TACT1, A Computer Program for the Transient Thermal Analysis of a Cooled Turbine Blade or Vane Equipped With a Coolant Insert

II - Programmers Manual

Raymond E. Gaugler

JANUARY 1979
TACT1, A Computer Program for the Transient Thermal Analysis of a Cooled Turbine Blade or Vane Equipped With a Coolant Insert

II - Programmers Manual

Raymond E. Gaugler
Lewis Research Center
Cleveland, Ohio

NASA
National Aeronautics
and Space Administration
Scientific and Technical Information Office
1979
# CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUMMARY</td>
<td>1</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>OVERVIEW</td>
<td>2</td>
</tr>
<tr>
<td>Method of Analysis</td>
<td>2</td>
</tr>
<tr>
<td>Blade geometric model</td>
<td>2</td>
</tr>
<tr>
<td>Numerical model</td>
<td>3</td>
</tr>
<tr>
<td>General Program Description</td>
<td>3</td>
</tr>
<tr>
<td>DETAILED PROGRAM PROCEDURE</td>
<td>5</td>
</tr>
<tr>
<td>Block Data NGASDAT</td>
<td>6</td>
</tr>
<tr>
<td>Main Program NTTACT</td>
<td>6</td>
</tr>
<tr>
<td>Subroutine FLOWS</td>
<td>7</td>
</tr>
<tr>
<td>Subroutine FLSPLT</td>
<td>7</td>
</tr>
<tr>
<td>Subroutine GASTBL</td>
<td>8</td>
</tr>
<tr>
<td>Subroutine GAUSS</td>
<td>8</td>
</tr>
<tr>
<td>Subroutine GETIN</td>
<td>8</td>
</tr>
<tr>
<td>Function HCFRCD</td>
<td>9</td>
</tr>
<tr>
<td>Subroutine HCOOL</td>
<td>9</td>
</tr>
<tr>
<td>Subroutine HCPINS</td>
<td>9</td>
</tr>
<tr>
<td>Subroutine INPRT</td>
<td>9</td>
</tr>
<tr>
<td>Subroutine PARRAY</td>
<td>9</td>
</tr>
<tr>
<td>Subroutine PLNUM</td>
<td>10</td>
</tr>
<tr>
<td>Subroutine PLOTMF</td>
<td>10</td>
</tr>
<tr>
<td>Subroutine PREP</td>
<td>10</td>
</tr>
<tr>
<td>Subroutine TARRAY</td>
<td>10</td>
</tr>
<tr>
<td>Subroutine TCOEF</td>
<td>11</td>
</tr>
<tr>
<td>Subroutine THRCON</td>
<td>11</td>
</tr>
<tr>
<td>Subroutine WROUT</td>
<td>11</td>
</tr>
<tr>
<td>DICTIONARY</td>
<td>11</td>
</tr>
<tr>
<td>PROGRAM LISTING</td>
<td>53</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>150</td>
</tr>
</tbody>
</table>
A FORTRAN IV computer program to calculate transient and steady-state temperatures, pressures, and flows in a cooled turbine blade or vane with an impingement insert has been developed and is described in this report. Coolant-side heat-transfer coefficients are calculated internally in the program, with the user specifying one of three modes of heat transfer at each station: (1) impingement, including the effect of crossflow; (2) forced-convection channel flow; or (3) forced convection over pin fins. Additionally, a limited capability to handle film cooling is available in the program. It is assumed that spent impingement air flows in a chordwise direction and is discharged through a split or drilled trailing edge and through film-cooling holes. The program does not allow for radial flow of the spent impingement air. The use of film cooling is restricted by a numerical model requirement for a continuous coolant-channel flow.

Input to the program includes a description of the blade geometry, coolant-supply conditions, outside thermal boundary conditions, and wheel speed. The user can divide the blade by chordwise cuts into as many as 15 slices and can divide each slice into as many as 79 stations around the blade. Each station in turn consists of four calculational nodes through the wall and one in the coolant channel. The blade wall can be two layers of different materials, such as a ceramic thermal-barrier coating over a metallic substrate. Program output includes the temperature at each node, the coolant pressures and flow rates, and the coolant-side heat-transfer coefficients.

INTRODUCTION

As core turbine-engine operating conditions become more severe, it becomes more difficult to effectively cool blades and vanes. Advanced transient thermal calculational techniques are needed to design reliable turbine blades. However, there appears to be no computer program generally available that uses these advanced techniques in combining the required heat-transfer and coolant-flow-distribution calculations. Thus, it was decided to create a computer program that would perform both transient and steady-state heat-transfer and coolant-flow analyses for a cooled blade, given the outside hot-gas boundary conditions, the coolant inlet pressure or flow rate, the geometry of the blade shell, and the cooling configuration.

The resulting program, TACT1, can handle a turbine blade or vane that is equipped with a central coolant-plenum insert from which coolant air flows through holes to impinge on the inner surface of the blade shell or directly into the trailing-edge region. It
is assumed that the spent impingement air then flows chordwise and is dumped through a split or drilled trailing edge and through film-cooling holes. The blade is modeled by dividing it by chordwise planes into as many as 15 slices, with each slice having as many as 79 calculational stations around the blade. Temperatures at each station are calculated for four points through the wall and one in the coolant channel. Included in this model is the capability to analyze a blade with a ceramic thermal-barrier coating. The ability of the program to model film cooling is limited by the numerical flow analysis requirement for a continuous coolant-channel flow.

The TACT1 program is used at the NASA Lewis Research Center on an IBM TSS/360-67 computer. The source program consists of approximately 6000 lines of code and the program requires about 60 000 words of storage. Typical running times for the program are 1.4 seconds of central processor unit (CPU) time per calculational station for a steady-state run and 0.4 second of CPU time per station per time step for a transient run.

The TACT1 program is reported in two parts. This report, part II, is a programmers manual and includes a complete program listing and a detailed description of the procedure. Part I (ref. 1) is a users manual and contains all the information necessary to run the program: a detailed description of the input, the method of solution, and the output as well as a sample problem.

OVERVIEW

Method of Analysis

The details of the analytical method are presented in part I (ref. 1). The blade model used in the analysis is described briefly in this section.

Blade geometric model. - The key to creating a usable computer program is to have as simple a geometric model as possible for the system being analyzed. In this program, the emphasis is on a blade or vane with a central coolant plenum and chordwise flow of the coolant after impingement. Therefore, it was decided that the primary calculational direction would also be chordwise. The blade is divided into layers that are bounded by chordwise cuts through the blade, as shown in figure 1. Each slice is treated separately in the program, with radial heat conduction in the wall the only communication between layers.

Figure 2 gives the details of the geometric model for a single blade slice and shows the breakdown of the blade or vane into calculational stations and nodes. Each calculational station consists of five nodes: one at the wall outer surface, one at the interface between the coating and blade metal, one at a point midway through the wall metal, one at the wall inner surface, and one in the middle of the coolant channel.
For input to the program, the following basic elements of the geometry are needed for each station: (1) the thicknesses of the wall coating and wall metal and the coolant-channel width, (2) the distance of each node from the adjacent lower-numbered node, and (3) the radial span for this slice. In addition, depending on the mode of heat transfer specified, the user must supply impingement-hole diameter and spacing or pin-fin diameter and spacing. Thermal properties of the blade materials must also be specified. The input is described in detail in reference 1.

Numerical model. - The numerical solution for the temperatures throughout the blade involves writing a transient energy-balance equation for each node and forming a set of equations to be solved for the temperature distribution. Similarly, the coolant pressure distribution is determined by writing the transient momentum equation for flow between adjacent fluid nodes and solving the resulting set of equations for static pressures.

The nodal energy balances are linearized, one-dimensional heat-conduction equations at the wall outer-surface node, at the coating-metal interface, and at the wall inner-surface node. At the midmetal node, a linearized, three-dimensional, heat-conduction equation is used. In the coolant channel, energy and momentum equations for one-dimensional compressible flow including friction and heat transfer are written for the elemental channel length between two coolant nodes. The equations used are presented in reference 1.

General Program Description

The TACT1 program is capable of performing a transient analysis as well as a steady-state analysis. In the case of a transient, the program first performs steady-state calculations to determine the initial conditions for the transient.

Figure 3 shows a schematic of the TACT1 calculational scheme. There are three basic, nested calculational loops that must converge for a steady-state solution to be reached. These loops are labeled A, B, and C in figure 3. The program begins a steady-state analysis with the coolant-supply pressure and total coolant flow fixed. The impingement flow is initially assumed to split uniformly at the leading-edge stagnation station, station 1. All coolant flows for the slice under consideration are calculated first, based on the current pressure distribution. The temperatures at each node are then calculated by solving simultaneously the energy equations presented in reference 1. The pressures at each coolant node are calculated by solving simultaneously the momentum equations presented in reference 1. This cycle, loop A in figure 3, is repeated until the pressure distribution no longer changes. The flow split between suction- and pressure-side coolant channels is then checked by comparing the pressures at the ends of the two channels. If they do not match, the impingement flow split at the leading edge...
is adjusted and the inner loop calculations are repeated. This adjustment comprises loop B in figure 3. Once the proper flow split is achieved, the program moves up the blade to the next slice and repeats this sequence. After all N slices have converged, the total coolant mass flow used is compared with the inlet coolant mass flow. If there is an imbalance, either the inlet flow or the supply pressure is adjusted, depending on which was specified in the input; and the calculations start over. This is loop C in figure 3. Once the overall coolant mass flow balance is satisfied, the steady-state solution is complete and the transient calculations begin. During a transient calculation, loop B is bypassed because the coolant flow-split is primarily a function of blade geometry. Loop C is also bypassed because the inlet coolant mass flow rate at a given time is estimated based on the coolant mass flow used at the previous time step and the change in supply pressure.

The TACT1 subprograms and the calling relations are shown in figure 4. Block data subprogram NGASDAT contains air properties, for use in TACT1, tabulated as functions of temperature at a pressure of 20 atmospheres from reference 2. This subprogram must be loaded before execution of the program. The main program, NTTACT, calls other subroutines in their proper order.

The first call from NTTACT is to GETIN, a subroutine that controls the reading, storing, and printing of input data. Subroutine GETIN calls INPUT to print the input data if the user specifies INEDIT > 0. Subroutine INPRT has a call to PREP to put the input data in its proper form for use. All data are input by using a NAMELIST format.

After the input data have been read, the number of time steps, NTYM, to be used in the transient is determined in NTTACT. If only a steady-state solution is to be calculated, NTYM = 1. Time-dependent boundary conditions are then evaluated, with the initial entries assumed to be steady-state values. Then NTTACT loops through the blade, calling on subroutines PLNUM, PREP, and TCOEF for each slice. The first time through is a steady-state calculation.

Subroutine PLNUM calculates the pressure distribution in the impingement plenum for the current slice, given the inlet pressure and coolant flow-rate. PLNUM calls GASTBL for gas properties.

Subroutine PREP extracts the input data for the current slice from the input arrays.

Subroutine TCOEF controls loop A in figure 3, the iterative calculations of temperature and pressure for the nodes of the current slice. Each iteration in TCOEF requires calls to subroutines FLOWS, HCOOL, THRCON, TARRAY, PARRAY, and GAUSS.

Subroutine FLOWS computes the impingement jet flow rates, coolant-channel mass flow rates, and channel Mach numbers for each station around the blade, given the plenum pressure and temperature and the current pressure distribution in the coolant channel. FLOWS calls GASTBL for gas properties.
Subroutine HCOOL is called to calculate coolant heat-transfer coefficients for all the stations of this slice, based on the latest values of mass flow rate. HCOOL calls function HCFRCD to calculate forced-convection heat-transfer coefficients and GASTBL for gas properties.

Subroutine THRCORX determines the wall thermal conductivity from the input table of conductivity as a function of temperature.

Subroutine TARRAY sets up the array of coefficients for the conduction and convection equations for each node. Calls are made to HCPINS for pin-fin heat-transfer coefficients, to HCFRCD for forced-convection heat-transfer coefficients, and to GASTBL for gas properties. TCOEF calls subroutine GAUSS to solve the set of equations for the temperature at each node.

Subroutine PARRAY sets up the array of coefficients for the momentum equations in the coolant channels and TCOEF calls subroutine GAUSS to solve the set of equations for the pressure at each coolant node.

After a new set of temperatures and pressures has been determined, convergence is checked by using the coolant-channel pressure at the blade leading edge. If this pressure stays within a tolerance band for four successive iterations, convergence is accepted. Once convergence is achieved, TCOEF calls subroutine FLSPLT to check the coolant flow-split between the pressure and suction sides. This is loop B in figure 3. Initially, the impingement jet flow at the forward stagnation station is assumed to split evenly between the suction- and pressure-side channels. If the coolant-channel pressures at the end of the impingement insert do not match, the flow split at the forward stagnation station is adjusted to increase the flow to the channel with the higher pressure at the end of the insert, and iteration loop A is repeated. Once a satisfactory flow split has been achieved, TCOEF calls subroutine WROUT to print the output for this slice and calls subroutine PLOTMF if there is to be graphical output. After NTACT has calculated all blade slices, the total coolant mass flow is compared with the impingement-plenum inlet mass flow rate used to start the calculations. If the two flow rates are not close enough, the inlet mass flow or supply pressure is adjusted and the calculations are repeated. This is loop C in figure 3.

When the initial steady-state solution has been completed, the transient calculations are started. The transient is continued until the time reaches the specified maximum.

Subroutine PLOTMF makes use of a TSS/360 graphics package at the NASA Lewis Research Center to plot temperature and pressure distributions for the blade.

**DETAILED PROGRAM PROCEDURE**

Table I lists the names of each of the subprograms in TACT1, the corresponding
TSS/360 source module names, the COMMON blocks used in each, the names of the subroutines called by each, and the names of subroutines calling each. Table II is a cross-reference listing of named COMMON blocks and the subprograms using them. This section gives a detailed description of each subprogram used in TACT1. All variable names used are defined in the section DICTIONARY. The BLOCK DATA subprogram and the MAIN PROGRAM are discussed first and then each subprogram is described, in alphabetical order.

Block Data NGASDAT

A BLOCK DATA subprogram, NGASDAT, is used to provide a table of gas properties to the program. The properties are put in the array GS through a DATA statement with 5xNG entries, where NG is the number of table entries for each property. The first NG values are temperatures, the second are thermal conductivities, the third are specific heats, the fourth are Prandtl numbers, and the final NG values are viscosities. The property values included are taken from reference 2 at a pressure of 20 atmospheres.

Main Program NTTACT

The MAIN PROGRAM for TACT1, NTTACT, has overall control of the program. Figure 5 is a flow chart for NTTACT. During initialization, a call is made to a system subroutine, TIME, to get a unique label to be used to identify the plotted output for a given run. After the call to GETIN, where all the input data are read, NTTACT initiates the solution procedure by searching the transient boundary condition tables and using linear interpolation to extract the values for the current time. The next step is to begin the loop, labeled C in figure 3. The solution progresses from hub to tip. For each slice, NTTACT calls PLNUM to calculate coolant-supply conditions; PREP to extract the input data from the input tables; and TCOEF to calculate flows, temperatures, and pressures. After the return from TCOEF, NTTACT updates the total amount of coolant used, WUSED, by adding the amount used in the current slice, WIM. The amount of coolant-plenum flow available for the remaining slices, WPLEN, is updated by subtracting WIM. After all slices have been done, the overall amount of coolant used is printed and then checked against the assumed coolant flow-rate. If the absolute value of the difference, EXCESW, is more than 1 percent of the assumed flow, the assumed flow or the supply pressure is adjusted and the calculations are repeated. For transient runs, after the initial steady-state coolant-flow balance, there are no more iterations on coolant flow. Instead the flow for a given time step is based on the actual flow used in
the preceding time step and on the ratio of supply pressure for the two steps. Finally, once all loops have been completed, NTTACT calls PLOTMF to get a final summary plot of blade temperatures.

Subroutine FLOWS

Subroutine FLOWS is a routine to calculate the flow rates through all impingement and film-cooling holes, the friction factor in the coolant channels, and film-cooling effectiveness. FLOWS makes use of the current impingement-plenum mean pressure and temperature and coolant-channel pressure and temperature distributions. The impingement jet flow-rate, WJ, is calculated for each station in the forward region and checked against the choked flow-rate, WCR. If WJ is greater than WCR, then WJ is set equal to WCR. If there is any film cooling on the blade, the film-cooling flow rates in the forward region, WFC, are also calculated. Then, the coolant-channel flow rates, WCROS, are computed by considering a mass balance between stations, as illustrated in figure 6. Once the forward-region coolant flows have been determined, the Reynolds numbers - RE for the coolant channel, and REFC for the film-cooling flow - and the square of the coolant Mach number, AM2, are computed for each forward station.

The next step is to calculate the amount of coolant, WDUMP, dumped directly into the trailing-edge region from the coolant plenum. Then the total amount of coolant used for this slice, WIM, is determined by summing the impingement jet flows and WDUMP. Following this, the flows in the trailing-edge region are computed, with the coolant flow being reduced by the amount of any film-cooling flow. Then, trailing-edge-region values of RE, REFC, and AM2 are calculated.

After all the coolant flow-rates are determined, the friction factor, FF, is calculated at each station. Finally, if there is any film cooling used, the film effectiveness is calculated by using the method of reference 3.

Subroutine FLSPLT

Subroutine FLSPLT is used to determine the location of the stagnation impingement jet, station JS, and the fraction of that jet's flow that splits to each side of the blade, DELTAN. Figure 7 is a detailed flow chart for subroutine FLSPLT. The primary variable carried into FLSPLT is the pressure-match parameter, EPSN, which is defined as

\[
EPSN = \frac{(P(2, ISLICE, NFWD-1) - P(2, ISLICE, NFWD))}{P(2, ISLICE, NFWD-1)} \tag{1}
\]

where the pressures are as illustrated in figure 8.
The magnitude and sign of EPSN are used to determine the adjustment of the stagnation impingement-jet row location and the fraction of that jet that splits to the suction-side channel. Initially, the stagnation jet row is located at station 1 and the split is \( \Delta T = 0.50 \). If EPSN is positive, \( \Delta T \) is set to 0.75 to increase the flow down the suction-side channel; if EPSN is negative, \( \Delta T \) is set to 0.25 to increase the flow down the pressure-side channel. For subsequent entries into FLSPLT, the value of \( \Delta T \) is adjusted by passing a straight line through the last two points on a plot of EPSN versus \( \Delta T \) and picking the value of \( \Delta T \) where this line crosses the axis at \( EPSN = 0 \). If this intercept falls outside the \( \Delta T \) range of 0 to 1, the stagnation station, JS, must be moved to an adjacent station and \( \Delta T \) set to 0.50. Once a sign change is observed in EPSN, a fine-tuning process is triggered in FLSPLT. In this case, the values of \( \Delta T \) and EPSN for the iteration preceding the sign change are saved and used as one of the points of the straight-line interpolation scheme for all subsequent iterations.

Subroutine GASTBL

Subroutine GASTBL is used to interpolate in the array GS for gas properties, given the absolute temperature. Linear interpolation is used.

Subroutine GAUSS

Subroutine GAUSS is a routine that uses Gaussian elimination to solve a set of simultaneous equations. The array of coefficients, TCOF, is in the form of a compressed, augmented band matrix. That is, only the matrix elements within the band and the constants from the right side are stored in TCOF. The matrix band width, \( K \), is determined by the node-numbering system used. In TACT1, the temperature calculations require a band width of 23 elements, and the pressure calculations require 19.

Subroutine GETIN

Subroutine GETIN is a routine used to initialize input-data default values and to read and store input data. Input is in NAMELIST form as described in reference 1. The entire data set is read, and the input variables for each slice are stored in two arrays: INDCHN for integer data, and CHANL for real-number data. If the input is provided in SI units, subroutine GETIN converts it to U.S. customary units for internal use. If the user specifies INEDIT > 0, GETIN calls subroutine INPRT to print out the input data.
Function HCFRCD

Function subprogram HCFRCD is a routine to calculate a turbulent, forced-convection heat-transfer coefficient for channel flow as described in reference 1.

Subroutine HCOOL

Subroutine HCOOL is a routine containing the correlations for impingement heat transfer. The first part of HCOOL deals with leading-edge-region impingement cooling. In this part, the inner surface length from the stagnation impingement jet to the end of the leading-edge impingement region is determined and then used in a correlation to compute the average heat-transfer coefficient in this region. Beyond this region, for stations starting at ICOR, calculations are done by using an impingement-with-crossflow correlation.

Subroutine HCPINS

Subroutine HCPINS is a routine to calculate coolant-side heat-transfer coefficients in regions of the blade equipped with pin fins. In addition, the effective heat-transfer area, which accounts for the pin surface area and the pin-fin effectiveness, is calculated.

Subroutine INPRT

Subroutine INPRT is a routine to print a listing of the input data. Also, INPRT sets up the initial temperature distribution in the blade. Subroutine PREP is called for each slice to extract input data from the arrays JNDCHN and CHANL.

Subroutine PARRAY

Subroutine PARRAY is a routine to set up the matrix to be solved for coolant-channel pressure distribution. The equations used are detailed in reference 1.

The array of coefficients generated in PARRAY, TCOF, is in the form of a compressed, augmented band matrix. Coefficients that would be on the main diagonal of the full matrix are stored in column 10 of the TCOF array. The terms from the right side of the equations are stored in column 20 of TCOF.
Subroutine PLNUM

Subroutine PLNUM is a routine to calculate the pressure and temperature distributions in the central coolant plenum. The mean plenum static pressure and temperature for each slice are used as the supply conditions for the impingement jets. The total temperature and pressure at the outlet of one plenum slice are used as input for the next slice.

