIP(IPA)*2.
C
      DO 1000 NS=1,NSTOT
         NP=NPIN(NS)
         DO 500 M=1,NP
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 18.03.1981
C           IF(NS.GT.NSTR) GO TO 100
            IF(ISUC.EQ.1 .OR. NS.GT.NSTR) GO TO 100
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 18.03.1981
            TBR(NS,M)=TSCH(NS,M)
            GO TO 500
  100       J=JPIN(NS,M)
            QF=QQ(NS,M)*QDEV/(LENGTH*PIG*D)
            QF=QF+QJ(J,NS)/(PIG*D*H)*GEO1(NS,M)
            TBR(NS,M)=TW(NS,M)-QF/ALFA(NS,M)
            IF(NS.LE.NSTR) GO TO 500
            NW=NS-NSTR
            ALF=ALFW
            IF(NTYP(NS).EQ.3) ALF=ALFC
            QF=(SHQ(NW,M)+SHQC(NW,M))/(RI*ALF*H)
            TBS(NW,M)=TLINER(NW,M)-QF/SALFA(NW,M)
  500       CONTINUE
 1000    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE TELIN(TW1,TLINER,TI,TE,TETA2,FTWA,QA,QALIN,NU1,NUTU,
     1 A1,KI,R1DR2,DEI,I,JPIN,YYI,FACHE,M)
C-----------------------------------------------------------------------
C     TELIN COMPUTES THE LINER TEMPERATURES AND CORRECTS THE PIN TEMPERA
C     OF THE EXTERNAL CHANNELS IN THE CASE OF HEATED LINER (TURB. FLOW)
C
      COMMON /SC17C/ SALFA( 18,2)
      COMMON /SC16C/ SNU( 18,2)
      COMMON /MART5/ NSTR
      COMMON /ISUP / IQLIN
      COMMON /ISMO/  COTW
      COMMON /ISMO1/ ITECO
C
      REAL NUTU,NU1,NU2,KI
C     ..................................................................
C     INLET EFFECT ON THE LINER NUSSELT NUMBER
C
      R1=DEI*R1DR2*0.5/(1.-R1DR2)
      R2=R1+0.5*DEI
      FACHE=TIS(R1,R2,3)
C
      TG=TE
      IF(ITECO.EQ.2)TG=TI
      FNU=(1.-A1*R1DR2**0.6)*NUTU*(TG+273.16)**COTW*YYI*FACHE
C     ..................................................................
C     ITERATION FOR THE CALCULATION OF THE LINER TEMPERATURE AT
C     (Q")ROD = 0 ( LOOP ITW1 )
C
      DO 1 ITW=1,10
      TW2=TLINER
      NU2=FNU/(TW2+273.16)**COTW
      ALFA2=NU2*KI/DEI
      TLINER=TI+QALIN/ALFA2
      IF(ABS(TLINER/TW2-1.).LE.1.E-04)GOTO 5
    1 CONTINUE
C     ..................................................................
C     CONVERGENCE PROBLEMS IN THE LOOP ITW1
C
      WRITE(6,2)I,JPIN,TW2
    2 FORMAT(1H1,5X,'STOP IN TELIN: I=',I5,5X,'PIN=',I5,5X,'TLINER=',
     1E15.5)
      STOP
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 22.05.1980
    5 CONTINUE
      II=I-NSTR
      SNU(II,M)=NU2
      SALFA(II,M)=ALFA2
      IF(IQLIN.NE.1 .OR. ABS(QALIN) .LE.1.E-06) GO TO 9999
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 22.05.1980
C     ..................................................................
C     CONVERGENCE IN LOOP ITW1; CALCULATION OF THE ROD TEMPERATURE AT
C     (Q")ROD = 0
      TW1 =FTWA/R1DR2*QALIN*DEI/KI+TI
      IF(TW1 .LE.TE)TW1 =TE
      TETA1=(TW1 -TI)*KI/(QALIN*DEI)
        IF(QA.LE.1.E-06)GOTO 10
C     ..................................................................
C     REAL ROD TEMPERATURE IN THE CASE OF HEATED ROD AND HEATED SHROUD
C
      NU1=NU1/(1.+QALIN/QA*TETA1*NU1)
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCC
      IF (NU1 .GT. 9999.) NU1= 9999.0
      IF (NU1 .LT.-9999.) NU1=-9999.0
      ALFA1=NU1*KI/DEI
CCCCCCCCCCCCCCCCCCCCCCCCCC
      TW1=TI+QA/ALFA1
C     ................................................................
C     REAL SHROUD TEMPERATURE IN THE CASE OF HEATED ROD AND HEATED SHROU
C
   10 NU2=NU2/(1.+QA/QALIN*TETA2*NU2)
CCCCCCCCCCCCCCCCCCCCCCCCCC
      IF (NU2 .GT. 9999.) NU2= 9999.0
      IF (NU2 .LT.-9999.) NU2=-9999.0
      ALFA2=NU2*KI/DEI
CCCCCCCCCCCCCCCCCCCCCCCCCC
      TLINER=TI+QALIN/ALFA2
 9999 RETURN
      END
```

```
      SUBROUTINE TEMCON(IRH,K,RH,SUR,D,PIG,TE1,PBT,*,H,V,R,INDICE,QTOT)
C ----------------------------------------------------------------------
C     ORGANIZES THE CALCULATIONS OF THE WALL TEMPERATURES.
C
      COMMON /GASD1/ NSTOT
      COMMON /MART5/ NSTR
      COMMON /SC05R/ IRAD
      COMMON /SC23C/ ICS
      COMMON /INPAR/ IPA
      COMMON /MART / ITCORR
C
      EXTERNAL RTRI,RTSI
C
      IF (QTOT.LT.1.E-06) GO TO 8000
      CALL TTOT(2)
C
C -------------------------------------------------------------------
C     CALCULATION OF THE NUSSELT NUMBERS AND OF THE WALL TEMPERATURES
C     AS FUCTION OF THE CONVECTIVE HEAT TANSFER ONLY.
C
      CALL TESTNU(0)
      IF(IRH.EQ.1 ) CALL WALLTE
     *        (K,NSTOT,NSTR,RH,SUR,D,PIG, TE1 ,PBT,&8500,RTSI,H)
      IF(IRH.EQ.2 ) CALL WALLTE
     *        (K,NSTOT,NSTR,RH,SUR,D,PIG, TE1 ,PBT,&8500,RTRI,H)
      IF ( ITCORR.GT.5 .AND. INDW.GT.0) CALL TESTNU(1)
      IF ( ITCORR.GT.5 .AND. INDW.GT.0) WRITE(6,1001)
C -------------------------------------------------------------------
C     DEFINITION OF THE DUMMIES BULK TEMPERATURES.
C
      CALL TBRTBS
C     ---------------------------------------------------------------
C     RADIATIONS HEAT TRANSFER.
C
      IF(IRAD.GT.0) CALL RADIA
C     ---------------------------------------------------------------
C     COMPUTES THE WALL TEMPERATURES
C
      CALL TORG(V,R,PIG,H)
      IF (ICS.GT.0) CALL CONSHR
C -------------------------------------------------------------------
C     CONVERGENCE TESTS FOR WALL AND BULK TEMPERATURES (LOOP ITCORR)
C
      INDICE=0
      INDW  =0
      INDB  =0
      CALL TESTB(INDB)
      CALL TESTW(INDW)
      INDICE=INDB+INDW
      IF(INDICE.GT. 0. AND. ITCORR .GT. 10) WRITE(6,1002) INDB,INDW
      IF(INDICE.GT. 0 .AND. ITCORR .GT. 15) CALL JELLA(-1)
C -------------------------------------------------------------------
C
 8000 RETURN
 8500 RETURN 1
 1001 FORMAT(5X,'**************** NUSSELT NUMBERS HAVE BEEN AVERAGED')
 1002 FORMAT( ' INDB= ',I3,' INDW = ',I3)
      END
```

```
      SUBROUTINE TEMLAM(*,PBT,TI,MASSI,DEIR,AREAI,QQ,QALIN,TE,I,II,M,
     & TW1,ITYP,F2ATIP,F2DTIP,DVOL)
C     -----------------------------------------------------------------
C     TEMLAM COMPUTES THE PIN TEMPERATURES AND THE TEMPERATURE  OF THE
C     LINER IN THE SUBCHANNELS WHERE THE FLOW IS LAMINAR ( THE VELOCITY
C     PROFILE IS ASSUMED TO BE ALREADY DEVELOPED AT THE POSITION WHERE T
C     HEATING STARTS)
C         ITYP=1 : CENTRAL SUBCHANNELS AND CENTRAL PART OF WALL SUBCHANN
C         ITYP=2 : WALL PART OF WALL SUBCHANNELS
C         ITYP=3 : CORNER CHANNELS
C
      REAL MASSI,KI,KAPPA,NU1,NU1IN,NU2,NU2IN,KZ
      COMMON/INPAR/IPA/LAMIN5/RTIP(7)/QPAR3/PERL(3)/IND3/NTYP( 42)
     1       /SUB1/ASCH( 42,3)/GEO0/ACH(3)/INITL/X/SHROUD/TLINER( 18,2)
     2       /SUB21/TSCHA( 18,2),TSCHB( 18,2)/MART2/NS1,NS2/MART3/TBEQR,
     3        TBEQL/ISUP/IQLIN
      COMMON /MART5/ NSTR
      COMMON /SC09C/ IS
      COMMON /SC10C/ ANU( 42,3)
      COMMON /SC15C/ ALFA( 42,3)
      COMMON /SC01Z/ YH( 42,2)
      COMMON /SC16C/ SNU( 18,2)
      COMMON /SC17C/ SALFA( 18,2)
C
      QA=QQ*DVOL/RTIP(IPA)*0.5
      IF ( II .GT. NSTR) TSCHA(II,M)=TI
      IF ( II .GT. NSTR) TSCHB(II,M)=TI
      NTYPI=NTYP(I)
      PW=4.*AREAI*F2ATIP/(DEIR*F2DTIP)
      PH=PW-PERL(ITYP)*ASCH(I,M)/ACH(NTYPI)
      R2=SQRT(RTIP(IPA)**2+2.*RTIP(IPA)*AREAI*F2ATIP/PH)
      DEI=2.*(R2-RTIP(IPA))
      RAS=RTIP(IPA)/R2
      KI=KAPPA(PBT,TI)
      ETAI=ETA(PBT,TI)
      RHOI=RHO(PBT,TI)
      CPI=CP(PBT,TI)
      REI=MASSI*DEI/(AREAI*F2ATIP*ETAI)
      PRI=ETAI*CPI/KI
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      CALL SPANU(REI,PRI,I,M,YYI)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 06.02.1980
      PEI=REI*PRI
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 12.11.1979
C     GRI=X/(DEI*PEI)
C     THE GRAETZ NUMBER IS COMPUTED AT THE SAME REFERENCE TEMPERATURE
C     DEFINED FOR THE REYNOLDS NUMBER USED IN THE CALCULATION OF THE
C     FRICTION FACTOR.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 12.11.1979
C
      TNY=TW1
      PIG=3.141593
      IF(ITYP .NE. 1 .AND. ITCORR .GT. 1)
     >   TNY=TNU(TW1,TLINER(II,M),ITYP,PERL(ITYP),PIG,RTIP(IPA))
      KZ=KAPPA(PBT,TNY)
      ETAZ=ETA(PBT,TNY)
      RHOZ=RHO(PBT,TNY)
      CPZ=CP(PBT,TNY)
      REZ=MASSI*DEI/(AREAI*F2ATIP*ETAZ)
      PRZ=ETAZ*CPZ/KZ
      PEZ=REZ*PRZ
```

```
      GRI=X/(DEI*PEZ)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 12.11.1979
C
C-------- (NU 1)INF IF (Q)LIN =0
C
      IF(ITYP.EQ.1)GOTO 1
      NU1IN=4.07+1.237/RAS**0.80272
      GOTO 2
    1 NU1IN=RAS/(1.+RAS)*(14.1207+4.1261*ALOG(0.952313/RAS-1.))
    2 CONTINUE
C
C-------- YNU1=(NU 1)/(NU 1)INF IF (Q)LINER =0
C
      IF(GRI.GT. 0.025)GOTO 3
      B=-0.19327+.121747/GRI**0.14828
      GOTO 4
    3 B=-0.0013376+0.0000277181/GRI**1.76255
      IF(B.LT.0.)B=0.
    4 YNU1=(RAS/0.00062)**B
C
      NU1=NU1IN*YNU1
CCCCCCCCCCCCCCCCCCCCCCC 18.02.1980
C     NU1=NU1*0.967
      IF(ITYP.NE.1) NU1=NU1*0.967
CCCCCCCCCCCCCCCCCCCCCCC
      ALFA1=NU1*KI/DEI*YYI
      TW1=TI+QA/ALFA1
CCCCCCCCCCCCCCCCCCCCCCC
      ANU(I,M)=NU1*YYI
      YH(I,M)=YYI
      ALFA(I,M)=ALFA1
CCCCCCCCCCCCCCCCCCCCCCC
      TL1=0.
      TETA2=0.
      IF(NTYP(I).EQ.2 .AND. ITYP.EQ.1 .AND. I.GE.NS1 .AND. I.LE.NS2)
     *CALL SIMLA2(TI,TW1,TL1,NU1,TETA2,TBEQR,TBEQL)
      IF(ITYP.EQ.1)RETURN 1
C
C--------CALCULATIONS ONLY FOR THE CORNER CHANNELS AND THE WALL PARTS OF
C--------THE WALL SUBCHANNELS (IF  (Q)LINER =0 )
C
C-------- (TETA 2)INF
C
        IF(RAS.GT. 0.1)GOTO 5
      TETA2I=-0.103313*RAS**0.9489
      GOTO 6
    5 TETA2I=0.0142-0.0784857*RAS**0.4828
    6 CONTINUE
C
C-------- YTE2=(TETA 2)/(TETA 2)INF
C
      IF(GRI.GT. 0.01)GOTO 7
      YTE2=31.105*GRI
      GOTO 9
    7 IF(GRI.GE. 0.025)GOTO 8
      YTE2=15.59936*GRI**0.8501383
      GOTO 9
    8 YTE2=1./(0.98293+0.000125822/GRI**2.242421)
      IF(YTE2.GT.1)YTE2=1.
C
    9 TETA2P=TETA2I*YTE2
      TLINER(II,M)=TETA2P*QA*DEI/KI+TI
      TETA2=TETA2P
      IF(I.GE.NS1 .AND. I.LE.NS2)CALL SIMLA1(TE,TI,TW1,TLINER(II,M),NU1,
     *      TETA2,I,M,TBEQR,TBEQL,II)
      IF(TLINER(II,M).LT.TE)TLINER(II,M)=TE
```

```
C
      IF(ABS(QALIN).LE.1.E-06 .OR. IQLIN.EQ.2)RETURN 1
C
C-------- CASE OF HEATED LINER ( FOR CORNER CHANNELS AND WALL PART OF TH
C-------- WALL SUBCHANNELS) : (NU 2) AND (TETA 1) IF (Q)ROD =0
C
      TETA2=0.
      IF(QA.GT.1.E-06)TETA2=(TLINER(II,M)-TI)*KI/(QA*DEI)
C
C-------- (NU 2)INF
C
      NU2IN=4.754*EXP(0.1246*RAS)
C
C-------- YNU2=(NU 2)/(NU 2)INF
C
      IF(GRI.GT. 0.003)GOTO 11
      YNU2=0.2861/GRI**0.3334
      GOTO 12
   11 YNU2=1.+0.060344/GRI**0.506*EXP(-49.*GRI)
   12 NU2=NU2IN*YNU2
      NU2=NU2*0.967
CCCCCCCCCCCCCCCCCCCCCCCC 16.04.1980
      SNU(II,M)=NU2
      ALFA2=NU2*KI/DEI
      SALFA(II,M)=ALFA2
CCCCCCCCCCCCCCCCCCCCCCCC 16.04.1980
C
C-------- (TETA 1)
C
  100 TETA1=TETA2P/RAS
      TW1=TETA1*QALIN*DEI/KI+TI
      ALFA2=NU2*KI/DEI
      TLINER(II,M)=TI+QALIN/ALFA2
      IF(I.GE.NS1 .AND. I.LE.NS2)CALL SIMLA2(TI,TLINER(II,M),TW1,NU2,
     *     TETA1,TBEQL,TBEQR)
      IF (TW1.LE.TE)TW1=TE
      TETA1=(TW1-TI)*KI/(QALIN*DEI)
C
C--------GENERAL CASE OF HEATED LINER AND ROD (CORNER CHANNELS AND WALL
C--------PART OF THE WALL SUBCHANNELS)
C
      NU2=NU2/(1.+QA/QALIN*TETA2*NU2)
CCCCCCCCCCCCCCCCCCCCCCC
      IF(NU2.GT. 9999.0) NU2= 9999.0
      IF(NU2.LT.-9999.0) NU2=-9999.0
      ALFA2=NU2*KI/DEI
CCCCCCCCCCCCCCCCCCCCCCC
      TLINER(II,M)=TI+QALIN/ALFA2
      IF(QA.LE.1.E-06)RETURN 1
C
      NU1=NU1/(1.+QALIN/QA*TETA1*NU1)
CCCCCCCCCCCCCCCCCCCCCCC
      IF(NU1 .GT. 9999.0) NU1= 9999.0
      IF(NU1 .LT.-9999.0) NU1=-9999.0
      ALFA1=NU1*KI/DEI*YYI
CCCCCCCCCCCCCCCCCCCCCCC
      TW1=TI+QA/ALFA1
      RETURN 1
      END
```

```
      SUBROUTINE TESTB (INDICE)
C ----------------------------------------------------------------------
C     TEST OF CONVERGENCE FOR THE BULK TEMPERATURES.
C
      COMMON /SUB2/   TSCH( 42,3),MSCH( 42,3)
      COMMON /HEA6 /  NPIN( 42),JPIN( 42,3)
      COMMON /SC14C/  TBOLD( 42,3)
      COMMON /GASD1/  NSTOT
      COMMON /MART5/  NSTR
C
      PERC=0.005
      INDICE=0
C
      DO 1000 NS=1,NSTOT
              NW=NS-NSTR
              NP=NPIN(NS)
              DO 500 M=1,NP
                     DELT1=ABS(TBOLD(NS,M)-TSCH(NS,M))
                     IF ( DELT1 .GT. 1.) INDICE=INDICE+1
  500                CONTINUE
 1000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE TESTNU(IND)
C -----------------------------------------------------------------------
C     ONLY IN CASE OF CONVERGENCE PROBLEMS : AVERAGING OF THE NU NUMBERS
C
      COMMON /SC10C/ ANU( 42,3)
      COMMON /SC15C/ ALFA( 42,3)
      COMMON /SC16C/ SNU( 18,2)
      COMMON /SC17C/ SALFA( 18,2)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /GASD1/ NSTOT
      COMMON /MART5/ NSTR
C
      DIMENSION ANO( 42,3),SNO( 18,2),ALO( 42,3),SAO( 18,2)
C
      IF ( IND .GT. 0 ) GOTO 500
C     -------------------------------------------------------------------
C     IND LE 0 : THE NUSSELTS OF THE PRECEDENT ITERATION ARE MEMORIZED
      DO 200 NS=1,NSTOT
             NP=NPIN(NS)
             DO 100 M=1,NP
                    ANO(NS,M) = ANU(NS,M)
                    ALO(NS,M) = ALFA(NS,M)
                    IF(NS.LE.NSTR) GOTO 100
                    NW=NS-NSTR
                    SNO(NW,M) = SNU(NW,M)
                    SAO(NW,M) = SALFA(NW,M)
  100               CONTINUE
  200        CONTINUE
      GOTO 9999
C     -------------------------------------------------------------------
C     IND GT 0 : CONVERGENCE PROBLEMS: THE NUSSELT ARE AVERAGED
  500 CONTINUE
      DO 700 NS=1,NSTOT
             NP=NPIN(NS)
             DO 600 M=1,NP
                    ANU(NS,M)   = (ANU(NS,M)   +ANO(NS,M))*0.5
                    ALFA(NS,M)  = (ALFA(NS,M)  +ALO(NS,M))*0.5
                    IF(NS.LE.NSTR) GOTO 600
                    NW=NS-NSTR
                    SNU(NW,M)   = (SNU(NW,M)   +SNO(NW,M))*0.5
                    SALFA(NW,M) = (SALFA(NW,M)+SAO(NW,M))*0.5
  600               CONTINUE
  700        CONTINUE
C
 9999 RETURN
      END
```

```
      SUBROUTINE TESTW  (INDICE)
C -------------------------------------------------------------------
C     TEST OF CONVERGENCE ON THE WALL TEMPERATURES.
C
      COMMON /SUB22/  TW( 42,3)
      COMMON /HEA6 /  NPIN( 42),JPIN( 42,3)
      COMMON /GASD1/  NSTOT
      COMMON /SC99C/  TD( 42,3)
      COMMON /SC08C/  TLD( 18,2)
      COMMON /MART5/  NSTR
      COMMON /SHROUD/ TLINER( 18,2)
C
      PERC=0.005
      INDICE=0
C
      DO 4000 NS=1,NSTOT
              NW=NS-NSTR
              NP=NPIN(NS)
              DO 3000 M=1,NP
                     DELT1=ABS(TD(NS,M)-TW(NS,M))
                     IF (DELT1 .GT. 1.) INDICE=INDICE+1
C
                     IF (NS.LE.NSTR) GO TO 3000
                     DELT1=ABS(TLINER(NW,M)-TLD(NW,M))
                     IF (DELT1 .GT. 1.) INDICE=INDICE+1
 3000                CONTINUE
 4000         CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE TEST1(NCA,X,XF,KK)
C -------------------------------------------------------------------
C     PERFORMS THE TEST OF CONVERGENCE ON THE CONDUCTIVITY OF THE FUEL.
C
      DIMENSION X(13),XF(13)
C
      REAL KFUEL
C
      KK=0
      NC1=NCA+1
      NC2=NCA*2+1
      DO 100 K=NC1,NC2
C            DELTA=ABS(X(K)-XF(K))
C            IF ( DELTA .GT. 1.) KK=KK+1
             DELTA=ABS(KFUEL(X(K))/KFUEL(XF(K))-1.0)
             IF (DELTA.GT.0.01) KK=KK+1
             XF(K)=X(K)
  100        CONTINUE
      RETURN
      END
```