There are five arguments used in the call statement for PLNUM: WXX is the mass flow rate into this plenum slice; PXX and TXX are the calculated, average static pressure and temperature; and PTEXIT and TTEXIT are total temperature and pressure, respectively. Going into the subroutine, PTEXIT and TTEXIT are the values at the entrance to this slice. On return, they are the values at the exit of this slice.

Subroutine PLOTMF

Subroutine PLOTMF is a routine that plots TACT1 output. PLOTMF makes use of a TSS/360 graphics package at the NASA Lewis Research Center. For an installation without this specific package, this subroutine would have to be revised or bypassed.

PLOTMF plots temperature and pressure versus surface distance from station 1 for each slice of the blade for a steady-state case. For transients, a set of two summary plots is made for each time step: the plots contain temperatures for all slices on one graph.

Subroutine PREP

Subroutine PREP is a routine to extract input data from storage and put it in the form used in the calculations. In PREP, the hot-gas-side boundary condition tables are searched and linear interpolation is done to extract the boundary condition values at each calculation station at the given time.

Subroutine TARRAY

Subroutine TARRAY is a routine to set up the matrix to be solved for the temperatures in each slice. The equations used are detailed in reference 1.

The array of coefficients generated in TARRAY, TCOF, is in the form of a compressed, augmented band matrix. The 12th column of TCOF contains the elements that would be on the main diagonal of a full matrix. The terms from the right side of the equations are stored in column 24.
Subroutine TCOEF

Subroutine TCOEF is a routine that controls the calculations for flow rates, temperatures, and pressures. The first time TCOEF is entered for each slice an initial estimate of the coolant-channel pressure distribution is set up. TCOEF controls the iterations in loops A and B in figure 3. Loop A consists of calls to subroutines FLOWS, TARRAY, and PARRAY. The variable IVERGE is used to count the number of iterations in loop A. Convergence is checked by comparing the four most-recent values of coolant-channel pressure at the flow-split point. When the ratio of the maximum difference among these four to the difference between coolant-supply pressure and trailing-edge exit pressure is less than PCNVRG, loop A is complete. Then the flow split at the stagnation impingement jet, JS, is checked by subroutine FLSPLT and adjusted if necessary. Loop B involves repeating loop A for a new flow split. The variable IDELT is used to count the number of flow-split iterations in loop B. Once flow-split convergence is achieved, WROUT is called to print the output for the current slice.

Subroutine THRCON

Subroutine THRCON is a routine that takes the wall temperatures and searches for the thermal conductivity values in the input tables.

Subroutine WROUT

Subroutine WROUT is a routine to control the printing of the output from TACT1. Output units are the same as the input data units.