```
      FUNCTION TIS(R1,R2,INU)
C-----------------------------------------------------------------------
C     TIS EVALUATES THE CORRECTION FACTOR FOR THE NUSSELT NUMBERS IN
C     THE REGION WHERE THE TEMPERATURE PROFILE IS NOT YET FULLY
C     DEVELOPED ( CASE OF TURBULENT FLOW )
C     INU=1 : FOR SMOOTH RODS
C     INU=2 : FOR ROUGH RODS
C     INU=3 : FOR SMOOTH LINER
C
      COMMON/INITL/X
      GOTO(1,2,3),INU
    1 TSI=0.86+0.8*(2.*(R2-R1)/X)**0.4*(R1/R2)**0.2
      GOTO 4
C
C     ******************************************************************
C     NO EQUATIONS ARE AVAILABLE AT THE MOMENT FOR THE INLET EFFECT IN
C     THE CASE OF ROUGHENED RODS: THUS, AT INU=2, TIS=1 IS IMPOSED
C     ******************************************************************
    2 TSI=1.
      GOTO 4
C
    3 TSI=0.86+0.54*(2.*(R2-R1)/X)**0.4*(1.+0.48*(R1/R2)**0.37)
C
C
    4 IF(TSI.LE.1.)TSI=1.
      TIS=TSI
      RETURN
      END
```

```
      SUBROUTINE TLINE(I,AI,ITTEMP,NS,K,ALFA,D,W,RH,DET,PROV,IRH,DAI,DBI
     *,AAI,ABI,RHPL,G,TWI,TE,QPLUS,ETAA,RHOA,ETAB,RHOB,ETAIW,RHOIW,ANGT,
     *EM1,XC1,XC2,T1,*,CS)
C-----------------------------------------------------------------------
C     SUBROUTINE TLINE EVALUATES THE POSITION OF THE TAU=0 LINE FOR EACH
C     "WALL-TYPE" SUB-SUBCHANNEL
C
      COMMON/REC1/      PVERT(90),PRAD(90)/REC2/E(90)/REC3/P(90)
      NNN=20
      SSCHFA=19.0986*ALFA
      XIRH=IRH
      I1=I-1
 8400 IF(I.GT.1)GOTO 1
C     .................................................................
C     STARTING POINT (F(P),P) FOR THE 1.ST SUB-SUBCHANNEL
C
      P1=1.0001-(W/D-1.)*0.39*(2.-XIRH)
      XX=0.39
      GOTO 2
C     .................................................................
C     STARTING POINT (F(P),P) FOR THE I.TH SUB-SUBCHANNEL ( I>1 )
C
    1 P1=P(I1) +0.08*(W/D-1.)*SSCHFA
      XX=-0.04*SSCHFA
C     .................................................................
C     RESEARCH OF TWO CONSECUTIVE POINTS (F(P),P) AT WHICH F= FAI-FBI
C     HAS DIFFERENT SIGNS ( ITERATION LOOP ITAU1 )
C
    2 CONTINUE
      DO 4 ITAU1=1,NNN
      P2=P1+XX*(W/D-1.)
      CALL TAU(I,AI,P2,ALFA,D,W,RH,DET,PROV,IRH,DAI,DBI,PAI,F2,RHPL,TWI,
     *TE,ITTEMP,QPLUS,ETAA,RHOA,ETAB,RHOB,ETAIW,RHOIW,ANGT,EM1,XC1,XC2,
     2T1,&8500,CS)
      IF(ITAU1.EQ.1)GOTO 3
      IF(F1*F2.LE.0.)GOTO 6
    3 F1=F2
    4 P1=P2
C     .................................................................
C     TWO CONSECUTIVE POINTS AT WHICH F =FAI-FBI HAS  DIFFERENT SIGNS
C     HAVE BEEN NOT FOUND : IT WILL BE TRIED TO START CLOSER TO THE RODS
C     ( IF IT HAS NOT YET BEEN TRIED AND IF IT IS I>1 )
C
      WRITE(6,5)I,ITTEMP,NS,K
    5 FORMAT(5X,'STOP IN TLINE IN LOOP ITAU1    FOR SUBCH.',I3,2X,'(ITTE
     *MP=',I2,')OF CHANNEL',I4,2X,'(AXIAL SECTION NR.',I4,')'/130('*'))
      IF(NNN.EQ.40)RETURN 1
      NNN=40
      IF(I.GT.2)I1=I-2
      GOTO 8400
C     .................................................................
C     TWO CONSECUTIVE POINTS (F(P),P) HAVE BEEN FOUND, AT WHICH
C     F= FAI-FBI HAS  DIFFERENT SIGNS; THE VALUE OF P AT WHICH F=0 WILL
C     BE NOW RESEARCHED BY MEANS OF THE TANGENT METHOD ( ITERATION LOOP
C     ITAU2 )
C
    6 CONTINUE
      DO 8  ITAU2=1,30
      PP=P1-F1*(P2-P1)/(F2-F1)
      CALL TAU(I,AI,PP,ALFA,D,W,RH,DET,PROV,IRH,DAI,DBI,PAI,F ,RHPL,TWI,
     1TE,ITTEMP,QPLUS,ETAA,RHOA,ETAB,RHOB,ETAIW,RHOIW,ANGT,EM1,XC1,XC2,
```

```
     2T1,&8500,CS)
      IF(ABS(PP/P1-1.).LE.1.E-04 .OR. ABS(PP/P2-1.).LE.1.E-04) GOTO 10
      IF(F*F1.GE.0.)GOTO 7
      F2=F
      P2=PP
      GOTO 8
    7 F1=F
      P1=PP
    8 CONTINUE
C     .................................................................
C     PROBLEMS IN FINDING THE POSITION OF THE TAU=0 LINE
C
      WRITE(6,9)I,ITTEMP,NS,K
    9 FORMAT(5X,'STOP  IN TLINE IN LOOP ITAU2    FOR SUBCH.',I3,2X,'(ITT
     *EMP=',I2,')OF CHANNEL',I4,2X,'(AXIAL SECTION NR.',I4,')')
 8500 RETURN 1
C     .................................................................
C     THE POSITION OF THE TAU=0 LINE HAS BEEN FOUND FOR SUB-SUBCHANNEL I
C     SOME GEOMETRIC PARAMETERS WILL BE NOW COMPUTED
C
C
   10 PBI=ALFA*D*0.5
      AAI=DAI*PAI*0.25
      ABI=DBI*PBI*0.25
      P(I)=PP
      EPS=SQRT(1.+DBI/D)
      G=GSTAR(EPS)
      RETURN
      END
```

```
      SUBROUTINE TMCF(I,NI,TT,TOTM,MAVI)
C-----------------------------------------------------------------------
C     TMCF EVALUATES THE AVERAGE CROSS-FLOW TEMPERATURES  FOR THE
C     CROSS-FLOW EXCHANGE BETWEEN CHANNELS
C
      REAL MAV,MAVI,MAVJ
      COMMON /MOB5/ T( 42)/MOB8/DP( 42)/MOB6/MAV( 42)
      COMMON/IJ1/NER( 42),NIS( 42,3)
     1      /GEO0/ACH(3)/IND3/NTYP( 42)/GEN2/A( 42)
     2      /HEA6/NPIN( 42),JPIN( 42,3)/GAMAR/CXX
      TT=0.
      TOTM=0.
      DO 2 M=1,NI
      J=NIS(I,M)
      NTYPJ=NTYP(J)
      MAVJ=MAV(J)*ACH( NTYPJ)/A(J)
      CXX=0.5
      IF((NTYP(I)+NPIN(I).EQ.4) .OR. (NTYP(J)+NPIN(J).EQ.4))CXX=1.
      CALL CF1(T(I),T(J),MAVI,  MAVJ,  DP(I),DP(J),2,TT,TOTM)
    2 CONTINUE
      CXX=1.
      TT =TT/TOTM
      RETURN
      END
```

```
      FUNCTION TME(PBT,M1,M2,T1,T2,LAM1,LAM2,A1,A2,CTURB)
C-----------------------------------------------------------------------
C     TME EVALUATES THE MASS FLOW RATES PER UNIT LENGTH EXCHANGED DUE
C     TO TURBULENCE
C
      REAL M1,M2,LAM1,LAM2
      T12=(T1*M1+T2*M2)/(M1+M2)
      RHO12=RHO(PBT,T12)
      RHO1=RHO(PBT,T1)
      RHO2=RHO(PBT,T2)
      UAST12=(SQRT(LAM1*0.125)*M1/RHO1+SQRT(LAM2*0.125)*M2/RHO2)/(A1+A2)
      TME=CTURB*RHO12*UAST12
      RETURN
      END
```

```
      SUBROUTINE TMPUN(NSTOT,NSTR,TE,PE,PEBAR,TE1,PE1,PE1BAR,
     *INDPR,MFLOW,IPAST,IPAEND,IREAD1,XLAM1,STLEN,*)
C
C     TMPUN PUNCHS THE CARDS WHICH MUST BE CHANGED TO START A NEW
C     CALCULATION STEP (PUNCHING UNITY=1)
C     THE ACTUAL CALCULATION STEP IS STOPPED BECAUSE THE ALLOWED
C     CALCULATION TIME TIMEPU HAS BEEN ELAPSED  OR BECAUSE THE END
C     OF THE AXIAL PORTION IPAEND (IPAEND<7) HAS BEEN OVERTAKEN
C     -----------------------------------------------------------------
      REAL MFLOW,MI,MSCH1,MSCWC1
      DIMENSION XLAM1(7)
      COMMON/GEN3/MI( 42)
      COMMON /GEN4/TEMP( 42)/SUB6/TSCH1( 42,3)/SUB8/MSCH1( 42,3)
     1       /WCSE2/MSCWC1( 18,2,2)/WCSE5/TSCWC1( 18,2,2)/IND3/NTYP( 42)
     2       /HEA6/NPIN( 42),JPIN( 42,3)
C
C     10TH CARD:
      IF(INDPR.EQ.1)GOTO 1
      PE=PEBAR
      PE1=PE1BAR
    1 WRITE(1,2)PE,PE1,TE,TE1,MFLOW,(XLAM1(I),I=1,3)
    2 FORMAT(8F10.5)
C
C     13TH CARD
      WRITE(1,3)IPAST,IPAEND,IREAD1
    3 FORMAT(3I10)
C
C     14TH CARD
      WRITE(1,2)STLEN
C
C     LAST BLOCK OF CARDS
      WRITE(1,4)(MI(NS),TEMP(NS),NS=1,NSTOT)
      DO 5 NS=1,NSTOT
      NSW=NS-NSTR
      NP=NPIN(NS)
      WRITE(1,4)(MSCH1(NS,M),TSCH1(NS,M),M=1,NP)
      IF(NTYP(NS).EQ.2)WRITE(1,4)((MSCWC1(NSW,M,JWC),TSCWC1(NSW,M,JWC),
     *                 JWC=1,2),M=1,2)
    4 FORMAT(8F10.5)
    5 CONTINUE
      RETURN 1
      END
```

```
      SUBROUTINE TNEW (NCA,X,J,NSE)
C ------------------------------------------------------------------
C     ASSIGNES THE NEW VALUES OF THE CLAD TEMPERATURE (X) TO ARRAY TW
C
      COMMON /SC01C/ NCAN( 19),LIPS( 19,10)
      COMMON /SC05C/ JZUR( 19, 42)
      COMMON /SUB22/ TW( 42,3)
C
      DIMENSION X(NSE)
C
      DO 100 K=1,NCA
             NS1 = LIPS(J,K)
             MZ1 = JZUR(J,NS1)
             TW(NS1,MZ1) = X(K)
  100        CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TNEWS(X)
C ----------------------------------------------------------------------
C     ASSIGNES THE NEW VALUES TO THE SHROUD TEMPERATURES.
C
      DIMENSION X( 34)
C
      COMMON /SHROUD/ TLINER( 18,2)
      COMMON /SC22C/  NTOT
      COMMON /MART5/  NSTR
      COMMON /GASD1/  NSTOT
      COMMON /HEA6/   NPIN( 42),JPIN( 42,3)
C
      N=0
      NW1=NSTR+1
      DO 1000 NS=NW1,NSTOT
              NW=NS-NSTR
              NP=NPIN(NS)
              DO 500 M=1,NP
                     N=N+1
                     TLINER(NW,M)=X(N)
  500                CONTINUE
 1000         CONTINUE
C
      IF(N.EQ.NTOT) GO TO 9999
      WRITE(6,2000) N,NTOT
 2000 FORMAT( ' SUB. TNEWS; N = ',I4,'  NTOT = ',I4,/5X,
     >        ' CALCULATION STOPS.')
      STOP
C
 9999 RETURN
      END
```

```
      FUNCTION TNU(TW,TL,ITYP,PERL,PIG,RTIP)
C     -----------------------------------------------------------------
C     TNU EVALUATES THE TEMPERATURE AT WHICH REYNOLDS NUMBERS MUST
C     COMPUTED FOR THE CALCULATION OF LAMINAR FRICTION FACTORS
C
      LPIG=ITYP**2-ITYP
      PHR=RTIP*PIG/LPIG*2.
      TNU=(TL*PERL+TW*PHR)/(PERL+PHR)
      RETURN
      END
```

```
      SUBROUTINE TORG (VDIAM,RINT,PIG,H)
C ----------------------------------------------------------------------
C     ORGANIZES THE CALCULATION OF THE THERMAL CONDUCTION WITHIN THE PIN
C
      COMMON /LAMIN5/RTIP(7)
      COMMON /INPAR/ IPA
      COMMON /CEV04/ LAMOP3
      COMMON /SC05R/ IRAD
      COMMON /SC19C/ ICON
C
C     1  --------------------------------------------------------------
C     CORRECTION FOR LAMINAR FLOW
C
      VDIA1=VDIAM
      IF(LAMOP3.EQ.1) VDIA1=RTIP(IPA)*2.
C
C     2  --------------------------------------------------------------
C     DECIDES WICH CONDUCTION MODEL IS TO BE USED:
C     TWO MODELS ARE AVAILABLE DEPENDING ON THE OPTION ICON :
C      ICON =    1   ==> CONDUCTION IN THE CLAD ONLY.
C                        (POWER GENERATED IN THE CAN)
C      ICON =   -1   ==> CONDUCTION IN THE CLAD AND IN THE FUEL.
C                        (POWER GENERATED IN THE FUEL)
C      ICON =    0   ==> CONDUCTION IS NOT TAKED INTO ACCOUNT.
C
      IF (ICON) 100,300,200
C
C     3  --------------------------------------------------------------
C     CONDUCTION IN THE CLAD AND IN THE FUEL.
C
  100 CALL BOTH (VDIAM,PIG,H,VDIA1,RINT)
      GO TO 9999
C
C     4. --------------------------------------------------------------
C     CONDUCTION WITHIN THE CLAD ONLY.
C
  200 CALL LEIST (VDIAM,PIG,H,VDIA1,RINT)
      GO TO 9999
C
C
C     5. --------------------------------------------------------------
C     THE USE OF A CONDUCTION MODEL IS NECESSARY WHEN YOU WISH CONSIDER
C     THE RADIATION EFFECT (IRAD=1)
C
  300 IF (IRAD.NE.1) GO TO 9999
      WRITE(6,330)
  330 FORMAT( 5X,'THE OPTION ICON=0 IS NOT ALLOWED WHEN IRAD =1.',
     >       /5X,'CALCULATION STOPS')
C
C
 9999 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TOTGEO(NSEL,D,C,Z,PIG,NEXCON,NRODS,W,WA,ZA,EM1,PERLT,
     &RTIP)
C-----------------------------------------------------------------------
C     TOTGEO CALCULATES FLOW AREAS , EQUIVALENT DIAMETERS AND OTHER
C     GEOMETRIC DATA FOR THE WHOLE BUNDLE FLOW SECTION , FOR THE
C     CHANNELS AND FOR THE SUBCHANNELS
C
C                 VERSION FOR HEXAGONAL BUNDLES
C     .................................................................
      COMMON/GEO0/ACH(3)/LAMIN2/FATIP(3),FDTIP(3)/QPAR3/PERL(3)
     1         /GEO2/ATOT,DETOT,ASEC/GEO5/ATC,DETC,ATW,DETW,ATA,DETA,AAC,
     2         AAW,AAA/WAKAO/CD,WD,ZD,ZWCD,AWD2,PWWD/GASD3/FSYMM
      SQ3=SQRT(3.)
      W=Z+D*0.5
      WA=W
      ZA=Z
      EXCON=NEXCON
      RODS=NRODS
      EM2=C*0.5-EM1
      ZWC=EM2/SQ3
      DTIP=RTIP*2.
      SIDE=EXCON*C+(2.*W-D)/SQ3
      RPER=RODS*PIG*D
      PERLT=6.*SIDE+EXCON*(-12.*EM2+24.*ZWC)
      ATOT=3.*SQ3/2.*SIDE**2-RPER*D/4.-6.*EM2*ZWC*EXCON
      DETOT=4.*ATOT/(RPER+PERLT)
      GOTO(20,21,22,24),NSEL
   20 ASEC=ATOT
      GOTO 23
   21 ASEC=ATOT*0.5
      GOTO 23
   22 ASEC=ATOT/12.
C     EXTENDED AT GA FOR OTHER SYMMETRY SECTIONS (NSEL=4)
      GOTO 23
   24 ASEC=ATOT/FSYMM
   23 CONTINUE
      ATC=(C**2*SQ3-PIG*D**2/2.)/4.
      DETC=4.*ATC/(PIG*D/2.)
      ATW=C*(W-D/2.)-D**2*PIG/8.-EM2*ZWC
      DETW=4.*ATW/(PIG*D*0.5+2.*EM1+4.*ZWC)
      ATA=(W-D/2.)**2/SQ3-D**2*PIG/24.
      DETA=4.*ATA/(D*PIG/6.+(W-D/2)*2./SQ3)
      AAC=ATC/6.
      AAW=ATW*0.5
      AAA=ATA*0.5
      ACH(1)=ATC
      ACH(2)=ATW
      ACH(3)=ATA
      PERL(1)=0.
      PERL(2)=4.*ATW/DETW-0.5*PIG*D
      PERL(3)=4.*ATA/DETA-PIG*D/6.
      FATIP(1)=(C**2*SQ3-PIG*DTIP**2*0.5)*0.25
      FDTIP(1)=4.*FATIP(1)/(PIG*0.5*DTIP)/DETC
      FATIP(1)=FATIP(1)/ATC
      FATIP(2)=C*(W-DTIP*0.5)-DTIP**2*PIG*0.125-EM2*ZWC
      FDTIP(2)=4.*FATIP(2)/(PIG*DTIP*0.5+2.*EM1+4.*ZWC)/DETW
      FATIP(2)=FATIP(2)/ATW
      FATIP(3)=(W-DTIP*0.5)**2/SQ3-DTIP**2*PIG/24.
      FDTIP(3)=4.*FATIP(3)/(DTIP*PIG/6.+(W-DTIP*0.5)*2./SQ3)/DETA
      FATIP(3)=FATIP(3)/ATA
```

```
      CD=C/DTIP
      WD=W/DTIP
      ZD=Z/DTIP
      ZWCD=ZWC/DTIP
      AWD2=AAW*FATIP(2)/DTIP**2
      PWWD=4.*AAW*FATIP(2)/(DETW*FDTIP(2)*DTIP)
      WRITE(6,1)ATOT,DETOT,ASEC
      WRITE(6,3)ATC,ATW,ATA,DETC,DETW,DETA
    1 FORMAT(/// 5X,'TOTAL FLOW AREA=',F10.2,1X,'SQCM'/5X,'TOTAL EQUIVAL
     1ENT DIAMETER=',F10.1,1X,'CM'/5X,'FLOW AREA OF SECTION=',F10.2,1X,'
     *SQCM'/)
    3 FORMAT(5X,'FLOW AREAS OF CHANNELS:'/5X,  'CENTRAL=',F10.2/5X,'WALL
     *=',F10.2/5X,'CORNER=',F10.2//5X,'EQUIVALENT DIAMETERS'/5X,'CENTRAL
     *=',F10.1/5X,'WALL=',F10.1/5X,'CORNER=',F10.1////130('*'))
      RETURN
      END
```

```
      SUBROUTINE TOTSEC(NSEL)
C ---------------------------------------------------------------------
C     COMPUTES THE TOTAL NUMBER OF SECTOR FOR THE RADIATION CALCULATIONS
C
      COMMON /GASD1/ NSTOT
      COMMON /MART5/ NSTR
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /SC01R/ NSECT,NSECP
C
      IF(NSEL.NE.1) GO TO 3000
C
      NSECP=0
      DO 1000 NS=1,NSTOT
              NP=NPIN(NS)
              DO 800 M=1,NP
                     NSECP=NSECP+1
  800                CONTINUE
 1000         CONTINUE
C
      NSECT=NSECP
      NSW=NSTR+1
      DO 2000 NS=NSW,NSTOT
              NP=NPIN(NS)
              DO 1800 M=1,NP
                      NSECT=NSECT+1
 1800                 CONTINUE
 2000         CONTINUE
      GO TO 4000
C
 3000 CONTINUE
      IF(NSEL.EQ.3) GO TO 9999
      WRITE(6,3500)
 3500 FORMAT( ' SUB TOTSEC - THE PRESENT VERSION CANNOT RUN WITH THE',
     >        ' ACTUAL NRODS,NSEL. CALCULATION STOPS')
      STOP
C
 4000 WRITE(6,5000) NSECP,NSECT
 5000 FORMAT( 5X,'SUB. TOTSEC ',
     >           'NUMBER OF SECTORS OF PIN = ',I5,
     >           'TOTAL NUMBER OF SECTORS  = ',I5,//)
C
 9999 RETURN
      END
```