DICTIONARY

All the important variable names used in the TACT1 code are defined in this section. The only names not defined are locally used indices. All dimensioned variables include the dimensions. The dictionary also indicates the COMMON block or subroutine in which each variable is used.
<table>
<thead>
<tr>
<th>Variable</th>
<th>Common</th>
<th>Subroutine</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>A(400)</td>
<td>TCO</td>
<td></td>
<td>cross-sectional area normal to chord-wise direction, ( \text{in}^2 ), accessed by node number</td>
</tr>
<tr>
<td>AA</td>
<td></td>
<td>GETIN</td>
<td>outer-surface length between stations, ( \text{in} ), used for calculating interpolated values of TDLX(2), TDLX(3), and TDLX(5)</td>
</tr>
<tr>
<td>AA</td>
<td></td>
<td>PLNUM</td>
<td>coolant-plenum cross-sectional area, ( \text{in}^2 ), used in plenum pressure-drop calculations</td>
</tr>
<tr>
<td>AB</td>
<td></td>
<td>PLNUM</td>
<td>maximum Mach number in coolant plenum</td>
</tr>
<tr>
<td>AC(5)</td>
<td></td>
<td>GASTBL</td>
<td>array of interpolated values of gas properties</td>
</tr>
<tr>
<td>ACH</td>
<td></td>
<td>PLNUM</td>
<td>coolant-plenum choked-flow indicator</td>
</tr>
<tr>
<td>ADUMP</td>
<td></td>
<td>TCO</td>
<td>area of slot or jets dumping coolant directly into trailing-edge region, ( \text{in}^2 )</td>
</tr>
<tr>
<td>ADUMPC</td>
<td></td>
<td>INPRT</td>
<td>same as ADUMP, but converted to ( \text{cm}^2 ) for input listing when input is in SI units</td>
</tr>
<tr>
<td>AHG</td>
<td></td>
<td>PREP</td>
<td>intermediate value of hot-gas-side heat-transfer coefficient, ( \text{Btu/hr} \cdot \text{ft}^2 \cdot \circledast \text{F} ), used for interpolating in input table during a transient</td>
</tr>
<tr>
<td>AHTRN1</td>
<td></td>
<td>TARRAY</td>
<td>inner-surface area for heat-transfer purposes, ( \text{in}^2 )</td>
</tr>
<tr>
<td>AHTTR</td>
<td></td>
<td>HCPINS</td>
<td>total surface area in pin-fin channel, ( \text{in}^2 )</td>
</tr>
<tr>
<td>AINTRV</td>
<td></td>
<td>PLOTMF</td>
<td>floating-point form of number of temperature intervals in summary plots</td>
</tr>
<tr>
<td>AJ</td>
<td></td>
<td>PLNUM</td>
<td>floating-point form of indicator ( J - 1 )</td>
</tr>
<tr>
<td>AJET(80)</td>
<td></td>
<td>TCO</td>
<td>total area of impingement jets at each station, ( \text{in}^2 )</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>-------------</td>
<td>--------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AJET(80)</td>
<td></td>
<td>FLSPLT</td>
<td>total area of impingement jets at each station, in², carried into subroutine as argument</td>
</tr>
<tr>
<td>AKC(15, 80)</td>
<td>TCO</td>
<td></td>
<td>wall outer-coating thermal conductivity, Btu/hr • ft. °F</td>
</tr>
<tr>
<td>AKCTBL(20)</td>
<td>BOUND</td>
<td></td>
<td>input table of wall outer-coating thermal conductivity, Btu/hr • ft. °F, versus temperature, °F</td>
</tr>
<tr>
<td>AKW(15, 80)</td>
<td>TCO</td>
<td></td>
<td>wall metal thermal conductivity, Btu/hr • ft. °F</td>
</tr>
<tr>
<td>AKWTBL(20)</td>
<td>BOUND</td>
<td></td>
<td>input table of wall metal thermal conductivity, Btu/hr • ft. °F, versus temperature, °F</td>
</tr>
<tr>
<td>ALABL(7)</td>
<td></td>
<td>PLOTMF</td>
<td>array containing time and date label for identification of output plots</td>
</tr>
<tr>
<td>ALPH(12)</td>
<td>NTTACT</td>
<td></td>
<td>alphanumerical array used to uniquely identify output of each job</td>
</tr>
<tr>
<td>ALPHA FRIC</td>
<td></td>
<td>FRIC</td>
<td>constant used in friction factor calculations</td>
</tr>
<tr>
<td>ALPH2(4)</td>
<td></td>
<td>PLOTMF</td>
<td>time and date information, generated in TCOEF and passed to plotting subroutine as argument</td>
</tr>
<tr>
<td>AM</td>
<td>HCOOL</td>
<td></td>
<td>exponent on Reynolds number in Kercher-Tabakoff impingement correlation</td>
</tr>
<tr>
<td>AMC(20)</td>
<td></td>
<td>PLNUM</td>
<td>Mach number distribution in coolant plenum for a given slice</td>
</tr>
<tr>
<td>AMCHOK</td>
<td></td>
<td>FLOWS</td>
<td>if any stations have a Mach number greater than 1.0, the value is saved in this variable and returned as an argument, to be printed by TCOEF</td>
</tr>
<tr>
<td>AMIN</td>
<td></td>
<td>FLOWS</td>
<td>area of coolant-flow channel at a given station, reduced by pin-fin blockage</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AM2(G0)</td>
<td>TCO</td>
<td></td>
<td>array containing square of coolant-channel Mach number at each station, for a given slice</td>
</tr>
<tr>
<td>APG</td>
<td>PREP</td>
<td></td>
<td>intermediate value of hot-gas-side pressure, lbf/in², used for interpolating in input table during transient</td>
</tr>
<tr>
<td>APLEN</td>
<td>GETIN</td>
<td></td>
<td>input value of coolant-plenum area for given slice, cm² (in²)</td>
</tr>
<tr>
<td>APLN(15)</td>
<td>RADL</td>
<td></td>
<td>internal array to store plenum area for each slice, in²</td>
</tr>
<tr>
<td>AP1</td>
<td>GASTBL</td>
<td></td>
<td>interpolating factor in gas property table lookup</td>
</tr>
<tr>
<td>AP2</td>
<td>GASTBL</td>
<td></td>
<td>1.0 - AP1</td>
</tr>
<tr>
<td>AQG</td>
<td>PREP</td>
<td></td>
<td>intermediate value of hot-gas-side heat flux, Btu/hr · ft², used for interpolating in input table during transient</td>
</tr>
<tr>
<td>ASTG</td>
<td>TCOEF</td>
<td></td>
<td>inner-surface area under stagnation-point impingement jet, in²</td>
</tr>
<tr>
<td>ATG</td>
<td>PREP</td>
<td></td>
<td>intermediate value of hot-gas-side temperature, °R, used for interpolating in input table during transient</td>
</tr>
<tr>
<td>ATMAXXP</td>
<td>PLOTMF</td>
<td></td>
<td>adjusted maximum temperature, °F, used as high endpoint on output plots</td>
</tr>
<tr>
<td>ATMINP</td>
<td>PLOTMF</td>
<td></td>
<td>adjusted minimum temperature, °F, used as low endpoint on output plots</td>
</tr>
<tr>
<td>ATYME</td>
<td>PLOTMF</td>
<td></td>
<td>value of time in transient, sec, used on output plots for identification</td>
</tr>
<tr>
<td>AVRGA</td>
<td>PARRAY</td>
<td></td>
<td>area ratio used in momentum equation at entrance to trailing edge</td>
</tr>
<tr>
<td>AZ</td>
<td>PLNUM</td>
<td></td>
<td>dummy variable, used as either diameter-area ratio or flow adjustment</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>A1</td>
<td>FLOWS</td>
<td></td>
<td>interpolating factor in friction factor calculation in transitional Reynolds number range</td>
</tr>
<tr>
<td>A1</td>
<td>TARRAY</td>
<td></td>
<td>upstream half of inside-wall heat-transfer area, in², associated with coolant-channel node</td>
</tr>
<tr>
<td>A2</td>
<td>FLOWS</td>
<td></td>
<td>interpolating factor in friction factor calculation in transitional Reynolds number range</td>
</tr>
<tr>
<td>A2</td>
<td>TARRAY</td>
<td></td>
<td>downstream half of inner surface heat-transfer area, in², associated with coolant-channel node</td>
</tr>
<tr>
<td>A3</td>
<td>TARRAY</td>
<td></td>
<td>same as A1, but on opposite wall, only used in trailing-edge region</td>
</tr>
<tr>
<td>A4</td>
<td>TARRAY</td>
<td></td>
<td>same as A2, but on opposite wall, only used in trailing-edge region</td>
</tr>
<tr>
<td>B</td>
<td>GETIN</td>
<td></td>
<td>ratio of length to thickness, used along with AA to calculate interpolated values of TDLX(2), TDLX(3), and TDLX(5)</td>
</tr>
<tr>
<td>B(20)</td>
<td>PLNUM</td>
<td></td>
<td>spanwise static-temperature distribution, °R, in coolant plenum for given slice</td>
</tr>
<tr>
<td>BC</td>
<td>GETIN</td>
<td></td>
<td>NAMELIST name</td>
</tr>
<tr>
<td>BCHGP(1000)</td>
<td>BOUND</td>
<td></td>
<td>input table of hot-gas, pressure-side heat-transfer coefficients, W/m²·K (Btu/hr·ft²·ºF)</td>
</tr>
<tr>
<td>BCHGS(1000)</td>
<td>BOUND</td>
<td></td>
<td>input table of hot-gas, suction-side heat-transfer coefficients, W/m²·K (Btu/hr·ft²·ºF)</td>
</tr>
<tr>
<td>BCPGP(1000)</td>
<td>BOUND</td>
<td></td>
<td>input table of hot-gas, pressure-side relative static pressure, kPa (lbf/in²)</td>
</tr>
<tr>
<td>BCPGS(1000)</td>
<td>BOUND</td>
<td></td>
<td>input table of hot-gas, suction-side relative static pressure, kPa (lbf/in²)</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>--------</td>
<td>------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BCQGP(1000)</td>
<td>BOUND</td>
<td></td>
<td>input table of hot-gas, pressure-side heat flux, ( \text{W/m}^2 ) (\text{Btu/hr ft}^2)</td>
</tr>
<tr>
<td>BCQGS(1000)</td>
<td>BOUND</td>
<td></td>
<td>input table of hot-gas, suction-side heat flux, ( \text{W/m}^2 ) (\text{Btu/hr ft}^2)</td>
</tr>
<tr>
<td>BCTGP(1000)</td>
<td>BOUND</td>
<td></td>
<td>input table of hot-gas, pressure-side temperature, ( \text{K} ) (\text{OF})</td>
</tr>
<tr>
<td>BCTGS(1000)</td>
<td>BOUND</td>
<td></td>
<td>input table of hot-gas, suction-side temperature, ( \text{K} ) (\text{OF})</td>
</tr>
<tr>
<td>BCTIME(50)</td>
<td>BOUND</td>
<td></td>
<td>input table of time at which transient input tables are specified, \text{sec}</td>
</tr>
<tr>
<td>BCXP(400)</td>
<td>BOUND</td>
<td></td>
<td>input table of outer-surface, pressure-side locations at which hot-gas conditions are input, \text{cm} (\text{in.})</td>
</tr>
<tr>
<td>BCXS(400)</td>
<td>BOUND</td>
<td></td>
<td>input table of outer-surface, suction-side locations at which hot-gas conditions are input, \text{cm} (\text{in.})</td>
</tr>
<tr>
<td>BES</td>
<td>HCOOL</td>
<td></td>
<td>equivalent slot width, \text{in.}, used in leading-edge impingement correlation</td>
</tr>
<tr>
<td>BETA</td>
<td>FRIC</td>
<td></td>
<td>constant used in friction factor calculations</td>
</tr>
<tr>
<td>BETA1</td>
<td>PLNUM</td>
<td></td>
<td>square of pressure at inlet to coolant plenum for given slice, ((\text{lbf/in}^2)^2)</td>
</tr>
<tr>
<td>BETTA(20)</td>
<td>PLNUM</td>
<td></td>
<td>spanwise static-pressure distribution in coolant plenum, \text{lbf/in}^2</td>
</tr>
<tr>
<td>BTA</td>
<td>TCO</td>
<td></td>
<td>indicates type of hot-gas boundary condition</td>
</tr>
<tr>
<td>C</td>
<td>FLOWS</td>
<td></td>
<td>gas thermal conductivity, \text{Btu/hr ft} \text{ OF}</td>
</tr>
<tr>
<td></td>
<td>GASTBL</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HCFRCD</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HCOOL</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HCPINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PLNUM</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TARRAY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>CD</td>
<td>TCO</td>
<td></td>
<td>impingement-jet discharge coefficient</td>
</tr>
<tr>
<td>CDEN(2)</td>
<td>UNITS</td>
<td></td>
<td>conversion factor for density units</td>
</tr>
<tr>
<td>CD1(200)</td>
<td>NTTACT</td>
<td></td>
<td>dummy variable used to print selected intermediate temperature values</td>
</tr>
<tr>
<td>CEXCSW</td>
<td>NTTACT</td>
<td></td>
<td>amount of excess coolant flow, in SI units, kg/hr</td>
</tr>
<tr>
<td>CGASC(2)</td>
<td>UNITS</td>
<td></td>
<td>conversion factor for gas constant</td>
</tr>
<tr>
<td>CH(15)</td>
<td>PLNUM</td>
<td></td>
<td>coolant-channel choking indicator</td>
</tr>
<tr>
<td>CHANL(8000)</td>
<td>SPECL</td>
<td></td>
<td>array for storing input data</td>
</tr>
<tr>
<td>CHANLS</td>
<td>GETIN</td>
<td></td>
<td>NAMELIST name</td>
</tr>
<tr>
<td>CHFLX(2)</td>
<td>UNITS</td>
<td></td>
<td>conversion factor for heat-flux units</td>
</tr>
<tr>
<td>CHTC(2)</td>
<td>UNITS</td>
<td></td>
<td>conversion factor for heat-transfer-coefficient units</td>
</tr>
<tr>
<td>CIMP1</td>
<td>IMPCOR</td>
<td></td>
<td>user-supplied constants for general impingement correlation</td>
</tr>
<tr>
<td>CIMP2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIMP3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIMP4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIMP5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIMP6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIMP7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CINCH(2)</td>
<td>UNITS</td>
<td></td>
<td>conversion factor for length units</td>
</tr>
<tr>
<td>CMSFL(2)</td>
<td>UNITS</td>
<td></td>
<td>conversion factor for mass flow rate units</td>
</tr>
<tr>
<td>CNUM(80)</td>
<td>TCO</td>
<td></td>
<td>number of impingement jets at each station for given slice</td>
</tr>
<tr>
<td>CONDUCT</td>
<td>HCOOL</td>
<td></td>
<td>coolant-air thermal conductivity, Btu/hr ft °F</td>
</tr>
<tr>
<td>CONTROL</td>
<td>GETIN</td>
<td></td>
<td>NAMELIST name</td>
</tr>
<tr>
<td>CP</td>
<td>TCO</td>
<td></td>
<td>gas specific heat at constant pressure, Btu/lbm °F</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>--------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CPC(80)</td>
<td>PRPS</td>
<td></td>
<td>coolant specific heat at constant pressure at each coolant node for given slice, Btu/lbm ( \cdot ^{\circ} \text{F} ) evaluated at a mean temperature between bulk coolant temperature and wall temperature</td>
</tr>
<tr>
<td>CPIM</td>
<td>NTTACT</td>
<td>PLOTMF</td>
<td>mean impingement-plenum pressure for given slice, in SI units, kPa</td>
</tr>
<tr>
<td>CPM</td>
<td></td>
<td>FLOWS</td>
<td>hot-gas-stream specific heat at constant pressure, Btu/lbm ( \cdot ^{\circ} \text{F} )</td>
</tr>
<tr>
<td>CPO</td>
<td>PRPS</td>
<td></td>
<td>specific heat at constant pressure, Btu/lbm ( \cdot ^{\circ} \text{F} ), evaluated at impingement-jet supply temperature</td>
</tr>
<tr>
<td>CPRSR(2)</td>
<td></td>
<td></td>
<td>conversion factor for pressure units</td>
</tr>
<tr>
<td>CRHOVG(2)</td>
<td></td>
<td></td>
<td>conversion factor for density ( \times ) velocity units</td>
</tr>
<tr>
<td>CRITR</td>
<td></td>
<td>FLSPLT</td>
<td>coolant flow-split convergence criterion</td>
</tr>
<tr>
<td>CSPHT(2)</td>
<td></td>
<td></td>
<td>conversion factor for specific-heat units</td>
</tr>
<tr>
<td>CTCON(2)</td>
<td></td>
<td></td>
<td>conversion factor for thermal conductivity units</td>
</tr>
<tr>
<td>CTMPF(2)</td>
<td></td>
<td></td>
<td>conversion factor for temperature units</td>
</tr>
<tr>
<td>CT0G</td>
<td></td>
<td>NTTACT</td>
<td>mean impingement-plenum static temperature for given slice, in SI units, K</td>
</tr>
<tr>
<td>CURV</td>
<td></td>
<td>TARRAY</td>
<td>factor to account for wall curvature in heat-conduction equations</td>
</tr>
<tr>
<td>CVISC(2)</td>
<td></td>
<td></td>
<td>conversion factor for viscosity units</td>
</tr>
<tr>
<td>CWPLEN</td>
<td></td>
<td>NTTACT</td>
<td>coolant-plenum flow rate at entrance to given slice, in SI units, kg/hr</td>
</tr>
<tr>
<td>CWUSED</td>
<td></td>
<td>NTTACT</td>
<td>total amount of coolant air used, in SI units, kg/hr</td>
</tr>
<tr>
<td>CX</td>
<td></td>
<td>PLNUM</td>
<td>function of isentropic exponent ( k ), ( -(\kappa + 1)/[2(\kappa - 1)] )</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>C1</td>
<td>PLNUM</td>
<td>PLNUM</td>
<td>function of isentropic exponent $k$, $2k/(k - 1)$</td>
</tr>
<tr>
<td>C3</td>
<td>PLNUM</td>
<td>PLNUM</td>
<td>computed constant involving wheel speed and isentropic exponent</td>
</tr>
<tr>
<td>C3</td>
<td>FLOWS</td>
<td></td>
<td>ratio of specific heats at constant pressure, coolant to hot gas</td>
</tr>
<tr>
<td>C5</td>
<td>PLNUM</td>
<td>PLNUM</td>
<td>computed constant involving isentropic exponent and gas constant</td>
</tr>
<tr>
<td>C6</td>
<td>PLNUM</td>
<td>PLNUM</td>
<td>function of isentropic exponent $k$, $(k - 1)/2$</td>
</tr>
<tr>
<td>C7</td>
<td>PLNUM</td>
<td>PLNUM</td>
<td>computed constant involving isentropic exponent and gas constant</td>
</tr>
<tr>
<td>C8</td>
<td>PLNUM</td>
<td>PLNUM</td>
<td>computed constant involving isentropic exponent and gas constant</td>
</tr>
<tr>
<td>D</td>
<td>PLNUM</td>
<td>PLNUM</td>
<td>convergence parameter in coolant-plenum pressure calculations</td>
</tr>
<tr>
<td>DD</td>
<td>PLNUM</td>
<td>HCOOL</td>
<td>coolant-plenum hydraulic diameter, in</td>
</tr>
<tr>
<td>DEH</td>
<td>HCOOL</td>
<td></td>
<td>hydraulic diameter of equivalent slot, in., used in leading-edge impingement correlation</td>
</tr>
<tr>
<td>DELAST</td>
<td>FLSPLT</td>
<td></td>
<td>variable used to save flow-split fraction at which pressure-match parameter, EPSN, changes sign</td>
</tr>
<tr>
<td>DELTA</td>
<td>FRIC</td>
<td></td>
<td>constant used in friction factor calculation</td>
</tr>
<tr>
<td>DELTAN(15)</td>
<td>FLOWS</td>
<td>FLSPLT</td>
<td>fraction of stagnation-point impingement-jet flow that splits to suction-side</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HCPINS</td>
<td>coolant-flow channel for each slice</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TCOEF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WROUT</td>
<td></td>
</tr>
<tr>
<td>DELTAO</td>
<td>FLSPLT</td>
<td></td>
<td>value of DELTAN from previous flow-split iteration</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>DENOM</td>
<td></td>
<td>NTTACT</td>
<td>intermediate variable used in time interpolation of some boundary conditions</td>
</tr>
<tr>
<td>DH(80)</td>
<td>TCO</td>
<td></td>
<td>coolant-channel hydraulic diameter at each station, in.</td>
</tr>
<tr>
<td>DHF(80)</td>
<td>TCO</td>
<td></td>
<td>effective diameter of film-cooling hole at each station, in., defined as hydraulic diameter of one hole multiplied by square root of number of holes at station</td>
</tr>
<tr>
<td>DHJ(80)</td>
<td>TCO</td>
<td></td>
<td>actual hydraulic diameter of an impingement hole at each station, in.</td>
</tr>
<tr>
<td>DHYD</td>
<td>GETIN</td>
<td></td>
<td>input value of coolant-plenum hydraulic diameter for a slice, cm (in.)</td>
</tr>
<tr>
<td>DIFN</td>
<td>TCOEF</td>
<td></td>
<td>pressure difference parameter used in checking convergence</td>
</tr>
<tr>
<td>DIFO</td>
<td>TCOEF</td>
<td></td>
<td>maximum pressure difference parameter used in checking convergence</td>
</tr>
<tr>
<td>DIFTOL</td>
<td>PLNUM</td>
<td></td>
<td>tolerance on coolant-plenum pressure-drop calculations</td>
</tr>
<tr>
<td>DIMP1</td>
<td>IMPCOR</td>
<td></td>
<td>user-supplied constants for leading-edge impingement correlation</td>
</tr>
<tr>
<td>DIMP2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DIMP3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DIMP4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DIMP5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DIMP6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLTAOP</td>
<td></td>
<td>FLSPLT</td>
<td>best value of DELTAN in the event of an unstable flow split</td>
</tr>
<tr>
<td>DLYME</td>
<td></td>
<td>TRNSNT</td>
<td>time step used in transient calculations, sec</td>
</tr>
<tr>
<td>DLX(400)</td>
<td>TCO</td>
<td></td>
<td>chordwise distance from each node to adjacent upstream node, in.</td>
</tr>
<tr>
<td>DP(80)</td>
<td>PRPS</td>
<td></td>
<td>diameter of pin fins at each station, in.</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DPLN(15)</td>
<td>RADL</td>
<td></td>
<td>coolant-plenum hydraulic diameter for each slice, in.</td>
</tr>
<tr>
<td>DR</td>
<td>PLNUM</td>
<td></td>
<td>radial length increment in coolant-plenum calculations, in.</td>
</tr>
<tr>
<td>DR2</td>
<td>PLNUM</td>
<td></td>
<td>radial increment squared, in²</td>
</tr>
<tr>
<td>DUMR1(80)</td>
<td>PRPS</td>
<td></td>
<td>dummy variable, not currently used</td>
</tr>
<tr>
<td>DUMR2(80)</td>
<td>PRPS</td>
<td></td>
<td>dummy variable, used to carry impingement-jet Reynolds number to output subroutine</td>
</tr>
<tr>
<td>DUMTER</td>
<td>PARRAY</td>
<td></td>
<td>intermediate variable in momentum equation involving coolant dumped directly into trailing edge</td>
</tr>
<tr>
<td>DUM1(10)</td>
<td>INPRT</td>
<td>WROUT</td>
<td>dummy variables used to print input listings and program output</td>
</tr>
<tr>
<td>DUM2(10)</td>
<td>INPRT</td>
<td>WROUT</td>
<td></td>
</tr>
<tr>
<td>DUM3(10)</td>
<td>INPRT</td>
<td>WROUT</td>
<td></td>
</tr>
<tr>
<td>DUM4(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM5(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM6(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM7(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM8(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM9(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM10(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM11(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM12(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM13(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM14(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM15(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM16(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DUM17(10)</td>
<td></td>
<td>INPRT</td>
<td>dummy variables used to print input listings and program output</td>
</tr>
<tr>
<td>DUM18(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM19(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM20(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM25(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM52(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM53(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DUM55(10)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DX</td>
<td></td>
<td>PLNUM</td>
<td>spanwise step size used in calculating coolant-plenum pressure and temper-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ture distributions</td>
</tr>
<tr>
<td>DXTEMP</td>
<td></td>
<td>PLNUM</td>
<td>variable to temporarily hold DX</td>
</tr>
<tr>
<td>DX1</td>
<td></td>
<td>TARRAY</td>
<td>path length between midwall node and adjacent upstream midwall node, in.</td>
</tr>
<tr>
<td>DX10</td>
<td></td>
<td>TARRAY</td>
<td>path length between outer-coating-wall junction node and adjacent downstr</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>emdownstream outer-coating-wall junction node, in.</td>
</tr>
<tr>
<td>DX2</td>
<td></td>
<td>TARRAY</td>
<td>path length between midwall node and adjacent downstream midwall node, in</td>
</tr>
<tr>
<td>DX3</td>
<td></td>
<td>TARRAY</td>
<td>path length between outer-surface node and adjacent upstream outer-surface</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>node, in.</td>
</tr>
<tr>
<td>DX4</td>
<td></td>
<td>TARRAY</td>
<td>path length between outer-surface node and adjacent downstream outer-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>surface node, in.</td>
</tr>
<tr>
<td>DX5</td>
<td></td>
<td>TARRAY</td>
<td>path length between inner-surface node and adjacent upstream inner-surface</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>node, in.</td>
</tr>
<tr>
<td>DX6</td>
<td></td>
<td>TARRAY</td>
<td>path length between inner-surface node and adjacent downstream inner-sur</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>face node, in.</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>DX7</td>
<td>TARRAY</td>
<td></td>
<td>path length between coolant node and adjacent upstream coolant node, in.</td>
</tr>
<tr>
<td>DX9</td>
<td>TARRAY</td>
<td></td>
<td>path length between outer coating - wall junction node and adjacent upstream outer coating - wall junction node, in.</td>
</tr>
<tr>
<td>D1</td>
<td>PLNUM</td>
<td></td>
<td>computed constant used in coolant-plenum pressure equations to account for effect of pumping due to wheel rotation</td>
</tr>
<tr>
<td>D2</td>
<td>PLNUM</td>
<td></td>
<td>computed constant used in coolant-plenum temperature equations to account for effect of pumping due to wheel rotation</td>
</tr>
<tr>
<td>E</td>
<td>PLNUM</td>
<td></td>
<td>factor used in adjusting convergence rate in coolant-plenum calculations</td>
</tr>
<tr>
<td>EFAREA(80)</td>
<td></td>
<td>HCPINS</td>
<td>effective area, $\text{in}^2$, for heat transfer at stations with pin fins, including pin-fin effectiveness for heat transfer</td>
</tr>
<tr>
<td>EFTVNS</td>
<td>HCPINS</td>
<td>TARRAY</td>
<td>pin-fin effectiveness</td>
</tr>
<tr>
<td>EMES(80)</td>
<td></td>
<td>HCPINS</td>
<td>for film cooling, ratio of coolant mass flux to free-stream mass flux, multiplied by equivalent slot width</td>
</tr>
<tr>
<td>EML</td>
<td>HCPINS</td>
<td></td>
<td>term used in pin-fin effectiveness calculation</td>
</tr>
<tr>
<td>ENDEFF</td>
<td>TARRAY</td>
<td></td>
<td>term in heat-transfer equations to account for convection to rear edge of blade when heat-transfer coefficients are input</td>
</tr>
<tr>
<td>ENDFLX</td>
<td>TARRAY</td>
<td></td>
<td>term in heat-transfer equations to account for convection to rear edge of blade when heat flux is input</td>
</tr>
<tr>
<td>EPLAST</td>
<td>FLSPLT</td>
<td></td>
<td>variable used to save latest value of pressure-match parameter, EPSN</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>--------</td>
<td>--------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>EPS</td>
<td>FRIC</td>
<td></td>
<td>constant used in friction factor calculations</td>
</tr>
<tr>
<td>EPSMIN</td>
<td>FLSPLT</td>
<td></td>
<td>minimum value attained by pressure-match parameter, EPSN, for unstable flow split</td>
</tr>
<tr>
<td>EPSN</td>
<td>FLSPLT</td>
<td>TCOEF</td>
<td>pressure-match parameter, defined as difference between suction- and pressure-side coolant-channel static pressures at end of insert, divided by suction-side coolant-channel static pressure</td>
</tr>
<tr>
<td>EPSO</td>
<td>FLSPLT</td>
<td></td>
<td>old value of pressure-match parameter, EPSN</td>
</tr>
<tr>
<td>ETAPRM</td>
<td>FLOWS</td>
<td></td>
<td>film-cooling effectiveness based on ratio of enthalpy differences</td>
</tr>
<tr>
<td>EXCESW</td>
<td>NTTACT</td>
<td></td>
<td>amount of excess coolant flow, difference between inlet flow and that actually used, lbm/hr</td>
</tr>
<tr>
<td>FACTOR</td>
<td>TARRAY</td>
<td></td>
<td>special variable to adjust amount of energy carried in by an impingement jet for case of a calculation station adjacent to flow-split station</td>
</tr>
<tr>
<td>FF(80)</td>
<td>TCO</td>
<td></td>
<td>value of friction factor at each flow station</td>
</tr>
<tr>
<td>FILM</td>
<td>PARRAY</td>
<td></td>
<td>term to account for momentum carried off by film-cooling air</td>
</tr>
<tr>
<td>FILMW</td>
<td>TARRAY</td>
<td></td>
<td>total film-cooling flow from given coolant node</td>
</tr>
<tr>
<td>FLMEFF(80)</td>
<td>FLMCOL</td>
<td></td>
<td>film-cooling effectiveness based on ratio of temperature differences</td>
</tr>
<tr>
<td>FM</td>
<td>GAUSS</td>
<td></td>
<td>multiplying factor used in Gauss elimination scheme</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FUNP</td>
<td></td>
<td>PLNUM</td>
<td>statement function to calculate pressure difference in coolant plenum</td>
</tr>
<tr>
<td>FUNT</td>
<td></td>
<td>PLNUM</td>
<td>statement function to calculate temperature difference in coolant plenum</td>
</tr>
<tr>
<td>F1(20)</td>
<td></td>
<td>PLNUM</td>
<td>friction factor in coolant plenum</td>
</tr>
<tr>
<td>GAM</td>
<td></td>
<td>TCO</td>
<td>ratio of specific heats</td>
</tr>
<tr>
<td>GAMC(80)</td>
<td></td>
<td>PRPS</td>
<td>ratio of specific heats at each coolant-channel node</td>
</tr>
<tr>
<td>GAMO</td>
<td></td>
<td>PRPS</td>
<td>ratio of specific heats at coolant-supply conditions</td>
</tr>
<tr>
<td>GEO</td>
<td></td>
<td>GETIN</td>
<td>NAMELIST name</td>
</tr>
<tr>
<td>GG</td>
<td></td>
<td>HCOOL</td>
<td>mass flux ratio, coolant crossflow to impingement-jet flow</td>
</tr>
<tr>
<td>GI</td>
<td></td>
<td>HCOOL</td>
<td>momentum flux ratio, coolant crossflow to impingement-jet flow</td>
</tr>
<tr>
<td>GMASS</td>
<td></td>
<td>HCOOL</td>
<td>mass flux from row of leading-edge impingement holes</td>
</tr>
<tr>
<td>GS(200)</td>
<td></td>
<td>GAAS</td>
<td>table of gas properties</td>
</tr>
<tr>
<td>G1</td>
<td></td>
<td>PLNUM</td>
<td>computed constant in coolant-plenum calculations, involving flow rate, gas</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>constant, and specific heat at constant pressure</td>
</tr>
<tr>
<td>G2</td>
<td></td>
<td>PLNUM</td>
<td>computed constant in coolant-plenum calculations, involving flow rate and</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>gas constant</td>
</tr>
<tr>
<td>HBAR</td>
<td></td>
<td>WROUT</td>
<td>average coolant-side heat-transfer coefficient for given slice, W/m²·K</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Btu/hr · ft² · °F)</td>
</tr>
<tr>
<td>HC(80)</td>
<td></td>
<td>TCO</td>
<td>coolant-side heat-transfer coefficients at each station for given slice,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Btu/hr · ft² · °F</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>HCAL (4)</td>
<td>INPRT</td>
<td></td>
<td>alphameric array containing labels identifying type of coolant-side heat transfer</td>
</tr>
<tr>
<td>HG (80)</td>
<td>TCO</td>
<td></td>
<td>hot-gas-side heat-transfer coefficient at each station for given slice, Btu/hr \cdot ft^2 \cdot ^\circ F</td>
</tr>
<tr>
<td>HSTGMMX</td>
<td>TCOEF</td>
<td></td>
<td>maximum physically possible value of coolant heat-transfer coefficient under stagnation jet, Btu/hr \cdot ft^2 \cdot ^\circ F</td>
</tr>
<tr>
<td>HUB1</td>
<td>TARRAY</td>
<td></td>
<td>term in conduction equation to account for specified hub temperature</td>
</tr>
<tr>
<td>HUB3</td>
<td>TARRAY</td>
<td></td>
<td>term in conduction equation to account for specified hub heat flux</td>
</tr>
<tr>
<td>HX</td>
<td>TARRAY</td>
<td></td>
<td>multiplying factor on coolant heat-transfer coefficient, initialized to 1.0 but may be changed dynamically</td>
</tr>
<tr>
<td>HYCOS</td>
<td>HCPINS</td>
<td></td>
<td>hyperbolic cosine term</td>
</tr>
<tr>
<td>HYSIN</td>
<td>HCPINS</td>
<td></td>
<td>hyperbolic sine term</td>
</tr>
<tr>
<td>IADJIN</td>
<td>SPECL</td>
<td></td>
<td>input variable that indicates which coolant-plenum supply property is to be held fixed</td>
</tr>
<tr>
<td>ICHK</td>
<td>PARRAY</td>
<td></td>
<td>indicates which side of blade a given station is on: 0 if suction side, 1 if pressure side</td>
</tr>
<tr>
<td>ICHNL</td>
<td>INPRT</td>
<td>PREP</td>
<td>slice number, carried through as argument</td>
</tr>
<tr>
<td>ICHOK</td>
<td>FLOWS</td>
<td></td>
<td>number of station that shows choked coolant flow</td>
</tr>
<tr>
<td>ICHOKP</td>
<td>PARRAY</td>
<td>TCOEF</td>
<td>number of station adjacent to impingement flow-split station in pressure-side direction</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ICOMS</td>
<td></td>
<td>TARRAY</td>
<td>number of station adjacent to impingement flow-split station in suction-side direction</td>
</tr>
<tr>
<td>ICONV</td>
<td></td>
<td>FLSPLT</td>
<td>indicator for convergence of flow-split iterations</td>
</tr>
<tr>
<td>ICOR</td>
<td>TCO</td>
<td>TCOEF</td>
<td>station at which use of impingement-with-crossflow correlation is to begin</td>
</tr>
<tr>
<td>IDELT</td>
<td></td>
<td>FLSPLT</td>
<td>counter of number of flow-split iterations performed</td>
</tr>
<tr>
<td>IDELT</td>
<td></td>
<td>TCOEF</td>
<td></td>
</tr>
<tr>
<td>IDNL</td>
<td></td>
<td>WROUT</td>
<td></td>
</tr>
<tr>
<td>IDN</td>
<td></td>
<td>PARRAY</td>
<td>downstream node number for coolant-channel pressure calculations</td>
</tr>
<tr>
<td>IDNS</td>
<td></td>
<td>PARRAY</td>
<td>downstream station number for coolant-channel pressure calculations</td>
</tr>
<tr>
<td>IDX</td>
<td></td>
<td>PARRAY</td>
<td>upstream node number for coolant-channel pressure calculations</td>
</tr>
<tr>
<td>IEND</td>
<td></td>
<td>GETIN</td>
<td>last point in CHANL array occupied by data for given slice</td>
</tr>
<tr>
<td>IFCP</td>
<td></td>
<td>FLOWS</td>
<td>indicator used in locating pressure-side film-cooling holes</td>
</tr>
<tr>
<td>IFCS</td>
<td></td>
<td>FLOWS</td>
<td>indicator used in locating suction-side film-cooling holes</td>
</tr>
<tr>
<td>IFILM</td>
<td>TCO</td>
<td></td>
<td>input indicator for film cooling</td>
</tr>
<tr>
<td>IFLU</td>
<td></td>
<td>PREP</td>
<td>coolant-channel node number</td>
</tr>
<tr>
<td>IFNL</td>
<td></td>
<td>PARRAY</td>
<td>number of coolant-channel nodes</td>
</tr>
<tr>
<td>IFNL</td>
<td></td>
<td>TCOEF</td>
<td>total number of stations, minus 1</td>
</tr>
<tr>
<td>IFSPLT</td>
<td></td>
<td>FLOWS</td>
<td>indicates in which direction film-cooling air flows from stagnation station</td>
</tr>
<tr>
<td>IGG(80)</td>
<td></td>
<td>HCOOL</td>
<td>array containing node numbers at which ratio of coolant crossflow to impingement-jet flow is out of Kercher-Tabakoff correlation range</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IGGC</td>
<td>HCOOL</td>
<td></td>
<td>counts number of entries in IGG array</td>
</tr>
<tr>
<td>IHC(80)</td>
<td>TCO</td>
<td></td>
<td>indicates type of coolant-side heat transfer at each station for given slice</td>
</tr>
<tr>
<td>IHCT</td>
<td>GETIN</td>
<td></td>
<td>input value of IHC for given station</td>
</tr>
<tr>
<td>IHUB</td>
<td>TCO</td>
<td></td>
<td>indicates type of boundary to be used at hub end of blade</td>
</tr>
<tr>
<td>II</td>
<td>HCOOL</td>
<td></td>
<td>coolant-channel node number</td>
</tr>
<tr>
<td>IIHCTZ</td>
<td>GETIN</td>
<td></td>
<td>locates IHC array in overall array</td>
</tr>
<tr>
<td>ILEAD</td>
<td>HCOOL</td>
<td></td>
<td>last station in range of leading-edge impingement correlation</td>
</tr>
<tr>
<td>IMS</td>
<td>FLOWS</td>
<td></td>
<td>location of film-cooling hole preceding current station</td>
</tr>
<tr>
<td>INDCHN(2000)</td>
<td>SPECL</td>
<td></td>
<td>array for storing integer input data</td>
</tr>
<tr>
<td>INEDIT</td>
<td>GETIN</td>
<td></td>
<td>input variable to control listing of input data</td>
</tr>
<tr>
<td>INN</td>
<td>PREP</td>
<td></td>
<td>location of IHC data in INDCHN array</td>
</tr>
<tr>
<td>INSTRRT</td>
<td>GETIN</td>
<td></td>
<td>starting point for storage of integer data in INDCHN array for given slice</td>
</tr>
<tr>
<td>INUM</td>
<td>NTTACT</td>
<td></td>
<td>number of stations on each side of blade</td>
</tr>
<tr>
<td>IN1</td>
<td>PREP</td>
<td></td>
<td>location of end of group of single-valued variables in INDCHN array</td>
</tr>
<tr>
<td>IPILOT</td>
<td>SPECL</td>
<td></td>
<td>input indicator to control plotting options</td>
</tr>
<tr>
<td>IPRES</td>
<td>WROUT</td>
<td></td>
<td>pressure-side, outer-surface node number</td>
</tr>
<tr>
<td>IPRSMN</td>
<td>WROUT</td>
<td></td>
<td>location of minimum outer-surface temperature on pressure side</td>
</tr>
<tr>
<td>IPRSMX</td>
<td>WROUT</td>
<td></td>
<td>location of maximum outer-surface temperature on pressure side</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>IRE(80)</td>
<td></td>
<td>HCOOL</td>
<td>list of station numbers at which impingement-jet Reynolds number is out of Kercher-Tabakoff correlation range</td>
</tr>
<tr>
<td>IRL</td>
<td></td>
<td>PARRAY</td>
<td>coolant node number</td>
</tr>
<tr>
<td>ISBLOK</td>
<td>TCO</td>
<td></td>
<td>starting point in CHANL array of data for given slice</td>
</tr>
<tr>
<td>ISEN</td>
<td>TARRAY</td>
<td></td>
<td>indicates which side of blade station IS is on</td>
</tr>
<tr>
<td>ISENS</td>
<td>FLOWS</td>
<td></td>
<td>indicates which side of blade a given station is on</td>
</tr>
<tr>
<td>ISENS</td>
<td>TARRAY</td>
<td></td>
<td>indicates which side of blade a given trailing-edge region station is on</td>
</tr>
<tr>
<td>ISLICE</td>
<td>TCO</td>
<td></td>
<td>current slice number</td>
</tr>
<tr>
<td>ISTA</td>
<td>GETIN</td>
<td></td>
<td>input station number</td>
</tr>
<tr>
<td>ISTAT</td>
<td>WROUT</td>
<td></td>
<td>outer-surface node number for given station</td>
</tr>
<tr>
<td>ISTATD</td>
<td>WROUT</td>
<td></td>
<td>outer-surface node number immediately downstream of ISTAT</td>
</tr>
<tr>
<td>ISTB</td>
<td>GETIN</td>
<td></td>
<td>input station number</td>
</tr>
<tr>
<td>ISTRRT</td>
<td>HCOOL</td>
<td></td>
<td>station at which use of Kercher-Tabakoff correlation begins</td>
</tr>
<tr>
<td>ISTRRT</td>
<td>TCOEF</td>
<td></td>
<td>first station in trailing-edge region</td>
</tr>
<tr>
<td>ISUCMN</td>
<td>WROUT</td>
<td></td>
<td>location of minimum outer-surface temperature on suction side</td>
</tr>
<tr>
<td>ISUCMX</td>
<td>WROUT</td>
<td></td>
<td>location of maximum outer-surface temperature on suction side</td>
</tr>
<tr>
<td>ISUCT</td>
<td>WROUT</td>
<td></td>
<td>suction-side station number</td>
</tr>
<tr>
<td>ISUP</td>
<td>TARRAY</td>
<td></td>
<td>adjacent station, upstream of current station, IS</td>
</tr>
<tr>
<td>ISYM(5)</td>
<td>PLOTF</td>
<td></td>
<td>data array containing plotting symbol codes</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>ITDHFZ</td>
<td>GETIN</td>
<td></td>
<td>locates film-cooling-hole data in CHANL array for given slice</td>
</tr>
<tr>
<td>ITDHJZ</td>
<td>GETIN</td>
<td></td>
<td>locates impingement-hole data in CHANL array for given slice</td>
</tr>
<tr>
<td>ITDLXZ</td>
<td>GETIN</td>
<td></td>
<td>locates node spacing data in CHANL array for given slice</td>
</tr>
<tr>
<td>ITDPZ</td>
<td>GETIN</td>
<td></td>
<td>locates pin-fin diameter data in CHANL array for given slice</td>
</tr>
<tr>
<td>ITHKZ</td>
<td>GETIN</td>
<td></td>
<td>locates wall and channel thickness data in CHANL array for given slice</td>
</tr>
<tr>
<td>ITIP</td>
<td>TCO</td>
<td></td>
<td>indicates type of boundary to be used at tip end of blade</td>
</tr>
<tr>
<td>ITRBG</td>
<td>INPRT</td>
<td></td>
<td>first station in trailing-edge region</td>
</tr>
<tr>
<td>ITRBG</td>
<td>WROUT</td>
<td></td>
<td>first station in trailing-edge region</td>
</tr>
<tr>
<td>ITREO</td>
<td>WROUT</td>
<td></td>
<td>last outside node at trailing edge</td>
</tr>
<tr>
<td>IRRRZ</td>
<td>GETIN</td>
<td></td>
<td>locates radial position data in CHANL array for given slice</td>
</tr>
<tr>
<td>ITSPZ</td>
<td>GETIN</td>
<td></td>
<td>locates pin-fin spacing data in CHANL array for given slice</td>
</tr>
<tr>
<td>ITXNZ</td>
<td>GETIN</td>
<td></td>
<td>locates impingement-hole spacing data in CHANL array for given slice</td>
</tr>
<tr>
<td>IUNITS</td>
<td>UNITS</td>
<td></td>
<td>indicates system of units used for input data</td>
</tr>
<tr>
<td>IUNSTB</td>
<td>FLSPLT</td>
<td></td>
<td>indicates whether flow split is stable or not</td>
</tr>
<tr>
<td>IUP</td>
<td>PARRAY</td>
<td></td>
<td>upstream node number for coolant-channel pressure calculations</td>
</tr>
<tr>
<td>IUPS</td>
<td>PARRAY</td>
<td></td>
<td>upstream station number for coolant-channel pressure calculations</td>
</tr>
<tr>
<td>IVARS(12)</td>
<td>PLOTMF</td>
<td></td>
<td>array of integer plotting controls</td>
</tr>
<tr>
<td>IVERGE</td>
<td>TCOEF</td>
<td></td>
<td>pressure iteration loop counter</td>
</tr>
<tr>
<td></td>
<td>WROUT</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