```
      SUBROUTINE TRICA1(K,NS,NN,IRH,PROV,PB,    RH,A,DE,MEC,AT,DET,DETOT,
     *H1,ALFA, H,M,PR1,PR2,SQDPG,TE,SUR,D,AMT,DDDD,ATSCH,*,C)
C-----------------------------------------------------------------------
C     SUBROUTINE TRICA1 CALCULATES FRICTION FACTORS AND APPROXIMATE
C     OUTLET MASS FLOW RATES AND TEMPERATURES FOR CENTRAL SUBCHANNELS
C
      REAL MEC,LAMSCH,KAPPA,LAMLAM,MSCH
      COMMON/SUB5/LAMSCH( 42,3)/CEN1/G(46)/SUB1/ASCH( 42,3)/INPAR/IPA
     1      /SUB6/TSCH1( 42,3)/CORR1/SIGMAI( 42,3),PHII( 42,3)
     2      /LAMIN5/RTIP(7)/DAT/PIG/SUB23/HPLUSB( 42,3),HPLUSW( 42,3)
     3      ,QPLUS( 42,3),PRB ( 42,3),YODH( 42,3)/HEA5/QQ( 42,3)
     4      /WSSCHO/TBSSC1( 42,3),TWSSC1( 42,3),
     A       TBSSC2( 42,3),TWSSC2( 42,3)
     5      /LAMIN0/I2TIP( 42,3)/LAMIN1/AKAPPA( 42) /LAMIN2/FATIP(3),
     6      FDTIP(3)/LAMIN3/F1ATIP( 42),F1DTIP( 42)/LAMIN4/F2ATIP( 42,3)
     7      ,F2DTIP( 42,3)/LAMIN7/F1PTIP/GEN2/ACHA( 42)/SUB2/TSCH( 42,3)
     8      ,MSCH( 42,3) /SUB22/TW( 42,3)/MART/ITCORR
     9      /LAMIN9/I3TIP( 42,3)
      DIMENSION A(30),DE(30),MEC(30)
      COMMON /HEA6 / NPIN( 42),JPIN( 42,3)
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC13C/ GEO1( 42,3)
C
      IF(M.GT.1)GOTO 2998
      F1ATIP(NS)=0.
      F1PTIP=0.
 2998 CONTINUE
      I2TIP(NS,M)=I3TIP(NS,M)
      IF(                    I2TIP(NS,M).EQ.1)GOTO 2999
C     ..................................................................
C     ..................................................................
C     I3TIP#1: THE TURBULENT CALCULATION MUST BE PERFORMED
C
      TWIAV=0.
      AMT=0.
      TT=0.
      DDDD=0.
      HPLUS1=0.
      HPLUS2=0.
C     ..................................................................
C     SUB-SUBCHANNEL CALCULATIONS ( I = SUB-SUBCHANNEL INDEX )
C
      DO 1 I=1,NN
      AM1=MEC(I)
      AA=A(I)
      DD=DE(I)
      GG=G(I)
      CALL CEWA(K,NS,IRH,PROV,PB,RH,AA,DD,GG,AM1,DETOT,H1,ALFA,I,M,H,PR1
     *,PR2,SQDPG,AMT,TT,DDDD,TE,SUR,1,III,HPLUS1,HPLUS2,TSCH1(NS,M),
     *SIGMAI(NS,M),PHII(NS,M),&777,D,TWI,TI,C)
      TWIAV=TWIAV+TWI*ALFA
      TBSSC2(NS ,M  )=TI
      TWSSC2( NS,M  )=TWI
      IF (I.GT.1) GOTO 1
      TBSSC1( NS,M  )=TI
      TWSSC1( NS,M  )=TWI
    1 CONTINUE
C     ..................................................................
C     ALL SUB-SUBCHANNELS HAVE BEEN COMPUTED; AVERAGE SUB-SUBCHANNEL
C     VARIABLES WILL BE NOW COMPUTED
C
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCC 21.09.1979
C       TWIAV=TWIAV*12./PIG
CCCCCCCCCCCCCCCCCCCCCCCCCC
        TWIAV=TWIAV*6./PIG
CCCCCCCCCCCCCCCCCCCCCCCCCC
        ATSCH=TT/AMT
      RHOT=RHO(PB,ATSCH)
      LAMSCH(NS,M)=((AT/DDDD)**2)*2.*DET*RHOT/H
      I2TIP(NS,M)=0
      F2ATIP(NS,M)=1.
      F2DTIP(NS,M)=1.
      IF(I3TIP(NS,M).EQ.2)GOTO 3000
C     ...............................................................
C     I3TIP=3: THE LAMINAR CALCULATION MUST BE ALSO PERFORMED
C
      IF(ITCORR.GT.1)GOTO 2999
      MSCH(NS,M)=AMT*ASCH(NS,M)/AT
      TSCH(NS,M)=ATSCH
      TW(NS,M)=TWIAV
C     ...............................................................
C     ...............................................................
C     FOR I3TIP=1 OR I3TIP=3
C
 2999 CONTINUE
      RELA=RELAM(ASCH(NS,M)*FATIP(1),DET*FDTIP(1),PB,TSCH(NS,M),TW(NS,M)
     &           ,MSCH(NS,M),0.,1,0.,1.)
      LAMLAM=AKAPPA(NS)/RELA
      ROCEN=C*SQRT(SQRT(3.)/(2.*PIG))
      CALL ENTRFR(K,1,1,RTIP(IPA),ROCEN,R2CEN,NS,III,M,DET*FDTIP(1),
     *           ASCH(NS,M)*FATIP(1),MSCH(NS,M),PB,TSCH(NS,M),LAMLAM)
      IF(                    I2TIP(NS,M).EQ.1)GOTO 2997
C     ...............................................................
C     I3TIP=3: SAGAPO DECIDES WHETHER THE FLOW IS LAMINAR OR TURBULENT
C
      IF(LAMSCH(NS,M).GT.LAMLAM)GOTO 3000
C     THE FLOW IS LAMINAR
C
 2997 CONTINUE
      LAMSCH(NS,M)=LAMLAM
      DDDD=AT*FATIP(1)/SQRT(LAMLAM*H/(2.*DET*FDTIP(1)*
     *RHO(PB,TSCH(NS,M))))
      AMT=MSCH(NS,M)*AT/ASCH(NS,M)
      ATSCH=TSCH(NS,M)
      I2TIP(NS,M)=1
      F2ATIP(NS,M)=FATIP(1)
      F2DTIP(NS,M)=FDTIP(1)
      HPLUSB(NS,M)=1.
      HPLUSW(NS,M)=1.
      QPLUS(NS,M)=1.
      PRB  (NS,M)=1.
      YODH(NS,M)=1.
      TBSSC1( NS,M  )=TSCH(NS,M)
      TBSSC2(NS ,M  )=TSCH(NS,M)
      TWSSC1( NS,M  )=TW(NS,M)
      TWSSC2( NS,M  )=TW(NS,M)
C     ...............................................................
C     FOR LAMINAR AND FOR TURBULENT FLOW ( HERE COMES THE CALCULATION
C     IN THE CASE OF TURBULENT FLOW )
C
 3000 CONTINUE
      F1ATIP(NS)=F1ATIP(NS)+ASCH(NS,M)/ACHA(NS)*F2ATIP(NS,M)
       F1PTIP=F1PTIP+ASCH(NS,M)/ACHA(NS) *F2ATIP(NS,M)/F2DTIP(NS,M)
      F1DTIP(NS)=F1ATIP(NS)/F1PTIP
      IF(IRH.EQ.1 .OR.      I2TIP(NS,M  ).EQ.1)RETURN
C     ...............................................................
C     FOR TURBULENT FLOW AND ROUGHENED RODS
```

```
C
      HPLUSB(NS,M)=HPLUS1/AT
      HPLUSW(NS,M)=HPLUS2/AT
      CPT=CP(PB,ATSCH)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     QPLUS(NS,M)=QQ(NS,M)*AT/(SUR*AMT*CPT*(TE+273.16))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      JP=JPIN(NS,M)
      QAD=  QQ(NS,M)
      IF (QDEV .GT. 1.E-06) QAD=QJ(JP,NS)*GEO1(NS,M)/(H1*QDEV)+QQ(NS,M)
      QPLUS(NS,M)=QAD*AT/(SUR*AMT*CPT*(TE+273.16))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      PRB  (NS,M)=ETA(PB,ATSCH)*CPT/KAPPA(PB,ATSCH)
      YODH(NS,M)=0.5*(SQRT(D**2+DET*D)-D)/RH
      RETURN
  777 RETURN 1
      END
```

```
      SUBROUTINE TTOT(INDEX)
C ----------------------------------------------------------------------
C     MEMORIZATION OF THE OLD BULK (INDEX=1) OR WALL (INDEX=2) TEMP.
C
      COMMON /SUB22/  TW( 42,3)
      COMMON /SUB2 /  TSCH( 42,3),MSCH( 42,3)
      COMMON /SC99C/  TD( 42,3)
      COMMON /SC14C/  TBOLD( 42,3)
      COMMON /SC08C/  TLD( 18,2)
      COMMON /MART5/  NSTR
      COMMON /SHROUD/ TLINER( 18,2)
      COMMON /GASD1/  NSTOT
      COMMON /HEA6 /  NPIN( 42),JPIN( 42,3)
C
      IF (INDEX .EQ. 1) GO TO 300
C
      DO 200 NS=1,NSTOT
             NW=NS-NSTR
             NP=NPIN(NS)
             DO 100 M=1,NP
                    TD(NS,M)=TW(NS,M)
                    IF(NS.GT.NSTR) TLD(NW,M)=TLINER(NW,M)
  100               CONTINUE
  200        CONTINUE
      GO TO 9999
C
  300 DO 500 NS=1,NSTOT
             NW=NS-NSTR
             NP=NPIN(NS)
             DO 400 M=1,NP
                    TBOLD(NS,M)=TSCH(NS,M)
  400               CONTINUE
  500        CONTINUE
C
 9999 CONTINUE
      RETURN
      END
```

```
      FUNCTION TUBENU(REI,PRI)
C-----------------------------------------------------------------------
C     TUBENU EVALUATES THE NUSSELT NUMBER OF A TUBE WITH THE SAME REYNOL
C     AND PRANDTL NUMBERS AS THE ANNULAR SECTION WHOSE CROSS SECTIONAL F
C     AREA IS EQUAL TO THE ACTUAL AREA ( TURBULENT FLOW, SMOOTH RODS )
C
      A=1.07+900./REI-0.63/(1.+10.*PRI)
      FTU=1./(1.82*ALOG10(REI)-1.64)**2
      TUBENU=FTU*0.125*REI*PRI/(A+12.7*SQRT(FTU*0.125)*(PRI**(2./3.)-1.)
     *)
      RETURN
      END
```

```
      FUNCTION TURBWC(CTU3,E,PRAD,D,W,C,GAMMA,A1,A2,DE1,DE2,EM1)
C-----------------------------------------------------------------------
C     TURBWC EVALUATES THE GEOMETRIC CONSTANTS FOR THE TURBULENT
C     EXCHANGE BETWEEN THE TWO PORTIONS OF THE WALL SUBCHANNELS
C
      SINGAM=SIN(GAMMA)
      COSGAM=COS(GAMMA)
      PERSEP=PRAD-0.5*D
      Z=W-D*0.5
      EM2=C*0.5-EM1
      ZWC=EM2/SQRT(3.)
      A3=EM2*ZWC*0.5
      D3=D**3
      C2=C**2
      E2=E**2
      Z2=Z**2
      YB3=C*0.5-EM2/3.
      XB3=Z-ZWC/3.
      YB1=(0.25*(C2*(0.5*Z-E/3.)+D3*(SINGAM-1.)/6.)-YB3*A3)/A1
      XB1=(0.25*(C*(Z2-E2/3.)-D3*COSGAM)-XB3*A3)/A1
      YB2=(E*C2-D3*SINGAM*0.5)/(12.*A2)
      XB2=(E2*C-D3*(1.-COSGAM)*0.5)/(12.*A2)
      DE12=(A1+A2)/(A1/DE1+A2/DE2)
      DELTA=SQRT((YB1-YB2)**2+(XB1-XB2)**2)
      TURBWC=0.05*CTU3*PERSEP/DELTA*DE12
      RETURN
      END
```

```
      FUNCTION TWCTEP(QRMDAR,TW)
C-----------------------------------------------------------------------
C     TWCTEP CORRECTS THE COMPUTED ROD TEMPERATURE TO TAKE INTO ACCOUNT
C     THE POSITION OF THE THERMOCOUPLES INSIDE THE CANNING
C
      COMMON/DATKM/C1(7),C2(7)/INPAR/IPA
      D1=C1(IPA)
      D2=C2(IPA)
      TWCTEP=(-D1+SQRT(D1**2+2.*D2*(D1*TW+0.5*D2*TW**2+QRMDAR)))/D2
      RETURN
      END
```

```
      FUNCTION TWFUN(NRODS,NSTOT,PIG,AAC,AAA)
C-----------------------------------------------------------------------
C     TWFUN EVALUATES AN AVERAGE VALUE FOR ROD TEMPERATURES
C
      COMMON/IND3/NTYP( 42)/HEA6/NPIN( 42),JPIN( 42,3)/SUB1/ASCH( 42,3)
     1       /SUB22/TW( 42,3)
      SANG=0.
      TWTIPA=0.
      DO 10 NS=1,NSTOT
      NP=NPIN(NS)
      ITYP=NTYP(NS)
      DO 10 M=1,NP
      GOTO(1,2,3),ITYP
    1 ANG=PIG/6.*ASCH(NS,M)/AAC
      GOTO 4
    2 ANG=PIG*0.25
      GOTO 4
    3 ANG=PIG/6.*ASCH(NS,1)/AAA
    4 SANG=SANG+ANG
      TWTIPA=TWTIPA+TW(NS,M)*ANG
   10 CONTINUE
      TWFUN=TWTIPA/SANG
      RETURN
      END
```

```
      FUNCTION UA(I,NI,ACHI,IUAV)
C-----------------------------------------------------------------------
C     UA EVALUATES THE AVERAGE CROSS-FLOW VELOCITIES FOR THE CROSS-FLOW
C     EXCHANGE BETWEEN CHANNELS
C
      COMMON/IJ1/NER( 42),NIS( 42,3)/MOB8/DP( 42)/MOB2/U( 42)
     1       /GEO0/ACH(3)/IND3/NTYP( 42)/LAMIN3/F1ATIP( 42),F1DTIP( 42)
     2       /HEA6/NPIN( 42),JPIN( 42,3)/GAMAR/CXX
      UU=0.
      AA=0.
      ACHN=ACHI*F1ATIP(I)
      DO 2 M=1,NI
      J=NIS(I,M)
      NTYPJ=NTYP(J)
      ACHJ=ACH(NTYPJ)*F1ATIP(J)
      CXX=0.5
      IF((NTYP(I)+NPIN(I).EQ.4) .OR. (NTYP(J)+NPIN(J).EQ.4))CXX=1.
      CALL CF1(U(I),U(J),ACHN,ACHJ,DP(I),DP(J),IUAV,UU,AA)
    2 CONTINUE
      CXX=1.
      UA=UU/AA
      RETURN
      END
```

```
      SUBROUTINE VFCAL
C ----------------------------------------------------------------------
C     COMPUTES THE VIEW FACTORS.
C
      COMMON /SC02R/ P,D,Z,ZWC,H,LENGTH
      COMMON /DAT/   PIG
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
C
      R=D*0.5
C
      CALL CFC1 (P,D,PIG,F1)
      CALL CFC2 (P,R,Z,ZWC,PIG,F2)
      CALL CFC3 (P,D,PIG,F3)
      CALL CFC4 (P,D,PIG,F4)
      CALL CFC5 (P,D,PIG,F5)
C
      CALL CF11(P,D,Z,ZWC,PIG,F11)
      CALL CF12(P,D,Z,ZWC,PIG,F12)
      CALL CF13(P,D,Z,ZWC,PIG,F13)
      CALL CF14(P,D,Z,ZWC,PIG,F14)
      CALL CF15(P,D,Z,ZWC,PIG,F15)
      CALL CF16(P,D,Z,ZWC,PIG,F16)
      CALL CF17(P,D,Z,ZWC,PIG,F17)
      CALL CF18(P,D,Z,ZWC,PIG,F18)
      CALL CF24(P,D,Z,ZWC,PIG,F24)
      CALL CF31(P,D,Z,ZWC,PIG,F31)
      CALL CF32(P,D,Z,ZWC,PIG,F32)
C
      F7 = F5
      F8 = F4
C
      G1  =F1 /(PIG*D/3.)
      G3  =F3 /(PIG*D/3.)
      G4  =F4 /(PIG*D/3.)
      G5  =F5 /(PIG*D/3.)
      GP15=F15/(PIG*D/3.)
      GP14=F14/(PIG*D/3.)
      GP24=F24/(PIG*D/3.)
C
      G2  =F2 /(PIG*D/2.)
      G7  =F7 /(PIG*D/2.)
      G8  =F8 /(PIG*D/2.)
      GP11=F11/(PIG*D/2.)
      GP12=F12/(PIG*D/2.)
      GP13=F13/(PIG*D/2.)
      GP16=F16/(PIG*D/2.)
      GP17=F17/(PIG*D/2.)
      GP18=F18/(PIG*D/2.)
C
      GS11=F11/PSTAR(ZWC,P)
      GS12=F12/PSTAR(ZWC,P)
      GS13=F13/PSTAR(ZWC,P)
      GS15=F15/PSTAR(ZWC,P)
      GS16=F16/PSTAR(ZWC,P)
      GS24=F24/PSTAR(ZWC,P)
      GS32=F32/PSTAR(ZWC,P)
      GW31=F31/PSTAR(ZWC,P)
C
      GS14=F14/(4.*Z/SQRT(3.))
      GS17=F17/(4.*Z/SQRT(3.))
```

```
      GS18=F18/(4.*Z/SQRT(3.))
      GA31=F31/(4.*Z/SQRT(3.))
C
      WRITE(6,100) G1,G2,G3,G4,G5,G7,G8
  100 FORMAT ( 5X,' F1   = ',F10.5,/5X,
     >            ' F2   = ',F10.5,/5X,
     >            ' F3   = ',F10.5,/5X,
     >            ' F4   = ',F10.5,/5X,
     >            ' F5   = ',F10.5,/5X,
     >            ' F7   = ',F10.5,/5X,
     >            ' F8   = ',F10.5,//)
C
      WRITE(6,200) GP11,GP12,GP13,GP14,GP15,GP16,GP17,GP18,GP24
  200 FORMAT ( 5X,' FP11 = ',F10.5,/5X,
     >            ' FP12 = ',F10.5,/5X,
     >            ' FP13 = ',F10.5,/5X,
     >            ' FP14 = ',F10.5,/5X,
     >            ' FP15 = ',F10.5,/5X,
     >            ' FP16 = ',F10.5,/5X,
     >            ' FP17 = ',F10.5,/5X,
     >            ' FP18 = ',F10.5,/5X,
     >            ' FP24 = ',F10.5,///)
C
      WRITE(6,300)GS11,GS12,GS13,GS14,GS15,GS16,GS17,GS18,GS24,GA31,
     >            GW31,GS32
  300 FORMAT( /5X,' FS11 = ',F10.5,/5X,
     >            ' FS12 = ',F10.5,/5X,
     >            ' FS13 = ',F10.5,/5X,
     >            ' FS14 = ',F10.5,/5X,
     >            ' FS15 = ',F10.5,/5X,
     >            ' FS16 = ',F10.5,/5X,
     >            ' FS17 = ',F10.5,/5X,
     >            ' FS18 = ',F10.5,/5X,
     >            ' FS24 = ',F10.5,/5X,
     >            ' FA31 = ',F10.5,/5X,
     >            ' FW31 = ',F10.5,/5X,
     >            ' F32  = ',F10.5,///)
C
      RETURN
      END
```

```
      SUBROUTINE VFCTR
C ---------------------------------------------------------------------
C     CONTROLS THE ARRAY OF VIEW-FACTORS.
C
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC13R/ NAFF(132)
      COMMON /SC14R/ KAFF(132, 13)
      COMMON /SC01R/ NSECT,NSECP
C
      DO 1000 L1=1,NSECT
          NAF1=NAFF(L1)
          IF(NAF1.LE.0) GO TO 1000
          DO 500 K1=1,NAF1
              L2=KAFF(L1,K1)
              NAF2=NAFF(L2)
              IF(NAF2.GT. 0) GO TO 200
              WRITE(6,100) L1,K1,L2,NAF2
  100         FORMAT( ' VFCTR',4I4)
  200         CONTINUE
              DO 400 K2=1,NAF2
                  IF(KAFF(L2,K2).NE. L1)  GO TO 400
                  IF(VFAC(L1,K1).EQ.0.0.OR .VFAC(L2,K2).EQ.0.0)GOTO 300
                  P=ABS(VFAC(L1,K1)/VFAC(L2,K2)-1.0)
                  P1=ABS(VFAC(L1,K1)*2./VFAC(L2,K2)-1.0)
                  P2=ABS(VFAC(L1,K1)/VFAC(L2,K2)/2.-1.0)
                  IF(P.LT.0.01) GO TO 400
                  IF(P1.LT.0.01) GO TO 400
                  IF(P2.LT.0.01) GO TO 400
  300             WRITE(6,1100) L1,K1,VFAC(L1,K1),L2,K2,VFAC(L2,K2)
  400             CONTINUE
  500         CONTINUE
 1000      CONTINUE
C
C
 1100 FORMAT ( 5X,'NO CORRISPONDENCE IN VIEW-FACTORS ARRAY:',/5X,
     >            'VIEW-FACTOR ',I3,',',I3,'  =  ',F10.5,/5X,
     >            'VIEW-FACTOR ',I3,',',I3,'  =  ',F10.5,/)
C
      DO 2000 L=1,NSECT
              TOT=0.0
              DA=DAREA(L)
              NAF=NAFF(L)
              IF(NAF.LE.0) GO TO 2000
              DO 1500 K=1,NAF
                     TOT=TOT+VFAC(L,K)/DA
 1500                CONTINUE
              WRITE(6,2200)  L,TOT
              IF (ABS(TOT-1.).GT.0.001) WRITE(6,2300)
              IF (TOT.GT.1.0) WRITE(6,2100) L,TOT
              IF (TOT.LT.1.0) CALL NORMA(TOT,L)
 2000         CONTINUE
C
 2100 FORMAT ( 5X,/5X,'VFCTR, ERROR IN SUM FOR ',I3,'  =  ',F10.5,/)
 2200 FORMAT ( 5X,'VFCTR, THE SUM OF VIEW FACTOR FOR THE SECTOR ',I4,
     >            ' IS ',F10.5)
 2300 FORMAT (1H+,80X, ' IT IS COMPENSED TO 1.0')
C
      RETURN
      END
```

```
      SUBROUTINE VFDET(NSEL,NRODS)
C ---------------------------------------------------------------------
C     ORGANIZES THE DETERMINATION OF THE ARRAYS NAFF, KAFF, VFAC
C
      IF (NRODS.NE.12) GO TO 100
C
C     -----------------------------------------------------------------
C     12-ROD BUNDLE
C
      IF(NSEL.NE.1) GO TO 50
      CALL VFDE1
      GO TO 999
   50 CONTINUE
      CALL VFD3
      GO TO 999
  100 CONTINUE
C
C     -----------------------------------------------------------------
C     HEXAGONAL BUNDLE
C
      GO TO (200,300,300,300),NSEL
  200 CALL VFDE1
      GO TO 999
  300 WRITE(6,350)
  350 FORMAT ( ' THE ACTUAL VERSION OF SAGAPO CANNOT DETERMINE THE',
     >         ' ARRAYS NAFF, KAFF, VFAC FOR THE ACTUAL NRODS AND NSEL',
     >         /5X,' CALCULATION STOPS')
      STOP
C
  999 RETURN
      END
```