30
<table>
<thead>
<tr>
<th>Variable</th>
<th>Common</th>
<th>Subroutine</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>IWR</td>
<td></td>
<td>GAUSS</td>
<td>control on debugging output of coefficient matrix</td>
</tr>
<tr>
<td>IWRITE</td>
<td>SPECL</td>
<td></td>
<td>input control on amount of printed output</td>
</tr>
<tr>
<td>DXAX</td>
<td>PLOTMF</td>
<td></td>
<td>logical variable with value .TRUE.</td>
</tr>
<tr>
<td>IYAX</td>
<td>PLOTMF</td>
<td></td>
<td>logical variable with value .FALSE.</td>
</tr>
<tr>
<td>I1</td>
<td>PREP</td>
<td></td>
<td>starting point in INDCHN array for integer data for given slice, also used as starting point for nodal data in CHANL array for given slice</td>
</tr>
<tr>
<td>I3</td>
<td>PREP</td>
<td></td>
<td>starting point in CHANL array for station data</td>
</tr>
<tr>
<td>JDIS</td>
<td>FLOWS</td>
<td></td>
<td>number of stations that impingement flow-split station is displaced from station 1</td>
</tr>
<tr>
<td>JHCAL</td>
<td>INPRT</td>
<td></td>
<td>indicates type of heat transfer at given station</td>
</tr>
<tr>
<td>JLSTM</td>
<td>THRCON</td>
<td></td>
<td>size of wall thermal conductivity tables</td>
</tr>
<tr>
<td>JNUMS</td>
<td>FLSPLT</td>
<td></td>
<td>indicates whether previous call to FLSPLT resulted in unstable flow split</td>
</tr>
<tr>
<td>JOUTRG</td>
<td>FLSPLT</td>
<td></td>
<td>indicates attempt to split more than 100 percent of flow to one side</td>
</tr>
<tr>
<td>JPIV</td>
<td>GAUSS</td>
<td></td>
<td>pivotal column in matrix to be reduced</td>
</tr>
<tr>
<td>JS</td>
<td>FLOWS</td>
<td>FLSPLT</td>
<td>station number at impingement flow-split point</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HCOOL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TCOEF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WROUT</td>
<td></td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>JSENS</td>
<td>FLOWS</td>
<td>GETIN</td>
<td>indicates which side of blade impingement flow-split is on</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FLSPLT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TCOEF</td>
<td></td>
</tr>
<tr>
<td>JSGNCH</td>
<td></td>
<td>FLSPLT</td>
<td>indicates whether a sign change has occurred in EPSN</td>
</tr>
<tr>
<td>JSO(15)</td>
<td></td>
<td>TCOEF</td>
<td>array for saving converged-flow-split station number for each slice</td>
</tr>
<tr>
<td>JSOLDS(25)</td>
<td></td>
<td>FLSPLT</td>
<td>array to keep track of stations checked as flow-split stations</td>
</tr>
<tr>
<td>JTIMES</td>
<td></td>
<td>FLSPLT</td>
<td>indicates whether DELTAN is to be rough adjusted or fine tuned</td>
</tr>
<tr>
<td>K</td>
<td>GAUSS</td>
<td></td>
<td>total bandwidth of matrix to be solved</td>
</tr>
<tr>
<td>K</td>
<td></td>
<td>NTTACT</td>
<td>counter of number of overall coolant-flow iterations</td>
</tr>
<tr>
<td>KSIG</td>
<td></td>
<td>PLNUM</td>
<td>coolant-plenum iteration counter</td>
</tr>
<tr>
<td>L</td>
<td></td>
<td>TARRAY</td>
<td>midwall node number at given station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>THRCON</td>
<td></td>
</tr>
<tr>
<td>LCOOL</td>
<td>FLOWS</td>
<td>HCFRCD</td>
<td>coolant-channel node number at given station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HCOOL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HCPINS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TCOEF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>WROUT</td>
<td></td>
</tr>
<tr>
<td>LCOOLP</td>
<td></td>
<td>HCPINS</td>
<td>inner-surface node across coolant-channel from given station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td>LCUP</td>
<td></td>
<td>HCPINS</td>
<td>coolant-channel node number upstream of given station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td>LCUPP</td>
<td></td>
<td>TARRAY</td>
<td>inner-surface node across coolant channel and upstream of given station</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>LCUPS</td>
<td>TARRAY</td>
<td></td>
<td>inner-surface node upstream of given station</td>
</tr>
<tr>
<td>LDN</td>
<td>TARRAY</td>
<td></td>
<td>midwall node number downstream of given station</td>
</tr>
<tr>
<td>LIN</td>
<td>FLOWS</td>
<td></td>
<td>inner-surface node number at given station</td>
</tr>
<tr>
<td></td>
<td>HCFRCD</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HCOOL</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HCPINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TARRAY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LJ</td>
<td>INPRT</td>
<td></td>
<td>node number at junction of outer coating and wall metal at given station</td>
</tr>
<tr>
<td></td>
<td>TARRAY</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>THRCON</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOUT</td>
<td>TARRAY</td>
<td></td>
<td>wall outer-surface node number at given station</td>
</tr>
<tr>
<td></td>
<td>THRCON</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LSP</td>
<td>PREP</td>
<td></td>
<td>starting point in BCXP array of data for given slice</td>
</tr>
<tr>
<td>LSS</td>
<td>PREP</td>
<td></td>
<td>starting point in BCXS array of data for given slice</td>
</tr>
<tr>
<td>LUP</td>
<td>TARRAY</td>
<td></td>
<td>midwall node number upstream of given station</td>
</tr>
<tr>
<td>MACH1</td>
<td>PLNUM</td>
<td></td>
<td>indicator to keep track of step-size change</td>
</tr>
<tr>
<td>MD1</td>
<td>SPECL</td>
<td></td>
<td>indicator to control special condensed, on-line output</td>
</tr>
<tr>
<td>MD2</td>
<td>SPECL</td>
<td></td>
<td>indicator that job is complete and summary plots are to be produced</td>
</tr>
<tr>
<td>MD3</td>
<td>SPECL</td>
<td></td>
<td>plot counter</td>
</tr>
<tr>
<td>NBC</td>
<td>INPRT</td>
<td></td>
<td>maximum of NBCS and NBCP</td>
</tr>
<tr>
<td>N</td>
<td>GAUSS</td>
<td></td>
<td>number of rows in matrix to be solved</td>
</tr>
<tr>
<td>NAG</td>
<td>PLNUM</td>
<td></td>
<td>indicator whether calculations have progressed beyond initial station</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>---------------</td>
<td>---------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>NBCP</td>
<td>BOUND</td>
<td></td>
<td>input number of boundary condition points on pressure side</td>
</tr>
<tr>
<td>NBCS</td>
<td>BOUND</td>
<td></td>
<td>input number of boundary condition points on suction side</td>
</tr>
<tr>
<td>NBFRP</td>
<td>INPRT</td>
<td></td>
<td>number of pressure-side boundary condition points preceding data for given</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>slice at given time</td>
</tr>
<tr>
<td>NBFRS</td>
<td>INPRT</td>
<td></td>
<td>number of suction-side boundary condition points preceding data for given</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>slice at given time</td>
</tr>
<tr>
<td>NBLKSZ</td>
<td>TCO</td>
<td></td>
<td>size of data block in CHANL array for given slice</td>
</tr>
<tr>
<td>NCC</td>
<td>PLNUM</td>
<td></td>
<td>loop counter</td>
</tr>
<tr>
<td>NCHAR</td>
<td>NTTACT</td>
<td></td>
<td>number of characters in ALPH2 array</td>
</tr>
<tr>
<td>NCOOL</td>
<td>PLOTMF</td>
<td></td>
<td>coolant node number</td>
</tr>
<tr>
<td>NEND</td>
<td>HCOOL</td>
<td></td>
<td>end of region that uses leading-edge impingement correlation</td>
</tr>
<tr>
<td>NFC</td>
<td>FLOWS</td>
<td></td>
<td>number of station containing film-cooling holes</td>
</tr>
<tr>
<td>NFCSUP(80)</td>
<td>FLMCOL</td>
<td></td>
<td>array identifying node supplying film cooling to each downstream node</td>
</tr>
<tr>
<td>NFLUID(200)</td>
<td>INPRT</td>
<td></td>
<td>array of coolant-channel node numbers for each station</td>
</tr>
<tr>
<td>NFWD</td>
<td>TCO</td>
<td></td>
<td>number of stations in forward region</td>
</tr>
<tr>
<td>NG</td>
<td>GAAS</td>
<td></td>
<td>number of temperature entries in gas property table, GS</td>
</tr>
<tr>
<td>NGEOM</td>
<td>GETIN</td>
<td></td>
<td>number of NAMELISTs /GEO/ to be read in</td>
</tr>
<tr>
<td>NINTRV</td>
<td>PLOTMF</td>
<td></td>
<td>number of temperature intervals in summary plots</td>
</tr>
<tr>
<td>NIT</td>
<td>TCOEF</td>
<td></td>
<td>counter of number of overall coolant-flow iterations</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>NL</td>
<td>INPRT</td>
<td></td>
<td>output line counter</td>
</tr>
<tr>
<td>NLBLS</td>
<td>PLOTMF</td>
<td></td>
<td>number of points at which symbols are to be plotted</td>
</tr>
<tr>
<td>NMM</td>
<td>PLOTMF</td>
<td></td>
<td>midwall node number</td>
</tr>
<tr>
<td>NMW</td>
<td>NTTACT</td>
<td></td>
<td>outer-surface node number</td>
</tr>
<tr>
<td>NODM</td>
<td>WROUT</td>
<td></td>
<td>midwall node</td>
</tr>
<tr>
<td>NODOUT</td>
<td>GETIN</td>
<td></td>
<td>outer-surface node</td>
</tr>
<tr>
<td>NODSF</td>
<td>FLOWS</td>
<td>GETIN</td>
<td>number of nodes in forward region</td>
</tr>
<tr>
<td></td>
<td></td>
<td>INPRT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PREP</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>FLSPLT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TCOEF</td>
<td></td>
</tr>
<tr>
<td>NODST</td>
<td>GETIN</td>
<td></td>
<td>total number of nodes for given slice</td>
</tr>
<tr>
<td></td>
<td></td>
<td>INPRT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PREP</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TCOEF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>NTTACT</td>
<td></td>
</tr>
<tr>
<td>NODSTM</td>
<td>INPRT</td>
<td></td>
<td>total number of nodes minus 4</td>
</tr>
<tr>
<td>NOS</td>
<td>FLOWS</td>
<td>INPRT</td>
<td>outer-surface node number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WROUT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PLOTMF</td>
<td></td>
</tr>
<tr>
<td>NPRCP</td>
<td>INPRT</td>
<td></td>
<td>number of points in each pressure-side boundary condition array for times preceding current time</td>
</tr>
<tr>
<td>NPRCS</td>
<td>INPRT</td>
<td></td>
<td>number of points in each suction-side boundary condition array for times preceding current time</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>NPRTP</td>
<td>INPRT</td>
<td></td>
<td>number of points in each pressure-side boundary condition array per time step</td>
</tr>
<tr>
<td>NPRTS</td>
<td>INPRT</td>
<td></td>
<td>number of points in each suction-side boundary condition array per time step</td>
</tr>
<tr>
<td>NPTS</td>
<td>PLOTMF</td>
<td></td>
<td>number of points on given plot</td>
</tr>
<tr>
<td>NROW</td>
<td>GAUSS</td>
<td></td>
<td>number of matrix row to be displayed by debug output</td>
</tr>
<tr>
<td>NSAVE</td>
<td>TCOEF</td>
<td></td>
<td>coolant node number just upstream of exit, location of TSAVE</td>
</tr>
<tr>
<td>NSLICE</td>
<td>TCO</td>
<td></td>
<td>current slice number</td>
</tr>
<tr>
<td>NSTA</td>
<td>TCO</td>
<td></td>
<td>number of stations per slice</td>
</tr>
<tr>
<td>NSTAPS</td>
<td>PLOTMF</td>
<td></td>
<td>number of stations on each side of blade</td>
</tr>
<tr>
<td>NSTNS</td>
<td>PLNUM</td>
<td></td>
<td>number of spanwise stations per slice in coolant plenum</td>
</tr>
<tr>
<td>NTBC</td>
<td>GETIN</td>
<td></td>
<td>number of entries in input BCTIME array</td>
</tr>
<tr>
<td>NTIMES</td>
<td>INPRT</td>
<td></td>
<td>number of entries in BCTIME array</td>
</tr>
<tr>
<td>NTTG</td>
<td>PREP</td>
<td></td>
<td>time step number</td>
</tr>
<tr>
<td>NTYM</td>
<td>NTTACT</td>
<td></td>
<td>number of time steps in transient</td>
</tr>
<tr>
<td>NUMS</td>
<td>FLSPLT</td>
<td></td>
<td>counter to force at least four attempts at a good flow split</td>
</tr>
<tr>
<td>P(2, 15, 80)</td>
<td>TCO</td>
<td></td>
<td>pressure at each node, for two consecutive time steps, ( \text{lbf/in}^2 )</td>
</tr>
<tr>
<td>PAVG</td>
<td>FLOWS</td>
<td></td>
<td>coolant-channel static pressure, ( \text{lbf/in}^2 ), used in calculating impingement hole flow rates</td>
</tr>
<tr>
<td>PBAR</td>
<td>FLOWS</td>
<td></td>
<td>pressure used in calculating square of coolant-channel Mach number</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PCNVRG</td>
<td></td>
<td>TCOEF</td>
<td>pressure-difference convergence criterion</td>
</tr>
<tr>
<td>PD</td>
<td></td>
<td>FLOWS</td>
<td>Prandtl number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GASTBL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HCFRCD</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HCOOL</td>
<td>Prandtl number based on coolant-supply temperature</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HCPINS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PLNUM</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td>PDTOG</td>
<td></td>
<td>HCOOL</td>
<td></td>
</tr>
<tr>
<td>PEX(400)</td>
<td>BOUND</td>
<td></td>
<td>input array containing tables of static pressure at trailing-edge coolant exhaust, lbf/in²</td>
</tr>
<tr>
<td>PEXC</td>
<td></td>
<td>INPRT</td>
<td>static pressure at trailing-edge coolant exhaust in SI units, kPa, for given slice</td>
</tr>
<tr>
<td>PEXIT(15)</td>
<td>TCO</td>
<td></td>
<td>static pressures at trailing-edge coolant exhaust for each slice at given time, lbf/in²</td>
</tr>
<tr>
<td>PEXOLD(15)</td>
<td>TCOEF</td>
<td></td>
<td>saved value of exhaust static pressure, lbf/in², used in setting initial guess of pressure distribution for subsequent time step</td>
</tr>
<tr>
<td>PEXTT</td>
<td>PLNUM</td>
<td></td>
<td>total pressure at exit of coolant plenum for given slice, lbf/in²</td>
</tr>
<tr>
<td>PG(80)</td>
<td>FLMCOL</td>
<td></td>
<td>array containing hot-gas-side static pressure, lbf/in², at each station</td>
</tr>
<tr>
<td>PI</td>
<td>HCOOL</td>
<td></td>
<td>constant, 3.14159</td>
</tr>
<tr>
<td>PIM</td>
<td>TCO</td>
<td></td>
<td>impingement-supply pressure, lbf/in²</td>
</tr>
<tr>
<td>PIMOLD(15)</td>
<td>TCOEF</td>
<td></td>
<td>saved value of impingement-supply pressure, lbf/in², for each slice</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------------</td>
<td>------------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PIN(15)</td>
<td>RADL</td>
<td></td>
<td>coolant total pressure, lbf/in², at entrance to each slice</td>
</tr>
<tr>
<td>PINS</td>
<td>HCPINS</td>
<td>TARRAY</td>
<td>number of pin fins at given station</td>
</tr>
<tr>
<td>PIVOT</td>
<td>GAUSS</td>
<td></td>
<td>main diagonal term of row of matrix being solved</td>
</tr>
<tr>
<td>PLEGN(5)</td>
<td>PLOTMF</td>
<td></td>
<td>alphanemic array to label pressure-data plots</td>
</tr>
<tr>
<td>PLTYME(2)</td>
<td>PLOTMF</td>
<td></td>
<td>alphanemic variable to print transient time on each plot</td>
</tr>
<tr>
<td>POLD(15,80)</td>
<td>TCOEF</td>
<td></td>
<td>saved values of coolant-channel pressure, lbf/in², from previous iteration</td>
</tr>
<tr>
<td>PP</td>
<td>PLNUM</td>
<td></td>
<td>intermediate pressure term</td>
</tr>
<tr>
<td>PPLEN</td>
<td>NTTACT</td>
<td></td>
<td>impingement-supply pressure, lbf/in²</td>
</tr>
<tr>
<td>PROD</td>
<td>HCOOL</td>
<td></td>
<td>intermediate calculation result</td>
</tr>
<tr>
<td>PROPS</td>
<td>GETIN</td>
<td></td>
<td>NAMELIST name</td>
</tr>
<tr>
<td>PSAV(5)</td>
<td>TCOEF</td>
<td></td>
<td>array to save last four values of pressure at flow-split station, used to check convergence</td>
</tr>
<tr>
<td>PTEMP</td>
<td>PLNUM</td>
<td></td>
<td>intermediate pressure term</td>
</tr>
<tr>
<td>PTEXIT</td>
<td>PLNUM</td>
<td></td>
<td>coolant-plenum total pressure, lbf/in²: entrance value going into subroutine, exit value coming out</td>
</tr>
<tr>
<td>PTIN</td>
<td>NTTACT</td>
<td></td>
<td>coolant-supply pressure for a given time, lbf/in²</td>
</tr>
<tr>
<td>PTIO(50)</td>
<td>BOUND</td>
<td></td>
<td>input array of coolant-supply pressure, kPa (lbf/in²), as function of time, sec</td>
</tr>
<tr>
<td>PTIOC</td>
<td>INPRT</td>
<td></td>
<td>initial coolant-supply pressure in SI units, kPa</td>
</tr>
<tr>
<td>PTNOLD</td>
<td>NTTACT</td>
<td></td>
<td>previous value of PTIN, lbf/in²</td>
</tr>
<tr>
<td>PT1</td>
<td>PLNUM</td>
<td></td>
<td>calculated coolant-plenum inlet total pressure, lbf/in²</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>PUMP(80)</td>
<td>TCO</td>
<td></td>
<td>term to account for coolant pumping due to wheel rotation</td>
</tr>
<tr>
<td>PUMTRM</td>
<td></td>
<td>PARRAY</td>
<td>term to account for coolant pumping due to wheel rotation</td>
</tr>
<tr>
<td>PXX</td>
<td></td>
<td>PLNUM</td>
<td>average static pressure, lbf/in$^2$, in coolant plenum for given slice</td>
</tr>
<tr>
<td>QG(80)</td>
<td>TCO</td>
<td></td>
<td>hot-gas heat flux to blade at each station, Btu/hr $\cdot$ ft$^2$</td>
</tr>
<tr>
<td>QHUB(80)</td>
<td>BOUND</td>
<td></td>
<td>heat flux conducted to blade wall from hub platform at each station, Btu/hr $\cdot$ ft$^2$</td>
</tr>
<tr>
<td>QHUBIN(400)</td>
<td>BOUND</td>
<td></td>
<td>input table of hub heat flux at each station as function of time, W/m$^2$ (Btu/hr $\cdot$ ft$^2$)</td>
</tr>
<tr>
<td>QSNK(80)</td>
<td>TCO</td>
<td></td>
<td>term to account for heat removal from wall by film-cooling flow through wall</td>
</tr>
<tr>
<td>QTIP(80)</td>
<td>BOUND</td>
<td></td>
<td>heat flux from blade wall at tip for each station, Btu/hr $\cdot$ ft$^2$</td>
</tr>
<tr>
<td>QTIPIN(400)</td>
<td>BOUND</td>
<td></td>
<td>input table of tip heat flux at each station as function of time, W/m$^2$ (Btu/hr $\cdot$ ft$^2$)</td>
</tr>
<tr>
<td>R</td>
<td>TCO</td>
<td></td>
<td>gas constant; value for air is built in, 53.35 ft-lbf/lbm $\cdot ^{0}$R</td>
</tr>
<tr>
<td>RATIO</td>
<td></td>
<td>THRCON</td>
<td>interpolating fraction</td>
</tr>
<tr>
<td>RCHRD</td>
<td></td>
<td>TARRAY</td>
<td>dimensionless ratio of time increment to chordwise length increment squared at each station</td>
</tr>
<tr>
<td>RCHRDM</td>
<td></td>
<td>TARRAY</td>
<td>maximum value of RCHRD for a given slice</td>
</tr>
<tr>
<td>RCVRY</td>
<td></td>
<td>TARRAY</td>
<td>recovery factor</td>
</tr>
<tr>
<td>RE(80)</td>
<td></td>
<td>PRPS</td>
<td>coolant-channel Reynolds number at each station</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>-------------</td>
<td>----------</td>
<td>------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>REFC(80)</td>
<td>FLMCOL</td>
<td></td>
<td>film-cooling flow Reynolds number at each station</td>
</tr>
<tr>
<td>REJ(80)</td>
<td>HCOOL</td>
<td></td>
<td>impingement-jet Reynolds number at each station</td>
</tr>
<tr>
<td>REJOVR(80)</td>
<td>HCOOL</td>
<td></td>
<td>array to save values of impingement-jet Reynolds number that are out of range of correlation</td>
</tr>
<tr>
<td>REY</td>
<td>PLNUM</td>
<td></td>
<td>coolant-plenum Reynolds number based on hydraulic diameter</td>
</tr>
<tr>
<td>RHOBAR</td>
<td>TARRAY</td>
<td></td>
<td>mean density in coolant channel, lbm/in³</td>
</tr>
<tr>
<td>RHOC</td>
<td>TRNSNT</td>
<td></td>
<td>input density of outer coating, kg/m³ (lbm/ft³)</td>
</tr>
<tr>
<td>RHOM</td>
<td>TRNSNT</td>
<td></td>
<td>input density of wall metal, kg/m³ (lbm/ft³)</td>
</tr>
<tr>
<td>RHOVG(400)</td>
<td>BOUND</td>
<td></td>
<td>input table of hot-gas-side, free-stream mass velocity at each station as function of time, kg/m²·sec (lbm/ft²·sec)</td>
</tr>
<tr>
<td>RHOVGA(80)</td>
<td>FLMCOL</td>
<td></td>
<td>hot-gas-side, free-stream mass velocity at each station for given slice, lbm/ft²·sec</td>
</tr>
<tr>
<td>RI</td>
<td>GETIN</td>
<td></td>
<td>input value of radial location of coolant-plenum inlet for given slice, cm (in.)</td>
</tr>
<tr>
<td>RIN(15)</td>
<td>RADL</td>
<td></td>
<td>table of RI values for each slice, in.</td>
</tr>
<tr>
<td>RO</td>
<td>GETIN</td>
<td></td>
<td>input value of radial location of coolant-plenum exit for given slice, cm (in.)</td>
</tr>
<tr>
<td>ROINVC</td>
<td>HCOOL</td>
<td></td>
<td>intermediate term in impingement correlation, ft³/lbm</td>
</tr>
<tr>
<td>ROINVJ</td>
<td>HCOOL</td>
<td></td>
<td>intermediate term in impingement correlation, ft³/lbm</td>
</tr>
<tr>
<td>ROOT</td>
<td>PARRAY</td>
<td></td>
<td>intermediate term in pressure calculations</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>-------------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>ROUT(15)</td>
<td>RADL</td>
<td></td>
<td>table of RO values for each slice, in.</td>
</tr>
<tr>
<td>RR(80)</td>
<td>TCO</td>
<td></td>
<td>mean radial location of each station for a given slice</td>
</tr>
<tr>
<td>RRP</td>
<td></td>
<td>PLNUM</td>
<td>radial location, in.</td>
</tr>
<tr>
<td>RTEMP</td>
<td></td>
<td>PLNUM</td>
<td>radial location, in.</td>
</tr>
<tr>
<td>RTNARR(2)</td>
<td></td>
<td>PLOTMF</td>
<td>array containing maximum and minimum values of plot variables</td>
</tr>
<tr>
<td>RTRNV</td>
<td></td>
<td>TARRAY</td>
<td>dimensionless ratio of time increment to through-the-wall length increment squared</td>
</tr>
<tr>
<td>RTRNVM</td>
<td></td>
<td>TARRAY</td>
<td>maximum value of RTRNV for given slice</td>
</tr>
<tr>
<td>S(15)</td>
<td>TCO</td>
<td></td>
<td>span of each slice, in.</td>
</tr>
<tr>
<td>SEGMTS</td>
<td></td>
<td>PLNUM</td>
<td>number of segments in coolant plenum for given slice</td>
</tr>
<tr>
<td>SIGB</td>
<td></td>
<td>PLNUM</td>
<td>dummy variable used in coolant-plenum calculations</td>
</tr>
<tr>
<td>SIGC</td>
<td></td>
<td>PLNUM</td>
<td>dummy variable used in coolant-plenum calculations</td>
</tr>
<tr>
<td>SIGMA(20)</td>
<td></td>
<td>PLNUM</td>
<td>coolant velocity distribution in coolant plenum</td>
</tr>
<tr>
<td>SLEGN(5)</td>
<td></td>
<td>PLOTMF</td>
<td>alphameric array to label suction-side plots</td>
</tr>
<tr>
<td>SLP</td>
<td></td>
<td>HCPINS</td>
<td>mean pin-fin length at given station, in.</td>
</tr>
<tr>
<td>SP(80)</td>
<td>PRPS</td>
<td></td>
<td>pin-fin spacing at each station, in.</td>
</tr>
<tr>
<td>SPAN</td>
<td>TCO</td>
<td></td>
<td>radial span of given slice, in.</td>
</tr>
<tr>
<td>SPANC</td>
<td></td>
<td>INPRT</td>
<td>radial span of given slice in SI units, cm</td>
</tr>
<tr>
<td>SPHTC</td>
<td></td>
<td>TRNSNT</td>
<td>input specific heat of outer coating, J/kg · K (Btu/lbm · °F)</td>
</tr>
<tr>
<td>SPHTM</td>
<td></td>
<td>TRNSNT</td>
<td>input specific heat of wall metal, J/kg · K (Btu/lbm · °F)</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>--------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ST</td>
<td>HCOOL</td>
<td></td>
<td>Stanton number calculated from user-supplied impingement correlation</td>
</tr>
<tr>
<td>STANMX</td>
<td>HCOOL</td>
<td></td>
<td>Stanton number calculated from leading-edge impingement correlation</td>
</tr>
<tr>
<td>SV(3)</td>
<td>PLNUM</td>
<td></td>
<td>array to save values of SIGC</td>
</tr>
<tr>
<td>SYMBOL(10)</td>
<td>PLOTMF</td>
<td></td>
<td>array of integers to be used as plot symbols</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>PLOTMF</td>
<td></td>
<td>particular entry from SYMBL array</td>
</tr>
<tr>
<td>T(2,15,400)</td>
<td>TCO</td>
<td></td>
<td>calculated temperature at each node for each slice for two time steps, °F</td>
</tr>
<tr>
<td>TABOVE</td>
<td>TARRAY</td>
<td></td>
<td>midwall temperature at given station in slice above current slice, °F</td>
</tr>
<tr>
<td>TAU(400)</td>
<td>TCO</td>
<td></td>
<td>array of thickness values, in.</td>
</tr>
<tr>
<td>TBAR</td>
<td>FLOWS</td>
<td></td>
<td>coolant temperature, °R, used to calculate Mach number</td>
</tr>
<tr>
<td>TBAR</td>
<td>HCPINS</td>
<td></td>
<td>ratio of temperature drops in pin fins, pressure-side wall temperature minus</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>mid-coolant-channel temperature to suction-side wall temperature minus</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>mid-coolant-channel temperature</td>
</tr>
<tr>
<td>TBAR</td>
<td>WROUT</td>
<td></td>
<td>mean outer-surface temperature for given slice, °F</td>
</tr>
<tr>
<td>TBARMD</td>
<td>WROUT</td>
<td></td>
<td>mean midwall temperature, °F</td>
</tr>
<tr>
<td>TBELOW</td>
<td>TARRAY</td>
<td></td>
<td>midwall temperature at given station in slice below current slice, °F</td>
</tr>
<tr>
<td>TBULK</td>
<td>WROUT</td>
<td></td>
<td>overall blade bulk-metal temperature, °F</td>
</tr>
<tr>
<td>TC</td>
<td>THRCON</td>
<td></td>
<td>mean temperature of blade cladding material, °F</td>
</tr>
<tr>
<td>TCIN</td>
<td>NTTACT</td>
<td></td>
<td>coolant temperature, °F</td>
</tr>
<tr>
<td>TCOF(400,30)</td>
<td>MATRIX</td>
<td></td>
<td>array of coefficients to be solved for temperature or pressure</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TDHF</td>
<td>GETIN</td>
<td></td>
<td>function of film-cooling-hole size and spacing</td>
</tr>
<tr>
<td>TDHJ</td>
<td>GETIN</td>
<td></td>
<td>hydraulic diameter of impingement hole</td>
</tr>
<tr>
<td>TDLX(5)</td>
<td>GETIN</td>
<td></td>
<td>array containing lengths from nodes at upstream station to corresponding nodes at current station</td>
</tr>
<tr>
<td>TDP</td>
<td>GETIN</td>
<td></td>
<td>input pin-fin diameter, cm (in.)</td>
</tr>
<tr>
<td>TEM</td>
<td>INPRT</td>
<td></td>
<td>coolant-inlet absolute temperature, K (°R)</td>
</tr>
<tr>
<td>TEPS</td>
<td>TRNSNT</td>
<td></td>
<td>factor used to define time mean properties</td>
</tr>
<tr>
<td>TERM</td>
<td>FLSPLT</td>
<td></td>
<td>adjustment to flow-split parameter</td>
</tr>
<tr>
<td>TG(80)</td>
<td>TCO</td>
<td></td>
<td>hot-gas reference temperature at each station for given slice, °R</td>
</tr>
<tr>
<td>THETA1</td>
<td>TARRAY</td>
<td></td>
<td>intermediate terms in nodal energy equations</td>
</tr>
<tr>
<td>THETA2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>THETA3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>THETA4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>THETA5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>THETA6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>THETA8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>THETA9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>THK(3)</td>
<td>GETIN</td>
<td></td>
<td>input array of thickness values, cm (in.)</td>
</tr>
<tr>
<td>THUB(80)</td>
<td>BOUND</td>
<td></td>
<td>specified temperature at each blade hub station for given time, °F</td>
</tr>
<tr>
<td>THUBIN(400)</td>
<td>BOUND</td>
<td></td>
<td>input table of hub temperatures at each station as function of time, K (°F)</td>
</tr>
<tr>
<td>TIKLE(30)</td>
<td>GETIN</td>
<td></td>
<td>blank alphameric array used to initialize title array</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>TIN(15)</td>
<td>RADL</td>
<td></td>
<td>inlet total temperature in coolant plenum for each slice, °F</td>
</tr>
<tr>
<td>TIP1</td>
<td>TARRAY</td>
<td></td>
<td>term in conduction equation to account for specified tip temperature</td>
</tr>
<tr>
<td>TIP3</td>
<td>TARRAY</td>
<td></td>
<td>term in conduction equation to account for specified tip heat flux</td>
</tr>
<tr>
<td>TITL</td>
<td>GETIN</td>
<td></td>
<td>NAMELIST name</td>
</tr>
<tr>
<td>TITLE(30)</td>
<td>SPECL</td>
<td></td>
<td>alphameric array of title information from input</td>
</tr>
<tr>
<td>TLABL1(21)</td>
<td>PLOTMF</td>
<td></td>
<td>alphameric arrays used to put input title on plots</td>
</tr>
<tr>
<td>TLABL2(9)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TMAXXP</td>
<td>PLOTMF</td>
<td></td>
<td>maximum pressure-side temperature to be plotted</td>
</tr>
<tr>
<td>TMAXXS</td>
<td>PLOTMF</td>
<td></td>
<td>maximum suction-side temperature to be plotted</td>
</tr>
<tr>
<td>TMFRAC</td>
<td>PREP</td>
<td></td>
<td>time-interpolating function</td>
</tr>
<tr>
<td>TMINP</td>
<td>PLOTMF</td>
<td></td>
<td>minimum pressure-side temperature to be plotted</td>
</tr>
<tr>
<td>TMINS</td>
<td>PLOTMF</td>
<td></td>
<td>minimum suction-side temperature to be plotted</td>
</tr>
<tr>
<td>TMP</td>
<td>GASTBL</td>
<td></td>
<td>temperature used to determine gas properties, °R</td>
</tr>
<tr>
<td>TMP1</td>
<td>GASTBL</td>
<td></td>
<td>temperature used to determine gas properties, °F</td>
</tr>
<tr>
<td>TOTSPN</td>
<td>WROUT</td>
<td></td>
<td>total span of blade, in.</td>
</tr>
<tr>
<td>TP</td>
<td>PLNUM</td>
<td></td>
<td>intermediate temperature variable</td>
</tr>
<tr>
<td>TPLEN</td>
<td>NTTACT</td>
<td></td>
<td>mean impingement-plenum static temperature, °F</td>
</tr>
<tr>
<td>TPM(500)</td>
<td>PLOTMF</td>
<td></td>
<td>table of pressure-side, midwall temperatures for summary plots, K (°F)</td>
</tr>
<tr>
<td>TPMAX</td>
<td>WROUT</td>
<td></td>
<td>maximum pressure-side, outer-surface temperature for given slice, K (°F)</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TMIN</td>
<td>WROUT</td>
<td></td>
<td>minimum pressure-side, outer-surface temperature for given slice, K (°F)</td>
</tr>
<tr>
<td>TPO(500)</td>
<td>PLOTMF</td>
<td></td>
<td>table of pressure-side, outer-surface temperatures for summary plots, K (°F)</td>
</tr>
<tr>
<td>TREDGE</td>
<td>TARRAY</td>
<td></td>
<td>intermediate term in nodal energy equation in trailing-edge region</td>
</tr>
<tr>
<td>TREPS</td>
<td>PARRAY</td>
<td></td>
<td>same as TEPS</td>
</tr>
<tr>
<td>TRR</td>
<td>GETIN</td>
<td></td>
<td>input mean radial location of given station, cm (in.)</td>
</tr>
<tr>
<td>TRTRM</td>
<td>PARRAY</td>
<td></td>
<td>intermediate transient term in pressure equations</td>
</tr>
<tr>
<td>TRTRMC</td>
<td>TARRAY</td>
<td></td>
<td>intermediate transient term involving outer coating</td>
</tr>
<tr>
<td>TRTRMG</td>
<td>TARRAY</td>
<td></td>
<td>intermediate transient term involving coolant</td>
</tr>
<tr>
<td>TRTRMJ</td>
<td>TARRAY</td>
<td></td>
<td>intermediate transient term for coolant channel at entrance to trailing edge</td>
</tr>
<tr>
<td>TSAVE</td>
<td>TCOEF</td>
<td></td>
<td>temperature in coolant channel, just upstream of exit</td>
</tr>
<tr>
<td>TSM(500)</td>
<td>PLOTMF</td>
<td></td>
<td>table of suction-side, midwall temperature for summary plots, K (°F)</td>
</tr>
<tr>
<td>TSMAX</td>
<td>WROUT</td>
<td></td>
<td>maximum suction-side, outer-surface temperature for given slice, K (°F)</td>
</tr>
<tr>
<td>TSMIN</td>
<td>WROUT</td>
<td></td>
<td>minimum suction-side, outer-surface temperature for given slice, K (°F)</td>
</tr>
<tr>
<td>TSO(500)</td>
<td>PLOTMF</td>
<td></td>
<td>table of suction-side, outer-surface temperatures for summary plots, K (°F)</td>
</tr>
<tr>
<td>TSP</td>
<td>GETIN</td>
<td></td>
<td>input pin-fin spacing, cm (in.)</td>
</tr>
<tr>
<td>TTEMP</td>
<td>PLNUM</td>
<td></td>
<td>variable to save temperature value, °R</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TTEXIT</td>
<td>PLNUM</td>
<td></td>
<td>coolant total temperature in plenum, °F</td>
</tr>
<tr>
<td>TTIN</td>
<td>NTTACT</td>
<td></td>
<td>coolant-supply temperature, °F, at entrance to coolant plenum for given time</td>
</tr>
<tr>
<td>TTIO(50)</td>
<td>BOUND</td>
<td></td>
<td>input table of coolant-supply temperature, K (°F), as function of time</td>
</tr>
<tr>
<td>TTIP(80)</td>
<td>BOUND</td>
<td></td>
<td>table of blade tip temperature, °F, for given time</td>
</tr>
<tr>
<td>TTIPIN(400)</td>
<td>BOUND</td>
<td></td>
<td>input table of blade tip temperature, K (°F), as function of time</td>
</tr>
<tr>
<td>TTOTC(80)</td>
<td>TCOEF</td>
<td></td>
<td>coolant total temperature at each station for given slice, °F</td>
</tr>
<tr>
<td>TTX</td>
<td>PLNUM</td>
<td></td>
<td>coolant total temperature at inlet to coolant plenum for given slice, °R</td>
</tr>
<tr>
<td>TTYME</td>
<td>WROUT</td>
<td></td>
<td>current time in transient, sec</td>
</tr>
<tr>
<td>TT1(20)</td>
<td>PLNUM</td>
<td></td>
<td>total-temperature distribution, °F, in coolant plenum</td>
</tr>
<tr>
<td>TW</td>
<td>THRCON</td>
<td></td>
<td>midwall temperature for evaluating thermal conductivity</td>
</tr>
<tr>
<td>TXN</td>
<td>GETIN</td>
<td></td>
<td>input spanwise spacing of impingement jets at given station, cm (in.)</td>
</tr>
<tr>
<td>TXX</td>
<td>PLNUM</td>
<td></td>
<td>average static temperature in coolant plenum for given slice, °F</td>
</tr>
<tr>
<td>TYME</td>
<td>TRNSNT</td>
<td></td>
<td>time in transient calculations, sec</td>
</tr>
<tr>
<td>TYMMAX</td>
<td>TRNSNT</td>
<td></td>
<td>maximum time to which transient is carried, sec</td>
</tr>
<tr>
<td>T0G</td>
<td>TCO</td>
<td></td>
<td>impingement-jet temperature for given slice, °R</td>
</tr>
<tr>
<td>T1</td>
<td>PLNUM</td>
<td></td>
<td>inlet total temperature in coolant plenum for given slice, °R</td>
</tr>
<tr>
<td>UA(2)</td>
<td>INPRT</td>
<td></td>
<td>alphanemic array containing area units</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
<td>------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UL(2)</td>
<td>INPRT</td>
<td></td>
<td>alphameric array containing length units</td>
</tr>
<tr>
<td>V</td>
<td>TCOEF</td>
<td></td>
<td>factor to accelerate convergence of pressure iterations</td>
</tr>
<tr>
<td>VARIB(15)</td>
<td>PLOTMF</td>
<td></td>
<td>alphameric array containing some plot labels</td>
</tr>
<tr>
<td>VARS(12)</td>
<td>PLOTMF</td>
<td></td>
<td>array containing plotting controls</td>
</tr>
<tr>
<td>VDP</td>
<td>HCPINS</td>
<td></td>
<td>pin-fin diameter at given station, in.</td>
</tr>
<tr>
<td>VOLBAR</td>
<td>TARRAY</td>
<td></td>
<td>coolant-channel volume element at entrance to trailing-edge region, in^3</td>
</tr>
<tr>
<td>VSP</td>
<td>HCPINS</td>
<td></td>
<td>pin-fin spacing at given station, in.</td>
</tr>
<tr>
<td>V1</td>
<td>PLNUM</td>
<td></td>
<td>intermediate term in coolant-plenum calculations</td>
</tr>
<tr>
<td>W(15)</td>
<td>RADL</td>
<td></td>
<td>coolant flow-rate at entrance to each slice, lbm/hr</td>
</tr>
<tr>
<td>WC</td>
<td>HCOOL</td>
<td></td>
<td>absolute value of coolant flow-rate, lbm/sec</td>
</tr>
<tr>
<td>WCHK(80)</td>
<td>CHKHOL</td>
<td></td>
<td>alphameric variable used to indicate choked flow in impingement jets</td>
</tr>
<tr>
<td>WCHKDM</td>
<td>CHKHOL</td>
<td></td>
<td>alphameric variable used to indicate choked flow in holes dumping coolant to trailing-edge region</td>
</tr>
<tr>
<td>WCHOKE</td>
<td>PLNUM</td>
<td></td>
<td>coolant flow-rate at entrance to given slice, lbm/hr</td>
</tr>
<tr>
<td>WCR</td>
<td>FLOWS</td>
<td></td>
<td>critical flow-rate, lbm/sec</td>
</tr>
<tr>
<td>WCROS(2, 15, 80)</td>
<td>TCO</td>
<td></td>
<td>coolant-channel crossflow rate at each station for each slice for two time steps, lbm/sec</td>
</tr>
<tr>
<td>WDUMP</td>
<td>TCO</td>
<td></td>
<td>rate of coolant flow being dumped directly from plenum to trailing-edge region, lbm/sec</td>
</tr>
<tr>
<td>WFC(80)</td>
<td>TCO</td>
<td></td>
<td>film-cooling flow rate at each station for given slice, lbm/sec</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------------</td>
<td>--------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>WFCDUM</td>
<td>FLOWS</td>
<td></td>
<td>intermediate variable in film-cooling flow rate calculation</td>
</tr>
<tr>
<td>WFCDUM</td>
<td>PARRAY</td>
<td></td>
<td>total film-cooling flow rate at given station, lbm/sec</td>
</tr>
<tr>
<td>WIM</td>
<td>TCO</td>
<td></td>
<td>total impingement-jet flow rate for given slice, lbm/sec</td>
</tr>
<tr>
<td>WJ(15,80)</td>
<td>TCO</td>
<td></td>
<td>impingement-jet flow rate for each station for each slice, lbm/sec</td>
</tr>
<tr>
<td>WPLEN</td>
<td>BOUND</td>
<td></td>
<td>input initial guess at total coolant flow, kg/hr (lbm/hr)</td>
</tr>
<tr>
<td>WPLENC</td>
<td>INPRT</td>
<td></td>
<td>total coolant flow in SI units, kg/hr</td>
</tr>
<tr>
<td>WPLENO</td>
<td>NTTACT</td>
<td></td>
<td>variable to save previous estimate of total coolant flow, lbm/hr</td>
</tr>
<tr>
<td>WS</td>
<td>RADL</td>
<td></td>
<td>rotor speed at given time, rpm</td>
</tr>
<tr>
<td>WSVST(50)</td>
<td>BOUND</td>
<td></td>
<td>input table of rotor speed, rpm, versus time, sec</td>
</tr>
<tr>
<td>WUSED</td>
<td>NTTACT</td>
<td></td>
<td>cumulative amount of coolant used, up to current slice, lbm/hr</td>
</tr>
<tr>
<td>WXCP</td>
<td>TARRAY</td>
<td></td>
<td>coolant flow-rate times specific-heat term</td>
</tr>
<tr>
<td>WXX</td>
<td>PLNUM</td>
<td></td>
<td>coolant flow-rate at entrance to given slice, lbm/hr</td>
</tr>
<tr>
<td>X(80)</td>
<td>TCOEF</td>
<td></td>
<td>coolant-channel node locations, in., used to set initial pressure distribution</td>
</tr>
<tr>
<td>XBAR</td>
<td>FLOWS</td>
<td></td>
<td>term in film-cooling effectiveness correlation</td>
</tr>
<tr>
<td>XCC</td>
<td>INPRT</td>
<td></td>
<td>coolant-channel distance from station 1 to given station, cm (in.)</td>
</tr>
<tr>
<td>XDUM</td>
<td>FLOWS</td>
<td></td>
<td>dummy variable used to save location of film-cooling hole</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>--------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>XFC(80)</td>
<td>FLMCOL</td>
<td></td>
<td>distance from station with film-cooling holes to downstream stations without film-cooling holes, in.</td>
</tr>
<tr>
<td>XIS</td>
<td></td>
<td>INPRT</td>
<td>inside-wall surface distance from station 1 to given station</td>
</tr>
<tr>
<td>XJN</td>
<td></td>
<td>INPRT</td>
<td>distance along junction of cladding and wall metal from station 1 to given station, cm (in.)</td>
</tr>
<tr>
<td>XK(4)</td>
<td></td>
<td>PLNUM</td>
<td>pressure change in coolant plenum</td>
</tr>
<tr>
<td>XL</td>
<td></td>
<td>HCOOL</td>
<td>length of inner surface used in leading-edge impingement correlation, in.</td>
</tr>
<tr>
<td>XL(4)</td>
<td></td>
<td>PLNUM</td>
<td>temperature change in coolant plenum</td>
</tr>
<tr>
<td>XLABL(29)</td>
<td></td>
<td>PLOTMF</td>
<td>alphanumerical array of plot labels</td>
</tr>
<tr>
<td>XLABL2(15)</td>
<td></td>
<td>PLOTMF</td>
<td>alphanumerical array of plot labels</td>
</tr>
<tr>
<td>XLBL(20)</td>
<td></td>
<td>PLOTMF</td>
<td>array of x-coordinates to be plotted as slice numbers</td>
</tr>
<tr>
<td>XMM</td>
<td></td>
<td>INPRT</td>
<td>midwall distance from station 1 to given station, cm (in.)</td>
</tr>
<tr>
<td>XMU</td>
<td>FLOWS</td>
<td></td>
<td>viscosity, lbm/ft⋅hr</td>
</tr>
<tr>
<td></td>
<td>GASTBL</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HCFRCD</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HCOOL</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HCPINS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PLNUM</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TARRAY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XMUC(80)</td>
<td>FLMCOL</td>
<td></td>
<td>coolant viscosity, lbm/ft⋅hr, at each station, evaluated at mean temperature between inner-wall surface and bulk coolant temperatures</td>
</tr>
<tr>
<td>XMUM</td>
<td></td>
<td>FLOWS</td>
<td>hot-gas viscosity, lbm/ft⋅hr</td>
</tr>
<tr>
<td>XMUTOG</td>
<td></td>
<td>HCOOL</td>
<td>coolant viscosity based on coolant-supply temperature, lbm/ft⋅hr</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>XN(80)</td>
<td>TCO</td>
<td></td>
<td>spanwise spacing of impingement holes at each station, in.</td>
</tr>
<tr>
<td>XNN</td>
<td>PLNUM</td>
<td></td>
<td>factor for increasing number of stations in coolant plenum</td>
</tr>
<tr>
<td>XOD</td>
<td>PREP</td>
<td></td>
<td>ratio of impingement-hole spacing to hydraulic diameter, at given station</td>
</tr>
<tr>
<td>XOS</td>
<td>INPRT</td>
<td></td>
<td>wall outer-surface distance from station 1 to given station, cm (in.)</td>
</tr>
<tr>
<td>XOVERD</td>
<td>HCOOL</td>
<td></td>
<td>ratio of impingement-hole spacing to hydraulic diameter, at given station</td>
</tr>
<tr>
<td>XOVRDL</td>
<td>HCPINS</td>
<td></td>
<td>location of zero temperature gradient in pin fins</td>
</tr>
<tr>
<td>XP</td>
<td>HCOOL</td>
<td></td>
<td>length of pressure-side inner-wall surface in leading-edge impingement region</td>
</tr>
<tr>
<td>XP(80)</td>
<td>PLOTMF</td>
<td></td>
<td>pressure-side, dimensionless distance along midwall plane from station 1 to each station</td>
</tr>
<tr>
<td>XP</td>
<td>PREP</td>
<td></td>
<td>distance of given pressure-side station from station 1, in.</td>
</tr>
<tr>
<td>XPF</td>
<td>PREP</td>
<td></td>
<td>interpolating fraction in BCXP table</td>
</tr>
<tr>
<td>XPL</td>
<td>PLOTMF</td>
<td></td>
<td>overall length along pressure-side, midwall plane, cm (in.)</td>
</tr>
<tr>
<td>XS</td>
<td>HCOOL</td>
<td></td>
<td>length of suction-side, inner-wall surface in leading-edge impingement region</td>
</tr>
<tr>
<td>XS(80)</td>
<td>PLOTMF</td>
<td></td>
<td>suction-side, dimensionless distance along midwall plane from station 1 to each station</td>
</tr>
<tr>
<td>XS</td>
<td>PREP</td>
<td></td>
<td>distance of given suction-side station from station 1, in.</td>
</tr>
<tr>
<td>XSF</td>
<td>PREP</td>
<td></td>
<td>interpolating function in BCXS table</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>XSL</td>
<td>PLOTMF</td>
<td></td>
<td>overall length along suction-side, mid-wall plane, cm (in.)</td>
</tr>
<tr>
<td>XTEST</td>
<td>PLNUM</td>
<td>PLNUM</td>
<td>convergence test variable</td>
</tr>
<tr>
<td>XTOT</td>
<td>WROUT</td>
<td>WROUT</td>
<td>overall outer-surface length around blade, in.</td>
</tr>
<tr>
<td>XTOTMD</td>
<td>WROUT</td>
<td></td>
<td>overall midwall length around blade, in.</td>
</tr>
<tr>
<td>XXN</td>
<td>PLNUM</td>
<td></td>
<td>number of spanwise stations per slice in coolant plenum</td>
</tr>
<tr>
<td>Y(320)</td>
<td>PLOTMF</td>
<td></td>
<td>array containing temperature values to be plotted</td>
</tr>
<tr>
<td>YCNVUU</td>
<td>TARRAY</td>
<td></td>
<td>indicates forced-convection heat transfer at last forward-region station on pressure side</td>
</tr>
<tr>
<td>YCONV</td>
<td>TARRAY</td>
<td></td>
<td>indicates forced-convection heat transfer at given station</td>
</tr>
<tr>
<td>YCONVU</td>
<td>TARRAY</td>
<td></td>
<td>indicates forced-convection heat transfer at station immediately upstream of given station</td>
</tr>
<tr>
<td>YFINS</td>
<td>TARRAY</td>
<td></td>
<td>indicates pin-fin heat transfer at given station</td>
</tr>
<tr>
<td>YFINSU</td>
<td>TARRAY</td>
<td></td>
<td>indicates pin-fin heat transfer at station immediately upstream of given station</td>
</tr>
<tr>
<td>YFNSUU</td>
<td>TARRAY</td>
<td></td>
<td>indicates pin-fin heat transfer at last forward-region station on pressure side</td>
</tr>
<tr>
<td>YIMP</td>
<td>TARRAY</td>
<td></td>
<td>indicates impingement heat transfer at given station</td>
</tr>
<tr>
<td>YIMPU</td>
<td>TARRAY</td>
<td></td>
<td>indicates impingement heat transfer at station immediately upstream of given station</td>
</tr>
<tr>
<td>YIMPUU</td>
<td>TARRAY</td>
<td></td>
<td>indicates impingement heat transfer at last forward-region station on pressure side</td>
</tr>
<tr>
<td>Variable</td>
<td>Common</td>
<td>Subroutine</td>
<td>Definition</td>
</tr>
<tr>
<td>---------------</td>
<td>--------</td>
<td>------------</td>
<td>-----------------------------------------------------</td>
</tr>
<tr>
<td>YLABL(7)</td>
<td></td>
<td>PLOTMF</td>
<td>alphameric array for labeling plots</td>
</tr>
<tr>
<td>YLABL2(11)</td>
<td></td>
<td>PLOTMF</td>
<td>alphameric array for labeling plots</td>
</tr>
<tr>
<td>YLBL (20)</td>
<td></td>
<td>PLOTMF</td>
<td>array of coordinates of points to be plotted as slice numbers</td>
</tr>
<tr>
<td>YMAX</td>
<td></td>
<td>PLOTMF</td>
<td>maximum value of y-coordinates on plot</td>
</tr>
<tr>
<td>YMIN</td>
<td></td>
<td>PLOTMF</td>
<td>minimum value of y-coordinates on plot</td>
</tr>
<tr>
<td>YPLABL(10)</td>
<td></td>
<td>PLOTMF</td>
<td>alphameric array for labeling plots</td>
</tr>
<tr>
<td>YTEM(80)</td>
<td></td>
<td>PLOTMF</td>
<td>array to be plotted</td>
</tr>
<tr>
<td>ZED</td>
<td></td>
<td>PLNUM</td>
<td>coolant-plenum pressure-drop parameter for given slice</td>
</tr>
<tr>
<td>ZOVERD</td>
<td></td>
<td>HCOOL</td>
<td>ratio of coolant-channel width to impingement-hole hydraulic diameter</td>
</tr>
<tr>
<td>Z1(15)</td>
<td></td>
<td>PLNUM</td>
<td>coolant-plenum pressure-drop parameter for each slice</td>
</tr>
<tr>
<td>Z3</td>
<td></td>
<td>PLNUM</td>
<td>intermediate term involving coolant flow</td>
</tr>
<tr>
<td>Z4</td>
<td></td>
<td>PLNUM</td>
<td>intermediate term involving coolant flow</td>
</tr>
</tbody>
</table>
C----SOURCE.NTACT---THIS IS THE MAIN PROGRAM. BLOCK DATA SUBPROGRAM NTTACT--0001