```
      SUBROUTINE VFDE1
C ----------------------------------------------------------------------
C     DETERMINES THE ARRAYS NAFF, KAFF, VFAC
C     CASE OF THE WHOLE BUNDLE (BOTH 12-ROD AND HEXAGONAL)
C
      COMMON /SC01R/ NSECT,NSECP
      COMMON /SC04R/ VFAC(132, 13)
      COMMON /SC06R/ ISU(132,2)
      COMMON /SC13R/ NAFF(132)
      COMMON /SC14R/ KAFF(132,13)
      COMMON /HEA6/  NPIN( 42),JPIN( 42,3)
      COMMON /IND3/  NTYP( 42)
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      DO 100 L=1,NSECT
             DO 50 K=1,13
                   VFAC(L,K)=0.0
   50              CONTINUE
  100        CONTINUE
      DO 1000 L=1,NSECT
              NAFF(L)=0
              NS=ISU(L,1)
              M =ISU(L,2)
              J =JPIN(NS,M)
              NTY=NTYP(NS)
              IF (L .GT. NSECP) GO TO 500
              GO TO (110,200,300), NTY
  110         CALL FINDF1(L,NAFF(L),NS,J)
              CALL FINDF3(L,NAFF(L),NS,J)
              CALL FINDF4(L,NAFF(L),NS,J)
              CALL FINDF5(L,NAFF(L),NS,J)
              GO TO 1000
  200         CALL FINDW2(L,NAFF(L),NS,J)
              CALL FW7   (L,NAFF(L),NS,J)
              CALL FW8   (L,NAFF(L),NS,J)
              CALL FW1112(L,NAFF(L),NS,J)
              CALL FW13  (L,NAFF(L),NS,J)
              CALL FW16  (L,NAFF(L),NS,J)
              CALL FW1718(L,NAFF(L),NS,J)
              GO TO 1000
  300         CALL FINDA(L,NAFF(L),NS,J)
              GO TO 1000
  500         CONTINUE
              GO TO (600,700,800), NTY
  600         WRITE(6,610)
  610         FORMAT( ' ERROR IN VCALC, CALCULATION PROCEEDING')
              GO TO 1000
  700         CALL FS1112(L,NAFF(L),NS,J)
              CALL FFS13 (L,NAFF(L),NS,J)
              CALL FFS16 (L,NAFF(L),NS,J)
              CALL FFS24 (L,NAFF(L),NS,J)
              CALL FF32  (L,NAFF(L),NS,J)
              CALL FFW31 (L,NAFF(L),NS,J)
              CALL FS2   (L,NAFF(L),NS,J)
              GO TO 1000
  800         CALL FSA   (L,NAFF(L),NS,J)
              CALL FFA31 (L,NAFF(L),NS,J)
 1000 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VFD3
C -----------------------------------------------------------------------
C     BUILDS THE ARRAY OF VIEW-FACTORS
C     VERSION FOR 1/3 OF THE 12-ROD BUNDLE
C     IN THIS CASE NAFF AND KAFF ARE GIVEN IN BLOCK DATA
C
      COMMON /SC01R/ NSECT,NSECP
      COMMON /SC03R/ F1,F2,F3,F4,F5,F6,F7,F8,F9,
     >               F11,F12,F13,F14,F15,F16,F17,F18,F24,F31,F32
      COMMON /SC04R/ VFAC(132,13)
C
      DO 100 L=1,NSECT
             DO 50 K=1,13
                   VFAC(L,K)=0.0
   50              CONTINUE
  100        CONTINUE
C
      VFAC( 1, 1) = 2.*F1
      VFAC( 1, 2) = F3/2.+F4
      VFAC( 1, 3) = F3/2.+F4
      VFAC( 1, 4) = F4+F5
      VFAC( 1, 5) = F4+F5
C
      VFAC( 2, 1) = F1/2.
      VFAC( 2, 2) = F1/2.
      VFAC( 2, 3) = F1
      VFAC( 2, 4) = F3
      VFAC( 2, 5) = F4
      VFAC( 2, 6) = F4
      VFAC( 2, 7) = F4
      VFAC( 2, 8) = F4+F5
      VFAC( 2, 9) = F5
C
      VFAC( 3, 1) = F1
      VFAC( 3, 2) = F1
      VFAC( 3, 3) = F15
      VFAC( 3, 4) = F15
      VFAC( 3, 5) = F4
      VFAC( 3, 6) = F4
      VFAC( 3, 7) = F4
      VFAC( 3, 8) = F4
      VFAC( 3, 9) = F5/2.
      VFAC( 3,10) = F5/2.
      VFAC( 3,11) = F5
C
      VFAC( 4, 1) = F1
      VFAC( 4, 2) = F1
      VFAC( 4, 3) = F4
      VFAC( 4, 4) = F4
      VFAC( 4, 5) = F4
      VFAC( 4, 6) = F4
      VFAC( 4, 7) = F15
      VFAC( 4, 8) = F15
      VFAC( 4, 9) = F5
      VFAC( 4,10) = F5
C
      VFAC( 5, 1) = F1
      VFAC( 5, 2) = F1
      VFAC( 5, 3) = F15
      VFAC( 5, 4) = F15
      VFAC( 5, 5) = F4
```

```
      VFAC( 5, 6) = F4
      VFAC( 5, 7) = F4
      VFAC( 5, 8) = F4
      VFAC( 5, 9) = F5
      VFAC( 5,10) = F5/2.
      VFAC( 5,11) = F5/2.
C
      VFAC( 6, 1) = F1
      VFAC( 6, 2) = F1/2.
      VFAC( 6, 3) = F1/2.
      VFAC( 6, 4) = F3
      VFAC( 6, 5) = F4
      VFAC( 6, 6) = F4
      VFAC( 6, 7) = F4+F5
      VFAC( 6, 8) = F4
      VFAC( 6, 9) = F5
C
      VFAC( 7, 1) = F1
      VFAC( 7, 2) = F1
      VFAC( 7, 3) = F3+2.*F4
      VFAC( 7, 4) = F4
      VFAC( 7, 5) = F4
      VFAC( 7, 6) = F5
      VFAC( 7, 7) = F5
C
      VFAC( 8, 1) = F1
      VFAC( 8, 2) = F1
      VFAC( 8, 3) = F3
      VFAC( 8, 4) = F4
      VFAC( 8, 5) = F4
      VFAC( 8, 6) = F4
      VFAC( 8, 7) = F5
      VFAC( 8, 8) = F5
C
      VFAC( 9, 1) = F2
      VFAC( 9, 2) = F11
      VFAC( 9, 3) = F12
      VFAC( 9, 4) = F16
      VFAC( 9, 5) = F8
      VFAC( 9, 6) = F7
      VFAC( 9, 7) = F18
C
      VFAC(10, 1) = F1
      VFAC(10, 2) = F1
      VFAC(10, 3) = F3
      VFAC(10, 4) = F4/2.
      VFAC(10, 5) = F4/2.
      VFAC(10, 6) = F4
      VFAC(10, 7) = F4
      VFAC(10, 8) = F5
      VFAC(10, 9) = F5
C
      VFAC(11, 1) = F1
      VFAC(11, 2) = F1
      VFAC(11, 3) = F3
      VFAC(11, 4) = F4
      VFAC(11, 5) = F4
      VFAC(11, 6) = F4
      VFAC(11, 7) = F5
      VFAC(11, 8) = F5
C
      VFAC(12, 1) = F2
      VFAC(12, 2) = F11
      VFAC(12, 3) = F12
      VFAC(12, 4) = F17
      VFAC(12, 5) = F18
```

```
      VFAC(12, 6) = F7
      VFAC(12, 7) = F8
C
      VFAC(13, 1) = F14
      VFAC(13, 2) = F24
      VFAC(13, 3) = F24
C
      VFAC(14, 1) = F2
      VFAC(14, 2) = F11
      VFAC(14, 3) = F12
      VFAC(14, 4) = F13
      VFAC(14, 5) = F17
      VFAC(14, 6) = F7
      VFAC(14, 7) = F8
C
      VFAC(15, 1) = F1
      VFAC(15, 2) = F1
      VFAC(15, 3) = F4
      VFAC(15, 4) = F4
      VFAC(15, 5) = F4
      VFAC(15, 6) = F3
      VFAC(15, 7) = F5
      VFAC(15, 8) = F5
C
      VFAC(16, 1) = F1
      VFAC(16, 2) = F1
      VFAC(16, 3) = F3
      VFAC(16, 4) = F4
      VFAC(16, 5) = F4
      VFAC(16, 6) = F4/2.
      VFAC(16, 7) = F4/2.
      VFAC(16, 8) = F5
      VFAC(16, 9) = F5
C
      VFAC(17, 1) = F2
      VFAC(17, 2) = F11
      VFAC(17, 3) = F12
      VFAC(17, 4) = F17
      VFAC(17, 5) = F13
      VFAC(17, 6) = F7
C
      VFAC(18, 1) = F14
      VFAC(18, 2) = F24
      VFAC(18, 3) = F24
C
      VFAC(19, 1) = F2
      VFAC(19, 2) = F11
      VFAC(19, 3) = F12
      VFAC(19, 4) = F18
      VFAC(19, 5) = F17
      VFAC(19, 6) = F7
C
      VFAC(20, 1) = F1
      VFAC(20, 2) = F1
      VFAC(20, 3) = F3+F4+F4
      VFAC(20, 4) = F4
      VFAC(20, 5) = F4
      VFAC(20, 6) = F5
      VFAC(20, 7) = F5
C
      VFAC(21, 1) = F1
      VFAC(21, 2) = F1
      VFAC(21, 3) = F3
      VFAC(21, 4) = F4
      VFAC(21, 5) = F4
      VFAC(21, 6) = F4
```

```
      VFAC(21, 7) = F5
      VFAC(21, 8) = F5
C
      VFAC(22, 1) = F2
      VFAC(22, 2) = F11
      VFAC(22, 3) = F12
      VFAC(22, 4) = F18
      VFAC(22, 5) = F16
      VFAC(22, 6) = F7
      VFAC(22, 7) = F8
C
      VFAC(23, 1) = F11
      VFAC(23, 2) = F12
      VFAC(23, 3) = F15
      VFAC(23, 4) = F32
      VFAC(23, 5) = F13
      VFAC(23, 6) = F16
C
      VFAC(24, 1) = F11
      VFAC(24, 2) = F12
      VFAC(24, 3) = F31
      VFAC(24, 4) = F24
      VFAC(24, 5) = F15
C
      VFAC(25, 1) = F14
      VFAC(25, 2) = F17
      VFAC(25, 3) = F17
      VFAC(25, 4) = F18
      VFAC(25, 5) = F18
      VFAC(25, 6) = F31
      VFAC(25, 7) = F31
C
      VFAC(26, 1) = F11
      VFAC(26, 2) = F12
      VFAC(26, 3) = F31
      VFAC(26, 4) = F24
      VFAC(26, 5) = F15
C
      VFAC(27, 1) = F11
      VFAC(27, 2) = F12
      VFAC(27, 3) = F31
      VFAC(27, 4) = F24
      VFAC(27, 5) = F15
C
      VFAC(28, 1) = F14
      VFAC(28, 2) = F17
      VFAC(28, 3) = F17
      VFAC(28, 4) = F31
      VFAC(28, 5) = F31
      VFAC(28, 6) = F18
      VFAC(28, 7) = F18
C
      VFAC(29, 1) = F11
      VFAC(29, 2) = F12
      VFAC(29, 3) = F15
      VFAC(29, 4) = F31
      VFAC(29, 5) = F24
C
      VFAC(30, 1) = F11
      VFAC(30, 2) = F12
      VFAC(30, 3) = F15
      VFAC(30, 4) = F32
      VFAC(30, 5) = F13
      VFAC(30, 6) = F16
C
      RETURN
      END
```