GASDAT MUST BE LOADED FIRST. NTTACT--0002

TRANSIENT THERMAL ANALYSIS OF A COOLED TURBINE BLADE NTTACT--0003

* * * NTTACT--0004

TACT 1 NTTACT--0005

COMMON /BOUND/ BCXS(400), BCXP(400), BCHGS(1000), BCHGP(1000), NTTACT--0006

Z BCTGS(1000), BCPG(1000), BCPQS(1000), BCPGP(1000), NTTACT--0007

Z THUBIN(400), THUB(80), NTTACT--0008

Z QTHUPIN(400), QTP(80), ROVQG(400), PEG(400), NTTACT--0009

Z DSCTIME(50), TTIO(50), PTIO(50), WPLEN, NTTACT--0010

Z WSVST(50), AKCTBL(20), AKWTBL(20), NBCS, NBCP NTTACT--0011

COMMON /FLMCOL/ RHOGA(80), PG(80), XPC(80), FLMEDF(80), NTTACT--0012

Z XMUC(80), EMES(80), REPC(80), NFCSUP(80), NTTACT--0013

COMMON /GAAS/ GS(200), NG NTTACT--0014

COMMON /RACL/ APLN(15), DPLN(15), RIN(15), ROUT(15), NTTACT--0015

Z PIN(15), TIN(15), U(15), NTTACT--0016

COMMON /SPECL/ CHAN(8000), TITLE(30), INDCHN(2000), NTTACT--0017

Z IPLOT, 1D1, XD2, MD3, IADJIN, IWRIrE NTTACT--0018

COMMON /TCO/ ADUMP, BTA, CD, CP, NTTACT--0019

Z GAM, PIM, R, SFAN, TOG, NTTACT--0020

Z WDUMP, WIM, AKC(15,80), AKW(15,80), NTTACT--0021

Z A(400), AJET(80), AM2(80), CHUM(80), NTTACT--0022

Z DH(80), DHF(80), DHJ(80), NTTACT--0023

Z DLX(400), PX(80), HC(80), HG(80), NTTACT--0024

Z P(2,15,80), PXTL(15), PUMP(80), QG(80), NTTACT--0025

Z QSNK(80), BU(80), S(15), T(2,15,400), NTTACT--0026

Z TG(80), Tau(400), WFC(80), NTTACT--0027

Z WI(15,80), WCROS(2,15,80), XN(80), NTTACT--0028

Z ICOR, IFILM, IHUB, ITIP, NTTACT--0029

Z INSPLOK, ISLICE, NSLICE, NTTACT--0030

Z NFWD, NSTA, IHC(80), NTTACT--0031

COMMON /TRANSL/ RHOC, RHOM, SPHTC, SPHTM, NTTACT--0032

Z DLYMME, TMEP, TEP, TYMAX, NTTACT--0033

COMMON /UNITS/ CINCH(2), CHTC(2), CHFLX(2), CPRSR(2), CMSPLF(2), NTTACT--0034

Z CTMPF(2), CTCON(2), CDEN(2), CSPHT(2), CGASC(2), NTTACT--0035

Z CVSIC(2), CRHOVG(2), UNITS NTTACT--0036

DIMENSION DP(80), SP(80), ALPH(12), ALPH2(4), CD1(200) NTTACT--0037

TTO = TOTAL TEMPERATURE OF BLADE COOLING AIR AT INLET NTTACT--0038

WPEN = ESTIMATE OF COOLANT FLOW RATE - USED AS FIRST GUESS NTTACT--0039

PTIO = TOTAL PRESSURE OF BLADE COOLING AIR AT INLET NTTACT--0040

PEX = EXTERNAL GAS STREAM STATIC PRESSURE AT TRAILING EDGE NTTACT--0041

DATA ALPH/'TSJ', 'BWA', 'ST', 'ARTE', 'ID AT', NTTACT--0042

DATA NCHAR/16/ NTTACT--0043

MD1 = 0 NTTACT--0044

MD2 = 0 NTTACT--0045
TYME = -1.0
DLTYME = 0.0

C TO GET AN ABBREVIATED OUTPUT OF MID-WALL TEMPERATURES AT THE TERMINAL
C FOR EACH SLICE, ENTER:
C AT TACT=50; SET TACT.MD1=1
C
MD3 = 0
K = 1

C RECORD STARTING TIME, TO BE USED TO IDENTIFY MICROFILM PLOTS
C
CALL TIME(NCHAR, ALPH2)
ALPH(7) = ALPH2(3)
ALPH(8) = ALPH2(4)
ALPH(10) = ALPH2(1)
ALPH(11) = ALPH2(2)
WRITE(6,425) (ALPH(I),I=1,12)
WRITE(8,425) (ALPH(I),I=1,12)

C READ IN DATA
C
CALL GETIN(IWRITE,TYMAX,WSVST,IADJIN)

C WRITE TITLE PAGE
C
WRITE(6,400)
WRITE(6,425) (ALPH(I),I=1,12)
WRITE(6,430) (TITLZ(I),I=1,30)

400 FORMAT(1H1,//////,50X,'***** OUTPUT *****',/////)
425 FORMAT(/36X,12A4)
430 FORMAT(/1X,30A4)

C TTIN = TTI0(1)
PTIN = PTIO(1)
WELONO = WPLEN
PTNOLD = PTIN
PTIO(1) = PTIN
TYME = 0.0
NTYM = 1
IF (DLTYME.GT.0.) NTYM = TYMAX/DLTYME + 1
NODST = 5*NSTA

C START MARCHING
C
DO 1100 IYTM = 1,NTYM
IYTM = IYTM-1
NTIG = IYTM
TYME = IYTM*DLTYME
IF (IYTM.EQ.1) TYME = -1.

C-- EVALUATE TIME DEPENDENT BOUNDARY CONDITIONS --------------------------

C
PTIN = PTIO(1)
IF (TYME.LT.0.0) GO TO 490

C

54
C-- LOCATE COOLANT SUPPLY PRESSURE FOR TYME
C
DO 450 I = 4,50,2
PTIN = PTIO(I-3)
IF (PTIO(I).LE.0.0) GO TO 460
IPTIO = I-1
IF (TYME.LE.PTIO(I).AND.TYME.GT.PTIO(I-2)) GO TO 455
450 CONTINUE
455 DENOM = PTIO(IPTIO+1)-PTIO(IPTIO-1)
IF (DENOM.GT.0.) PTIN = PTIO(IPTIO-2) +
Z (PTIO(IPTIO)-PTIO(IPTIO-2))*(TYME-PTIO(IPTIO-1))/DENOM
460 CONTINUE
C
C-- LOCATE COOLANT SUPPLY TEMPERATURE FOR TYME
C
DO 470 I = 4,50,2
TTIN = TTIO(IO)-3
IF (TTIO(I).LE.0.0) GO TO 490
ITTIO = I-1
IF (TYME.LE.TTIO(I) .AND. TYHE.GT.TTIO(I-2)) GO TO 475
470 CONTINUE
475 DENOM = TTIO(ITTIO+1)-TTIO(ITTIO-1)
IF (DENOM.GT.0.) TTIN = TTIO(ITTIO-2) +
Z (TTIO(ITTIO)-TTIO(ITTIO-2))*(TYME-TTIO(ITTIO-1))/DENOM
490 CONTINUE
C
C LOCATE THE VALUE OF THE WHEEL SPEED FOR THE CURRENT TIME.
C
WS = WSVST(I)
IIF (TYME.LE.0.0) GO TO 530
DO 510 I = 4,50,2
WS = WSVST(I-3)
IF (WSVST(I).LE.0.0) GO TO 530
WS = I-1
IF (TYME.LE.WSVST(I).AND.TYME.GT.WSVST(I-2)) GO TO 520
510 CONTINUE
520 DENOM = WSVST(IWS+1)-WSVST(IWS-1)
IF (DENOM.GT.0.) WS = WSVST(IWS-2) +
Z (WSVST(IWS)-WSVST(IWS-2))*(TYME-WSVST(IWS-1))/DENOM
530 CONTINUE
C
C LOCATE THE VALUE FOR PEXIT, GAS STATIC PRESSURE AT EXIT OF BLADE,
C FOR THE CURRENT TIME AND ALL SLICES.
C
IF (TYME.GT.0.0) GO TO 533
DO 532 I = 1,NSLICE
PEXIT(I) = PEX(I)
IF (PEX(I).LE.0.0) PEXIT(I) = PEX(1)
532 CONTINUE
533 CONTINUE
IF (BCTIME(2).LE.0.0) GO TO 545
DO 535 I = 2,50
IPEX = I-1
IF (TYME.LE.BCTIME(I).AND.TYME.GT.BCTIME(I-1)) GO TO 540
535 CONTINUE
540 DENOM = BCTIME(IPEX+1) - BCTIME(IPEX)
IF (DENOM.GT.0.0) GO TO 545
TYMFRC = (TYME - BCTIME(IPEX))/DENOM
DO 542 I = 1,NSLICE
IC = (IPEX-1)*NSLICE +.I
PEXIT(I) = PEX(IC) + (PEX(IC+NSLICE)-PEX(IC))*TYMPRC
542 CONTINUE
IF (IPFILM.NE.2) GO TO 545
C
C-- SET INTERPOLATED VALUES OF FREE STREAM RHO*V FOR THIS TIME
C
DO 543 I = 1,NSTA
IRO = (IPEX-1)*NSTA + I
IRN = IPEX*NSTA + I
543 RHOVGA(I) = RHOVG(IRO) + TYMPRC*(RHOVG(IRN)-RHOVG(IRO))
C
C-- SET TIME INTERPOLATED VALUES OF QHUB OR THUB AND QTIP OR TTIP.
C
IF (BCTIME(2).LE.0.0) GO TO 555
DO 550 I = 1,NSTA
IQO = (IPEX-1)*NSTA + I
IQN = IPEX*NSTA + I
IF (IHUB.EQ.1) THUB(I) = THUBIN(IQO) +
Z TMYPRC*(THUBIN(IQN)-THUBIN(IQO))
Z IF (IHUB.EQ.3) QHUB(I) = QHUBIN(IQO) +
Z TMYPRC*(QHUBIN(IQN)-QHUBIN(IQO))
Z IF (ITIP.EQ.1) TTIP(I) = TTIPIN(IQO) +
Z TMYPRC*(TTIPIN(IQN)-TTIPIN(IQO))
Z IF (ITIP.EQ.3) QTIP(I) = QTIPIN(IQO) +
Z TMYPRC*(QTIPIN(IQN)-QTIPIN(IQO))
550 CONTINUE
GO TO 565
C
555 CONTINUE
DO 560 I = 1,NSTA
IF (IHUB.EQ.1) THUB(I) = THUBIN(I)
IF (IHUB.EQ.3) QHUB(I) = QHUBIN(I)
IF (ITIP.EQ.1) TTIP(I) = TTIPIN(I)
IF (ITIP.EQ.3) QTIP(I) = QTIPIN(I)
560 CONTINUE
C
556 TCIN = TTIN
IF (ITYM.GT.1) WPLEN = WUSED*PTIN/PTNOLD
570 WPLEN = WPLEN
PTNOLD = PTIN
WUSED = 0.0
C
C CALCULATE TEMPERATURE AND PressURES FOR EACH SLICE OF THE BLADE
C
DO 1000 I = 1,NSLICE
ISLICE = I
C
C FIRST DETERMINE IMPINGEMENT PLENUM CONDITIONS
C
CALL PNUM(WPLEN,PPLEN,PTIN,TPLEN,TCIN)
TOG = TPLEN + 460.
PIM = PPLEN
IF (IUNITS.EQ.1) GO TO 860
WRITE(6,800) I,PIM,TOG
800 FORMAT(1H2//10X,100('*')//30X,'THE IMPINGEMENT PLENUM CONDITIONS'//60X,'PIM = ',F7.2,
Z FOR SLICE NO.'//60X,'I2,' ARE: /*10X,'T0G = ',F7.2)
Z 'PSIA,'/60X,'TOG = ','F7.2,' DEG. B'//10X,100('**))
C IF (I.EQ.1) WRITE(6,850) WPLEN
850 FORMAT('H',//30X,'CENTRAL PLENUM FLOW IS ','F6.1,' LBM/HR',//)
C GO TO 890
C 860 CTOG = TOG/1.8
CPIM = PIM/CPRSR(1)
WRITE(6,'(E7.0)') CTOG, CPIM
D70 FORMAT('H2',//10X,100('**'),//30X,'THE IMPINGEMENT PLENUM CONDITIONS'//)
Z ' FOR SLICE NO.',I2, ' ARE: '/60X,'PIM = ','F7.2,' K
Z ' KPA, '/60X,'TOG = ','F7.2,' K '//lox, 100
C CONTINUE

C THEN COMPUTE TEMPERATURES AND PRESSURES
C CALL TREP(I,NTTG)
C CALL TCOEP(IWRITE,WS,K,IPILOT,ALPH2)
IF (IPILOT.GT.0) CALL PLOTMF(ALPH2)
IF (MD1.EQ.0) GO TO 975
C
C THIS BLOCK PRINTS SPECIAL CONDENSED OUTPUT TO THE TERMINAL IF MD1 > 0
C
IC1 = 0
DO 955 II = 1,NSTA,2
IC1 = IC1 + 1
NMW = 5*II - 4
955 CD1(IC1) = T(2,II,NMW) - 460.
DO 360 II = 2,NSTA,2
IC1 = IC1 + 1
NMW = 5*II-4
360 CD1(IC1) = T(2,II,NMW) - 460.
WRITE(8,962) I,K
C52 FORMAT('H',//8,962) I,K
C52 FORMAT('H',//12,'OVERALL FLOW LOOP 'I2,'Z
C52,' SURFACE TEMPERATURE, (F), STARTING AT LEADING EDGE'//)
INUM = NSTA/2 + 1
WRITE(8,964) (CD1(I),II=1,INUM)
560 FORMAT('H',//10(2X,F7.1))
INUM = INUM + 1
WRITE(8,966) CD1(I), (CD1(I),II=INUM,NSTA)
570 FORMAT('H',//8,966)
C CONTINUE
C CHECK HOW MUCH PLENUM FLOW IS LEFT
C
WUSED = WUSED + 3600.*WIM
EXCESW = WPLEN - 3600.*WIM
IF (EXCESW.GT.0..AND.I.LT.NS克莱) WPLEN = EXCESW
I' (EXCESW.LT.0..AND.I.LT.NS克莱) WRITE(6,985) I
985 FORMAT('H',//10(2X,'RAN OUT OF MASS FLOW IN BRANCH NO. ',I2)
1000 CONTINUE
IF (IUNITS.EQ.2) WRITE(6,1010) WUSED
1010 FORMAT(/30X,'AMOUNT OF COOLANT ACTUALLY USED AT THIS TIME'
      ' STEP IS ',F6.1,' LBM/HR')
      CWUSED = WUSED/CMSFL(1)
      IF (IUNITS.EQ.1) WRITE(6,1011) CWUSED
1011 FORMAT(/30X,'AMOUNT OF COOLANT ACTUALLY USED AT THIS TIME'
      ' STEP IS ',F6.1,' KG/HR')
C
DO 1040 I = 1,NSLICE
DO 1020 J = 1,NODST
1020 T(I,1,J) = T(2,I,J)
DO 1040 J = 1,NSTA
1040 P(I,ISLICE,J) = P(Z,ISLICE,J)
IF (TYME.GT.0.0) GO TO 1100
C
EXCESW = WPLENO - WUSED
IF (ABS(EXCESW).LT.0.01*WPLENO) GO TO 1100
IF (IADJIN.GT.0.0) GO TO 1050
C
--- ADJUST COOLANT FLOW AND RECALCULATE TEMPERATURES, ETC. FOR STEADY
STATE CASE OR TIME =0.0
C
WPLEN = WPLENO -.995*EXCESW
K = K + 1
PTIN = PTNOLD
TCIN = TTIN
GO TO 570
C
--- SPECIAL CASE, FOR IADJPT > 0, ADJUSTMENT IS ON PTIN
C
1050 PTIN = PEXIT(1) + (PTNOLD-PEXIT(1))*WPLEN/WUSED**1.6
WPLEN = WPLENO
TCIN = TTIN
K = K + 1
GO TO 570
C
1100 CONTINUE
C
IF (IUNITS.EQ.1) GO TO 3860
3000 WRITE(6,3500) K,WPLENO
3500 FORMAT(/,20X,80(' '),/,'LOOP(S) ON INITIAL COOLANT FLOW'
      ' WERE USED. FINAL VALUE IS ',F7.2,' LBM/HR')
      WRITE(6,3600) EXCESW
3600 FORMAT(5X,'RESIDUAL COOLING AIR FLOW IS ',F9.4,' LBM/HR',/
      20X,80('(''))
      WRITE(6,425) (ALPH(1),I=1,12)
58
C==--------------------------------------------------------------------

WRITE (6,425) (ALPH(I),I=1,12)

C CONTINUE
MD2 = 1
IF (NSLICE.GT.1.AND.IPLOT.GT.0) CALL PLOTMF(ALPH2)
WRITE (6,425) (ALPH(I),I=1,12)
STOP

END

C==--SOURCE.NFLOEST

SUBROUTINE FLOWS(JS,DELTAN,ICHOKE,AMCHOR)

C THIS SUBROUTINE COMPUTES THE FOLLOWING---
WJ, IMPINGEMENT JET FLOW RATES (LBM/SEC).
WCROS, THE CHANNEL CROSSFLOWS (LBM/SEC).
AM2, THE SQUARE OF THE CHANNEL MACH NUMBER.
WDUMP, A FLOW RATE DUMPED DIRECTLY FROM CENTRAL PLENUM INTO TRAILING EDGE CHANNEL (LBM/SEC).
WIM, THE TOTAL COOLANT FLOW USED FOR THIS SLICE (LBM/SEC).
WFC, FILM COOLING FLOW RATES (LBM/SEC).
FF, THE CHANNEL FRICTION FACTOR, EITHER FOR LAMINAR, TURBULENT, OR A PIN FIN ARRAY.
PLMEFF, A FILM COOLING EFFECTIVENESS.

C==============================================

COMMON /CHKHOL/ UCHK(80), WCHKDN
COMMON /FLMCOL/ RHOVGA(80), PG(80), XFC(80), PLMEFF(80), XMUC(80), EMSS(80), REFC(80), NFCSUP(80)
COMMON /FRIC/ ALPHA, BETA, DLLTA, EPS
COMMON /PRPS/ CPo, GAMO, DP(80), KSU(80), R2(80), CPC(80), GANC(80), DUMR1(80), DUMR2(80)
COMMON /TCO/ ADUMP, BTA, CD, CF, GAM, PIN, R, SPAN, TOG
COMMON /TFKNT/ RHOC, RHOM, SPHTC, SPHTM

DATA CHKD/*am/ , UNCHKD/*I/ 

DIMENSION DELTAN(15)

C==--SOURCE.NFLOEST

SUBROUTINE FLOWS(JS,DELTAN,ICHOKE,AMCHOR)

C THIS SUBROUTINE COMPUTES THE FOLLOWING---
WJ, IMPINGEMENT JET FLOW RATES (LBM/SEC).
WCROS, THE CHANNEL CROSSFLOWS (LBM/SEC).
AM2, THE SQUARE OF THE CHANNEL MACH NUMBER.
WDUMP, A FLOW RATE DUMPED DIRECTLY FROM CENTRAL PLENUM INTO TRAILING EDGE CHANNEL (LBM/SEC).
WIM, THE TOTAL COOLANT FLOW USED FOR THIS SLICE (LBM/SEC).
WFC, FILM COOLING FLOW RATES (LBM/SEC).
FF, THE CHANNEL FRICTION FACTOR, EITHER FOR LAMINAR, TURBULENT, OR A PIN FIN ARRAY.
PLMEFF, A FILM COOLING EFFECTIVENESS.

C==============================================

COMMON /CHKHOL/ WCHK(80), WCHKDM
COMMON /FLMCOL/ RHOVGA(80), PG(80), XFC(80), PLMEFF(80), XMUC(80), EMSS(80), REFC(80), NFCSUP(80)
COMMON /FRIC/ ALPHA, BETA, DLLTA, EPS
COMMON /PRPS/ CPo, GAMO, DP(80), KSU(80), R2(80), CPC(80), GANC(80), DUMR1(80), DUMR2(80)
COMMON /TCO/ ADUMP, BTA, CD, CF, GAM, PIN, R, SPAN, TOG
COMMON /TFKNT/ RHOC, RHOM, SPHTC, SPHTM

DATA CHKD/*am/ , UNCHKD/*I/ 

DIMENSION DELTAN(15)

C==--SOURCE.NFLOEST

SUBROUTINE FLOWS(JS,DELTAN,ICHOKE,AMCHOR)

C THIS SUBROUTINE COMPUTES THE FOLLOWING---
WJ, IMPINGEMENT JET FLOW RATES (LBM/SEC).
WCROS, THE CHANNEL CROSSFLOWS (LBM/SEC).
AM2, THE SQUARE OF THE CHANNEL MACH NUMBER.
WDUMP, A FLOW RATE DUMPED DIRECTLY FROM CENTRAL PLENUM INTO TRAILING EDGE CHANNEL (LBM/SEC).
WIM, THE TOTAL COOLANT FLOW USED FOR THIS SLICE (LBM/SEC).
WFC, FILM COOLING FLOW RATES (LBM/SEC).
FF, THE CHANNEL FRICTION FACTOR, EITHER FOR LAMINAR, TURBULENT, OR A PIN FIN ARRAY.
PLMEFF, A FILM COOLING EFFECTIVENESS.

C==============================================

COMMON /CHKHOL/ WCHK(80), WCHKDM
COMMON /FLMCOL/ RHOVGA(80), PG(80), XFC(80), PLMEFF(80), XMUC(80), EMSS(80), REFC(80), NFCSUP(80)
COMMON /FRIC/ ALPHA, BETA, DLLTA, EPS
COMMON /PRPS/ CPo, GAMO, DP(80), KSU(80), R2(80), CPC(80), GANC(80), DUMR1(80), DUMR2(80)
COMMON /TCO/ ADUMP, BTA, CD, CF, GAM, PIN, R, SPAN, TOG
COMMON /TFKNT/ RHOC, RHOM, SPHTC, SPHTM

DATA CHKD/*am/ , UNCHKD/*I/ 

DIMENSION DELTAN(15)
C FOLLOWING VARIABLES NEEDED TO CALCULATE FILM COOLING EFFECTIVENESS.
C THEY ARE TRANSMITTED THROUGH COMMON FLMCOL
C WHERE-
C PG IS GAS SIDE STATIC PRESSURE DISTRIBUTION; XFC IS THE
C DISTANCE A STATION IS DOWNSTREAM FROM THE NEAREST
C ROW OF FILM COOLING HOLES, (IN); FLMEFF IS THE CALCULATED
C FILM EFFECTIVENESS AT EACH STATION;
C XMUC IS COOLANT VISCOSITY BASED ON LOCAL COOLANT TEMPERATURE;
C EMES IS THE PRODUCT M*S, WHERE M IS THE BLOWING
C RATIO, AND S IS AN EQUIVALENT SLOT WIDTH; REFC IS THE FILM
C REYNOLDS NUMBER, BASED ON S; AND NFCSUP IDENTIFIES
C THE STATION NUMBER SUPPLYING FILM COOLING TO DOWNSTREAM STATIONS.

C
100 CONTINUE
C INITIALIZE HOLE CHOKING INDICATOR TO UNCHOKED
DO 101 I = 1,NSTA
101 WCHK(I) = UNCHKD
ICHOKE = 0
N = NSTA-1
C
C N = NODE NUMBER OF LAST FLOW CHANNEL NODE, AT EXIT OF TRAILING EDGE
C
C CALCULATE IMPELLING FLOWS, AND FORWARD REGION FILM COOLING FLOWS
C
TMP = T0G
CALL GASTBL(TMP,C,CP,GAM,PD,R,XMU)
GAMO = GAM
CP0 = CP
NODSF = 5*NFWD
C
PAVG = P(2,ISLICE,1)
WJ(ISLICE,1) = 0.0
IF (PAVG.GT.PIM) GO TO 120
WCR=CD*PAVG*AJet (1)/ (R*T0G) * SQRT (64.4*GAMO*R*T0G/(GAMO+1.)) * 
Z * ((PIM/PAVG)**((GAMO-1.0)/GAMO)) 
WJ(ISLICE,1)=PAVG/(R*T0G) * AJet (1) * CD* 
Z* SQRT (64.4*GAMO*R*T0G/(GAMO-1.)* (1.- (PAVG/PIM)**((GAMO-1.0)/GAMO)))) 
1 * (PIM/PAVG)**((GAMO-1.0)/GAMO)
WPC(1) = 0.0
IF (P(2,ISLICE,1),GT,PG(1)) WFC(1) = CD*.25*3.1415926*(DHF(1)**2) 
Z* SQRT (32.2*P(2,ISLICE,1)* (P(2,ISLICE,1)-PG(1))/ (R*T(2,ISLICE,5))) 
IF (WCR.LT.WJ(ISLICE,1)) WCHK(1) = CHKD 
IF (WCR.LT.WJ(ISLICE,1)) WJ(ISLICE,1) = WCR
C
120 PAVG = P(2,ISLICE,2)
WJ(ISLICE,2) = 0.0
IF (PAVG.GT.PIM) GO TO 130
WCR=CD*PAVG*AJet (2)/(R*T0G) * SQRT (64.4*GAMO*R*T0G/(GAMO+1.)) * 
Z * ((PIM/PAVG)**((GAMO-1.0)/GAMO)) 
WJ(ISLICE,2)=PAVG/(R*T0G) * AJet (2) * CD* 
Z* SQRT (64.4*GAMO*R*T0G/(GAMO-1.)* (1.- (PAVG/PIM)**((GAMO-1.0)/GAMO)))) 
Z * ((PIM/PAVG)**((GAMO-1.0)/GAMO)) 
WPC(2) = 0.0
IF (P(2,ISLICE,2),GT,PG(2)) WFC(2) = CD*.25*3.1415926*(DHF(2)**2) 
Z* SQRT (32.2*P(2,ISLICE,2)* (P(2,ISLICE,2)-PG(2)))/ 
(R*T(2,ISLICE,10))) 
IF (WCR.LT.WJ(ISLICE,2)) WCHK(2) = CHKD 
IF (WCR.LT.WJ(ISLICE,2)) WJ(ISLICE,2) = WCR
C
PAVG = P(2, ISLICE, 3)
WJ(ISLICE, 3) = 0.0
IF (PAVG .GT. PIH) GO TO 140
WCR = CD * PAVG * A JET (3) / (R * TOG) * SQRT (64.4 * GAMO * R * TOG / (GAMO - 1.)) * (1. - (PAVG / PIH) ** ((GAMO - 1.) / GAMO))
Z = (PIN / PAVG) ** ((GAMO - 1.0) / GAMO)
WJ(ISLICE, 3) = PAVG / (R * TOG) * A JET (3) * CD *
Z SQRT (64.4 * GAMO * R * TOG / (GAMO - 1.)) * (1. - (PAVG / PIH) ** ((GAMO - 1.) / GAMO))
Z * (PIN / PAVG) ** ((GAMO - 1.0) / GAMO)
WPC(3) = 0.0
IF (P(2, ISLICE, 3) .GT. PG(3)) YFC(3) = CD * .25 * 3.1415926 * (DHP(3)**2) * Z * SQRT (32.2 * P(2, ISLICE, 3) * (P(2, ISLICE, 3) - PG(3)) / R * T(2, ISLICE, 15))
IF (WCR .LT. WJ(ISLICE, 3)) YJ(ISLICE, 3) = WCR
CONTINUE