```
      SUBROUTINE WALLTE(K,NSTOT,NSTR,RH,SUR,D,PIG,TE,PBT,*,RTI,H)
C-----------------------------------------------------------------------
C     WALLTE ORGANIZES THE CALCULATION OF THE PIN AND OF THE SHROUD
C     TEMPERATURES
C
      REAL LAMSCH,LAMWC,LAMB,MSCH,MAWC,LENGTH
      COMMON/HEA5/QQ( 42,3)/HEA6/NPIN( 42),JPIN( 42,3)/GEN5/DE( 42)
     0      /LAMIN4/F2ATIP( 42,3),F2DTIP( 42,3)
     1      /SUB1/ASCH( 42,3)/SUB2/TSCH( 42,3),MSCH( 42,3)
     2      /SUB3/ADAB( 18,2),DETB( 18,2)/SUB4/LAMB( 18,2)
     3      /SUB5/LAMSCH( 42,3)/SUB22/TW( 42,3)
     4      /WCSE1/DEWC( 18,2,2),PHWC( 18,2,2)/WCSE3/LAMWC( 18,2,2)
     5      /WCSE7/MAWC( 18,2,2)/WCSE8/ASCHWC( 18,2,2)
     6      /WCSE12/TWWC( 18,2,2)/IND3/NTYP( 42)/QPAR3/PERL(3)
     A      /GEO0/ACH(3)
     7      /WACO1/XMSCHB( 18,2),XMSCHA( 18,2)/SHROUD/TLINER( 18,2)
     8      /PARTB/TEMPB( 42,3),XMASSB( 42,3),YDH( 42,3)/EXAVTW/IEXAV
     9      /SUB21/TSCHA( 18,2),TSCHB( 18,2)/QPAR1/QDEV/IROSMO/IRH
      COMMON /WCSE9/TAVWC( 18,2,2)
      COMMON /SC02C/ QJ( 19, 42)
      COMMON /SC07C/ H1
      COMMON /SC13C/ GEO1( 42,3)
      COMMON /SC10C/ ANU( 42,3)
      COMMON /SC15C/ ALFA( 42,3)
      COMMON /SC01Z/ YH( 42,3)
      COMMON /INPAR/ IPA
      COMMON /LAMIN5/ RTIP(7)
      COMMON /CEV03/ LAMOP3
      COMMON /SC05L/ PERLT
      COMMON /QSHR / QA11
      COMMON /SC03L/ IPHUD
      COMMON /SC04L/ TAMB,HUDFAC,TMS
      COMMON /SC02R/ DUM1,DUM2,DUM3,DUM4,DUM5,LENGTH
      COMMON /SC08R/ QPR( 42,3)
      COMMON /SC09R/ QSR( 18,2)
      COMMON /SC06L/ SHQ( 18,2)
      COMMON /SC21C/ SHQC( 18,2)
C
C
C     ..................................................................
C     I=CHANNEL INDEX
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979
      TMS=0.0
      PMS=0.0
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979
      DO 11 I=1,NSTOT
      NP=NPIN(I)
      ITYP=NTYP(I)
      II=I-NSTR
C     ..................................................................
C     M=SUBCHANNEL INDEX
C
      DO 9 M=1,NP
      PERS=PERL(ITYP)*ASCH(I,M)/ACH(ITYP)
      TW(I,M)=TSCH(I,M)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     QA=QQ(I,M)/SUR*QDEV                        OLD
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                NEW
      IPIN=JPIN(I,M)
```

```
      NTOP=NTYP(I)
      QA=QQ(I,M)/SUR*QDEV + QJ(IPIN,I)/(PIG*D*H/GEO1(I,M))
      IF (I.EQ.1.AND.IPIN.EQ.1)
     >QA=QQ(1,1)/SUR*QDEV + QJ(1,1)/(PIG*D*H/6.0)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      GOTO(1,2,7),ITYP
C     ................................................................
C--A) CENTRAL SUBCHANNELS
C
    1 CALL RTI (PBT,TSCH(I,M),MSCH(I,M),DE(I),ASCH(I,M),1.,LAMSCH(I,M),
     1         QA,FACHE,TE,RH,I,II,M,JPIN(I,M),TW(I,M),1.,1,
     2   DE(I),D,YDH(I,M),&8500,F2ATIP(I,M),F2DTIP(I,M),QALIN)
      TEMPB(I,M)=TSCH(I,M)
      XMASSB(I,M)=MSCH(I,M)
      GOTO 9
C     ................................................................
C--B) WALL SUBCHANNELS
C
    2 TW(I,M)=0.
      DO 5 JWC=1,2
       TWWC(II,M,JWC)=TSCH(I,M)
      GOTO(3,4),JWC
C
C  -1-WALL TYPE PART
C
    3 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979
      TS=TAVWC(II,M,1)
      DO 1000 ITHUD=1,10
      IF(IPHUD. EQ. 1) SHQ(II,M)=HUDFAC*PERS/PERLT*(TS-TAMB)*H
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      QALIN=(SHQ(II,M)+SHQC(II,M))/(PERS*H)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      RU1DRU=XMSCHB(II,M)*ADAB(II,M)/MAWC(II,M,1)
      CALL RTI (PBT,TAVWC(II,M,1),MAWC(II,M,1),DETB(II,M),ASCHWC(II,M,1)
     1         ,ADAB(II,M),LAMB(II,M),QA,FACHE,TE,RH,I,II,M,
     2         JPIN(I,M),TWWC(II,M,1),RU1DRU,2,DEWC(II,M,1),D,XXXX,&8500
     3         ,1.,1.,QALIN)
C
      IF (IPHUD. EQ. 0) GO TO 1200
      DELT=ABS(TLINER(II,M)/TS-1.0)
      IF( DELT .LT. 0.01) GO TO 1200
      TS=TLINER(II,M)
 1000 CONTINUE
      WRITE(6,1100) DELT
 1100 FORMAT( 5X,'WALLTE, WALL: NO CONVERGENCE IN ITHUD, DELT=',F10.5,
     >       /5X,'CALCULATION STOPS.')
      STOP
 1200 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979 AND 14.11.1980
      ALFA1=ALFA(I,M)
      ANU1 = ANU(I,M)
CCCCCCCCCCCCCCCC 02.11.1979
      TMS=TMS+TLINER(II,M)*PERL(2)*0.5
      PMS=PMS+PERL(2)*0.5
CCCCCCCCCCCCCCCC
      GOTO 5
C
C  -2-CENTRAL TYPE PART
C
    4 CALL RTI (PBT,TAVWC(II,M,2),MAWC(II,M,2),DEWC(II,M,2),ASCHWC(II,M,
     1         2),1.,LAMWC(II,M,2),QA,FACHE,TE,RH,I,II,M,JPIN
     2         (I,M),TWWC(II,M,2),1.,1,DEWC(II,M,2),D,XXXX,&8500,1.,1.,
     3         0.0)
```

```
C
CCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979 AND 14.11.1980
      ALFA2=ALFA(I,M)
      ANU2 = ANU(I,M)
CCCCCCCCCCCCCCCCC
    5 TW(I,M)=TW(I,M)+PHWC(II,M,JWC)*TWWC(II,M,JWC)
    6 CONTINUE
      TW(I,M)=TW(I,M)*4./(D*PIG)
CCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979 AND 14.11.1980
      ALFA(I,M)= (ALFA1*PHWC(II,M,1) + ALFA2*PHWC(II,M,2))/(D*PIG/4.)
      ANU (I,M)= ( ANU1*PHWC(II,M,1) +  ANU2*PHWC(II,M,2))/(D*PIG/4.)
      ANU (I,M)= ANU(I,M)/YH(I,M)
      IF (QQ(I,M) .GT. 1.E-06) ALFA(I,M)=QA/(TW(I,M)-TSCH(I,M))
CCCCCCCCCCCCCCCCC
      XMASSB(I,M)=XMSCHB(II,M)+MAWC(II,M,2)
      TEMPB(I,M)=(XMSCHB(II,M)*TSCHB(II,M)+MAWC(II,M,2)*TAVWC(II,M,2))/
     /XMASSB(I,M)
      IF (IRH.EQ.2)
     *YDH(I,M)=0.5*(SQRT(D**2+16./PIG*ASCH(I,M))-D)/RH
      GOTO 9
C     ...............................................................
C--C) CORNER CHANNELS
C
    7 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979
      TS=TSCH(I,1)
      DO 2000 ITHUD=1,10
      IF(IPHUD. EQ. 1) SHQ(II,M)=HUDFAC*PERS/PERLT*(TS-TAMB)*H
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      QALIN=(SHQ(II,M)+SHQC(II,M))/(PERS*H)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     QALI1=QA11
C     IF(IPHUD.EQ.1) QALI1=HUDFAC/PERLT*(TS-TAMB)
C     QALIN=QALI1+QSR(II,M)/LENGTH
C 700 FORMAT( 5X,' WALLTE, CHANNEL =',I3,' SUB. =',I3,' QALI1 =',E12.6,
C    >        2X,' QSR=',E12.6,' QALIN =',E12.6)
C
      RU1DRU=XMSCHB(II,1)*ADAB(II,1)/MSCH(I,1)
      CALL RTI (PBT,TSCH(I,1),MSCH(I,1),DETB(II,1),ASCH(I,1),ADAB(II,1),
     1        LAMB(II,1),QA,FACHE,TE,RH,I,II,1,JPIN(I,1),
     2         TW(I,1),RU1DRU,3,DE(I),D,YDH(I,M),&8500,F2ATIP(I,1),
     3        F2DTIP(I,1),QALIN)
C
      IF (IPHUD. EQ. 0) GO TO 2200
      DELT=ABS(TLINER(II,1)/TS-1.0)
      IF( DELT .LT. 0.01) GO TO 2200
      TS=TLINER(II,1)
 2000 CONTINUE
      WRITE(6,2100) DELT
 2100 FORMAT( 5X,'WALLTE, COR.: NO CONVERGENCE IN ITHUD, DELT=',F10.5,
     >       /5X,'CALCULATION STOPS.')
      STOP
 2200 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCC 02.11.1979
CCCCCCCCCCCCCCCCC 02.11.1979
      TMS=TMS+TLINER(II,M)*PERS
      PMS=PMS+PERS
CCCCCCCCCCCCCCCCC
      TEMPB(I,1)=TSCHB(II,1)
      XMASSB(I,1)=XMSCHB(II,1)
    9 CONTINUE
   11 CONTINUE
CCCCCCCCCCCCCCCCC 02.11.1979
      TMS=TMS/PMS
CCCCCCCCCCCCCCCCC
```

```
C      ..................................................................
C     IF AN AVERAGE VALUE IS DESIRED FOR THE PIN AND THE SHROUD
C     TEMPERATURES OF THE EXTERNAL CHANNELS
C
      IF(IEXAV.EQ.1)RETURN
      PERLT=0.
      PERRT=0.
      TLM=0.
      TWM=0.
      NSTR1=NSTR+1
      DO 20 I=NSTR1,NSTOT
      ITYP=NTYP(I)
      NP=NPIN(I)
      DO 20 M=1,NP
      PERLSC=PERL(ITYP)*ASCH(I,M)/ACH(ITYP)
      PERLT=PERLT+PERLSC
      PERRSC=1./NTYP(I)
      PERRT=PERRT+PERRSC
      TLM=TLM+TLINER(I-NSTR,M)*PERLSC
   20 TWM=TWM+TW(I,M)*PERRSC
      TLM=TLM/PERLT
      TWM=TWM/PERRT
      DO 30 I=NSTR1,NSTOT
      NP=NPIN(I)
      DO 30 M=1,NP
      TLINER(I-NSTR,M)=TLM
   30 TW(I,M)=TWM
      RETURN
 8500 RETURN 1
      END
```