C
DO 150 I = 4, NFWD
PAVG = P(2, ISLICE, I)
WJ(ISLICE, I) = 0.0
IF (PAVG .GT. PIH) GO TO 150
WCR = CD * PAVG * A JET (I) / (R * TOG) * SQRT (64.4 * GAMO * R * TOG / (GAMO - 1.)) * (1. - (PAVG / PIH) ** ((GAMO - 1.) / GAMO))
Z = (PIN / PAVG) ** ((GAMO - 1.0) / GAMO)
WJ(ISLICE, I) = PAVG / (R * TOG) * A JET (I) * CD *
Z SQRT (64.4 * GAMO * R * TOG / (GAMO - 1.)) * (1. - (PAVG / PIH) ** ((GAMO - 1.) / GAMO))
Z * (PIN / PAVG) ** ((GAMO - 1.0) / GAMO)
WFC(I) = 0.0
IF (P(2, ISLICE, I) .GT. PG(I)) WFC(I) = CD * .25 * 3.1415926 * (DHP(I)**2) * Z * SQRT (32.2 * P(2, ISLICE, I) * (P(2, ISLICE, I) - PG(I)) / R * T(2, ISLICE, 5*I))
IF (WCR .LT. WJ(ISLICE, I)) YJ(ISLICE, I) = WCR
CONTINUE

C
C CALCULATE CROSSFLOW RATE AT EACH STATION
C
200 WCROS(2, ISLICE, JS) = 0.0
JDIS = JS / 2
C**** JDIS IS THE DISPLACEMENT OF THE FLOW SPLIT STATION FROM STATION 1
JSSENS = JS - 2 * JDIS
C********* JSSENS = 0, STATION IS ON SUCTION SIDE (EVEN);
C = 1, STATION IS ON PRESSURE SIDE (ODD);
C
JP = JS + 2
JM = JS - 2
IF (JM .LT. 1) JM = 1
JPIN = JS + 1
JSTART = JS + 3
IF (JSSENS.EQ.1) GO TO 220
IF (JSSENS.EQ.0) GO TO 230
IF (JSSENS.EQ.1) GO TO 270
C
CONTINUE
C
C**** WCROS AT A GIVEN STATION IS THE CROSSFLOW AT UPSTREAM STATION
C PLUS IMPINGEMENT JET
C FLOW FROM UPSTREAM STATION MINUS ANY FILM COOLING
C FLOW AT THIS STATION.
C
C** THIS BLOCK IS EXECUTED IF FLOW SPLIT OCCURS AT STATION 1. ISTART=5
WCROS(2, ISLICE, 2) = DELTAN(ISLICE) * (WJ(ISLICE, 1) - WFC(1)) - WFC(2)
WCROS(2, ISLICE, 3) = (1. - DELTAN(ISLICE)) * (WJ(ISLICE, 1) - WFC(1)) - WFC(3)
GO TO 320
C 230 CONTINUE
C**************** THIS BLOCK IS EXECUTED IF FLOW SPLIT STATION IS ON
C THE PRESSURE SIDE: JSENS=1
C WCROS(2,ISLICE,JP) = (1. - DELTAN(ISLICE))*(WJ(ISLICE,JS) - WFC(JS))
C Z - WFC(JP)
C WCROS(2,ISLICE,JM) = DELTAN(ISLICE)*(WJ(ISLICE,JS) - WFC(JS)) - WFC(JM)
C IF (JM.EQ.1) GO TO 250
C IRNG = JDIS - 1
C DO 240 I = 1,IRNG
C IBK = (JM - 2*I)
C IBKP = IBK + 2
C 240 WCROS(2,ISLICE,IBK) = WCROS(2,ISLICE,IBKP) + WJ(ISLICE,IBKP)
C Z - WFC(IBK)
C 250 JNOD = JFIN
C DO 260 I = 2,JNOD,2
C IM8 = I-2
C IF (IM8.EQ.0) IM8 = 1
C 260 WCROS(2,ISLICE,I) = WCROS(2,ISLICE,IM8) + WJ(ISLICE,IM8) - WFC(I)
C GO TO 320
C 270 CONTINUE
C**************** THIS BLOCK IS EXECUTED IF FLOW SPLIT STATION IS ON
C SUCTION SIDE: JSENS=0
C WCROS(2,ISLICE,JP) = DELTAN(ISLICE)*(WJ(ISLICE,JS) - WFC(JS)) - WFC(JP)
C IF (JS.EQ.2) JM = 1
C WCROS(2,ISLICE,JM) = (1. - DELTAN(ISLICE))*(WJ(ISLICE,JS) - WFC(JS))
C Z - WFC(JM)
C IF (JM.EQ.1) GO TO 300
C IF (JM.EQ.2) GO TO 290
C JM1 = JM-2
C DO 280 I = 2,JM1,2
C IBK = (JM-I)
C IBKP = IBK + 2
C 280 WCROS(2,ISLICE,IBK) = WCROS(2,ISLICE,IBKP) + WJ(ISLICE,IBKP)
C Z - WFC(IBK)
C 290 WCROS(2,ISLICE,1) = WCROS(2,ISLICE,2) + WJ(ISLICE,2) - WFC(1)
C 300 CONTINUE
C---------NOW UP THE PRESSURE SIDE
C JNOD = JFIN
C DO 310 I = 3,JNOD,2
C WCROS(2,ISLICE,I) = WCROS(2,ISLICE,I-2) + WJ(ISLICE,I-2) - WFC(I)
C 320 CONTINUE
C ISTART = JSTART
C DO 330 I = ISTART,NFWD
C WCROS(2,ISLICE,I) = WCROS(2,ISLICE,I-2) + WJ(ISLICE,I-2) - WFC(I)
C 330 CONTINUE
C C CALCULATE CROSSFLOW RE & MACH NUMBER SQUARED, AND FILM COOLING RE,
C FOR THE FORWARD REGION.
C DO 360 I = 1,NFWD
C LCOOL = 5*I
C LIN = LCOOL-1
C AMIN = A(LCOOL)
C C EVALUATE COOLANT PROPERTIES AT MEAN TEMPERATURE BETWEEN WALL
C AND COOLANT BULK
\[
\text{TMP} = (T(2,\text{ISLICE,LCOOL}) + T(2,\text{ISLICE,IMU}))/2.
\]

CALL GASTBL(TMP,C,CP,GAM,PD,R,IMU)

\[
X\text{MUC}(I) = X\text{M}
\]

CPC(I) = CP

\[
G\text{MAC}(I) = G\text{AM}
\]

\[
\text{REE}(I) = 12. \times 3600. \times \text{ABS}(\text{WCROS}(2,\text{ISLICE},I) \times \text{DH}(I)/(\text{AMIN} \times X\text{MU})
\]

\[
\text{REFC}(I) = 12. \times 3600. \times \text{WCR}(I)/(S(\text{ISLICE}) \times X\text{MU})
\]

IF (HIC(I) .EQ. 3) AMIN = A(LCOOL) \times (SP(I) - DP(I))/SP(I)

\[
W = \text{WCROS}(2,\text{ISLICE},I)
\]

\[
\text{PBAR} = \text{P}(2,\text{ISLICE},I)
\]

\[
\text{TBAR} = T(2,\text{ISLICE},\text{LCOOL})
\]

\[
\text{AM2}(I) = \left(\frac{W}{(\text{PBAR} \times \text{AMIN})} \right) ^2 \times \text{TBAR}/G\text{MAC}(I)/32.2
\]

IF (AM2(I) .LT. 1.0) GO TO 340

\[
\text{ICHOKE} = I
\]

IF (HIC(I) .NE. 3) GO TO 350

\[
\text{AU2}(I) = \left(\frac{A(I) - \text{SP}(I)}{\text{SP}(I)}\right)
\]

C

C "CALCULATE FLOW DUMPED DIRECTLY INTO TRAILING EDGE REGION.

C"

\[
\text{PAVG} = .5 \times (P(2,\text{ISLICE},\text{NPWD}) + P(2,\text{ISLICE},\text{NPWD}-1))
\]

IF (PAVG .GT. PIH) GO TO 370

\[
\text{WCR} = \text{CD} \times \text{PAVG} \times \text{ADUMP}/(R \times \text{TOG}) \times \sqrt{(32.2 \times \text{PAVG} \times \text{R} \times \text{TOG} / (\text{GAMO} + 1.0) \times (\text{PAVG} / \text{PIH}) ^ 2}
\]

\[
\text{Z} \times \text{SQRT}(32.2 \times \text{PAVG} \times \text{R} \times \text{TOG} / (\text{GAMO} + 1.0) \times (\text{PAVG} / \text{PIH}) ^ 2)
\]

\[
\text{Z} \times \text{SQRT}(32.2 \times \text{PAVG} \times \text{R} \times \text{TOG} / (\text{GAMO} + 1.0) \times (\text{PAVG} / \text{PIH}) ^ 2)
\]

\[
\text{WCR} = \text{CD} \times \text{PAVG} \times \text{ADUMP} / (R \times \text{TOG}) \times \sqrt{(32.2 \times \text{PAVG} \times \text{R} \times \text{TOG} / (\text{GAMO} + 1.0) \times (\text{PAVG} / \text{PIH}) ^ 2}
\]

\[
\text{Z} \times \text{SQRT}(32.2 \times \text{PAVG} \times \text{R} \times \text{TOG} / (\text{GAMO} + 1.0) \times (\text{PAVG} / \text{PIH}) ^ 2)
\]

IF (WCR .LT. YDUHP) WCHKDH = CHKD

IF (WCR .LT. WDUHP) YDUMP = WCR

IF (P(2,ISLICE,I) .GT. PG(I)) WPCDUU = \text{CD} \times 25 \times 3.1415926 \times \text{Z} \times \text{SQRT}(32.2 \times P(2,\text{ISLICE},I) - PG(I)) / \text{Z} \times (R \times \text{WCR}(I) ** 2)

\[
\text{WPC}(I) = \text{WPCDUU} \times \text{DHP}(I) ** 2
\]

\[
\text{WFC}(I+1) = \text{WPCDUU} \times \text{DHP}(I+1) ** 2
\]

400 CONTINUE

C

C "TRAILING EDGE REGION, CALCULATE CROSSFLOW, RE, MACH NUMBER

C SQUARED, AND FILM COOLING RE.

C"

\[
\text{WCROS}(2,\text{ISLICE},\text{ISTRT}) = \text{WCROS}(2,\text{ISLICE},\text{NPWD}-1) + \text{Z}
\]

\[
\text{WCROS}(2,\text{ISLICE},\text{NPWD}) + \text{WDUMP} + \text{WJ}(\text{ISLICE},\text{NPWD}) + \text{WFC}(\text{ISTRT}) - \text{WCROS}(2,\text{ISLICE},\text{ISTRT}+1)
\]

\[
\text{AMIN} = A(\text{NODSP}+5)
\]
EVALUATE COOLANT PROPERTIES AT MEAN TEMPERATURE BETWEEN WALL AND COOLANT BULK

\[ \text{TMP} = \frac{(T(2, \text{ISLICE}, 5\*\text{ISTRT}) + T(2, \text{ISLICE}, 5\*\text{ISTRT}-1))}{2}. \]

CALL GASTBL(TMP,C,CP,GAM,PD,R,XMU)

\[ \text{XMC} = \text{XMU} \]

\[ \text{XMC}(\text{ISTRT}+1) = \text{XMU} \]

\[ \text{CP}(\text{ISTRT}) = \text{CP} \]

\[ \text{CP}(\text{ISTRT}+1) = \text{CP} \]

\[ \text{GAM}(\text{ISTRT}) = \text{GAM} \]

\[ \text{GAM}(\text{ISTRT}+1) = \text{GAM} \]

\[ \text{THP} = \frac{(T(2, \text{ISLICE}, \text{LCOOL}) + T(2, \text{ISLICE}, \text{LCOOL}-1))}{2}. \]

CALL GASTBL(THP,C,CP,GAM,PD,R,XMUC)

\[ \text{XMUC}(\text{ISTRT}) = \text{XMU} \]

\[ \text{XMUC}(\text{ISTRT}+1) = \text{XMU} \]

\[ \text{CP}(\text{ISTRT}) = \text{CP} \]

\[ \text{CP}(\text{ISTRT}+1) = \text{CP} \]

\[ \text{GAM}(\text{ISTRT}) = \text{GAM} \]

\[ \text{GAM}(\text{ISTRT}+1) = \text{GAM} \]

\[ \text{REFC}(\text{ISTRT}) = 12.\*3600.\*\text{WFC}(\text{ISTRT})/(\text{S} (\text{ISLICE}) \* \text{XMUC}) \]

\[ \text{REFC}(\text{ISTRT}+1) = 12.\*3600.\*\text{WFC}(\text{ISTRT}+1)/(\text{S} (\text{ISLICE}) \* \text{XMUC}) \]

\[ \text{RE}(\text{ISTRT}) = 12.\*3600.\*\text{ABS}(\text{WCROS}(2, \text{ISLICE}, \text{ISTRT})) \* \text{DH}(\text{ISTRT})/Z \]

\[ \text{WBAR} = \text{P}(2, \text{ISLICE}, \text{ISTRT}) \]

\[ \text{TBAR} = T(2, \text{ISLICE}, \text{ISTRT}+1) \]

\[ \text{AB2}(\text{ISTRT}) = \left(\frac{T}{\text{WBAR} \* \text{AUN}}\right)^2 \* \text{R} \* \text{TBAR} / \text{GAMC} \]

\[ \text{IF} (\text{AB2}(\text{ISTRT}) \leq 1.0) \text{GO TO} 410 \]

\[ \text{AM2}(\text{ISTRT}) = (\text{W} / (\text{WBAR} \* \text{AMIN}))^{**2} \* \text{CTB} / \text{GAMC} \]

\[ \text{DO} 450 \text{ I = ISTRT, N, 2} \]

\[ \text{LCOOL} = 5\*\text{I} \]

\[ \text{AMIN} = \text{A} (\text{LCOOL}) \]

\[ \text{IS} = \text{IS} + 2 \]

\[ \text{IF} (\text{IHC} (\text{IS}) \cdot \text{EQ.} 3) \text{ AMIN} = \text{A} (\text{NODSF}+5) \* (\text{SP} (\text{IS}) - \text{DP} (\text{IS})) / \text{SP} (\text{IS}) \]

\[ \text{W} = \text{WCROS}(2, \text{ISLICE}, \text{ISTRT}) \]

\[ \text{PBAR} = \text{P}(2, \text{ISLICE}, \text{ISTRT}) \]

\[ \text{TBAR} = T(2, \text{ISLICE}, \text{NODSF}+5) \]

\[ \text{AB2}(\text{ISTRT}) = \left[\frac{1}{\text{SP} (\text{IS})} \right] \]

\[ \text{IF} (\text{AB2}(\text{ISTRT}) \leq 1.0) \text{GO TO} 410 \]

\[ \text{AM2}(\text{ISTRT}) = (\text{W} / (\text{WBAR} \* \text{AMIN}))^{**2} \* \text{CTB} / \text{GAMC} \]

\[ \text{DO} 450 \text{ I = ISTRT, N, 2} \]

\[ \text{LCOOL} = 5\*\text{I} \]

\[ \text{AMIN} = \text{A} (\text{LCOOL}) \]

\[ \text{IS} = \text{IS} + 2 \]

\[ \text{IF} (\text{IHC} (\text{IS}) \cdot \text{EQ.} 3) \text{ AMIN} = \text{A} (\text{NODSF}+5) \]
IF (AM2(I).LT.1.6) GO TO 430  
AMCHOK = AM2(I)  
ICHOKE = I  
430 IF (IHC(IS).NE.3) GO TO 440  
AM2(I) = AM2(I)*((AMIN/A(LCOOL))*2)  
440 AM2(I+1) = AM2(I)  
450 CONTINUE  
C  
C CALCULATE COOLANT CHANNEL FRICTION FACTOR AT EACH STATION.  
C  
DO 560 I = 1,NSTA  
LCOOL = 5*I  
510 IF (WCROS(2,ISLICE,I).LE.0.0) GO TO 550  
C  
C DETERMINE IF RE IS LAMINAR, TRANSITIONAL, OR TURBULENT  
C AND CALCULATE THE FRICTION FACTOR  
C  
IF (IHC(I).EQ.3) GO TO 540  
IF(RE(I).GT.2300.) GO TO 510  
520 FF(I) = DELTA*RE(I)**EPS  
GO TO 560  
530 A1=DELTA*2300.**EPS  
A2=ALPHA*4000.**BETA  
FF(I) = (A2*(RE(I)-2300.)*A1*(4000.-RE(I)))/1700.  
GO TO 560  
540 CONTINUE  
C  
C FOR A PIN PIN ARRAY:  
FF(I) = 1.060*(RE(I)**{-0.3301})  
GO TO 560  
550 CONTINUE  
FF(I) = 0.0  
560 CONTINUE  
C  
C THE FOLLOWING BLOCK IS USED TO COMPUTE THE FILM COOLING EFFECTIVENESS.  
C IF IPILF IS SET = 2  
C  
IF (IPILF.LT.2) GO TO 690  
IFSPLT = 0  
DO 610 I = 1,NSTA  
610 XFC(I) = 0.0  
N = NSTA-1  
C LOCATE FILM COOLING HOLES AND SET UP THE XFC ARRAY  
C  
C---IFSPLT IS AN INDICATOR THAT TELLS WHICH SIDE OF THE BLADE STATION IS TO BE CONSIDERED A PART OF FOR FILM COOLING PURPOSES.  
C--- = 0 IS THE DEFAULT, AND INDICATES SUCTION SIDE  
C--- = 1 WILL INDICATE PRESSURE SIDE.  
C FIRST, MARCH DOWN THE PRESSURE SIDE SEARCHING FOR FILM COOLING HOLES  
IF (DHF(I).GT.0.0) NFC = IFSPLT  
IF (NFC.EQ.0) NFC = NSTA + 1  
XNUM = 0.0  
DO 615 I = 3,NSTA,2  
NOS = 5*I - 4  
615 IF (I.GT.NFC) XFC(I) = XNUM + DLX(NOS)  
C  
C  
C
IF (DHF(I).GT.0.0) GO TO 612
XDUM = XFC(I)
GO TO 615
612
NFC = I
XDUM = 0.0
615 CONTINUE
C
C-- SUCTION SIDE
C
IF (DHF(I).GT.0.0) NFC = 1 - IPSPLT
IF (NFC.EQ.0) NFC = NSTA + 1
XDUM = 0.0
DO 625 I = 2, NSTA
NOS = 5*I - 4
IF (I.GT.NFC) XPC(I) = XDUM + DLX(NOS)
IF (DHF(I).GT.0.0) GO TO 622
XDUM = XFC(I)
GO TO 625
622
NFC = I
XDUM = 0.0
625 CONTINUE
C
C INT. J. HT. & MASS TRANS., VA, 1965, PP 55-65
C
FLMEFP = 3.09*((X/(M*S))*(RE*MUC/MUG)**(-1/4) + 4.1)**(-.8)
C
IFCS = 0
IFCP = 0
IF (WFC(I).LE.0.0) GO TO 630
IFCS = 1 - IPSPLT
IFCP = IPSPLT
IF (RHOGA(I).GT.0.0) EMES(I) = 144.*WFC(I)/(RHOGA(I)*S(ISLICE))
630 CONTINUE
DO 650 I = 2, NSTA
ISENS = I - 2*(I/2)
IF (RHOGA(I).GT.0.) EMES(I) = 144.*WFC(I)/(RHOGA(I)*S(ISLICE))
FLMEFP(I) = 0.0
IF (ISENS.EQ.0) GO TO 640
C PRESSURE SIDE SUPPLY HOLE LOCATIONS
IF (WFC(I).GT.0.0) IFCP = I
NFCSUP(I) = IFCP
GO TO 650
640 CONTINUE
C SUCTION SIDE SUPPLY HOLE LOCATIONS
IF (WFC(I).GT.0.0) IFCS = I
NFCSUP(I) = IFCS
650 CONTINUE
C
TMP = TG(1)
CALL GASTBL(TMP,C,CPM,GAM,PD,XMUM)
C
C FINALLY, CALCULATE THE EFFECTIVENESS
C
DO 680 I = 1, NSTA
IMS = NFCSUP(I)
IF (XFC(I).EQ.0.0.OR.EMES(IMS).EQ.0.0.OR.REPC(IMS).EQ.0.0) GO TO 680
Z = 1
C3 = CPC(IMS)/CPM
XBAR = (XFC(I)/EMES(IMS))*((REPC(IMS)*XMUC(I)/XMUM)**(-.25))
ETAPRM = 3.09*(XBAR+4.1)**(-.8)
FLMEFP(I) = C3*ETAPRM/(1.0 + (C3-1.0)*ETAPRM)
IF (FLMEFP(I).GT.1.0) FLMEFF(I) = 1.0
680 CONTINUE
C---- SOURCE.NFLSPLP

SUBROUTINE FLSPLT(AJET,EPSN,ISLICE,NODSF,DELT,J,DELTAN,ICONV)
DIMENSION DELTAN(15), AJET(80), JSOLDS(25)

C--- SOURCE.NFLSPLP---A SUBROUTINE TO SET THE STATION AT WHICH COOLING
AIR FLOW SPLITS BETWEEN THE SUCTION
AND THE PRESSURE SIDE FLOW CHANNELS.

C
 INPUT TO FLSPLT IS THE PRESSURE MATCH PARAMETER, EPSN; THE NO. OF
 NODES (NODSF) IN THE IMPINGEMENT REGION;
 JS COMES IN AS THE CURRENT FLOW SPLIT STATION NO., AND IS
 RETURNED AS THE NEW STATION IF A CHANGE IS NEEDED.
 DELTAN COMES IN AS THE CURRENT FRACTION OF FLOW SPLIT TO
 SUCTION SIDE FROM AN IMPINGEMENT
 JET AT JS. IF A CHANGE IN JS IS NOT NEEDED, DELTAN IS
 USED TO FINE TUNE THE SPLIT.
 ICONV INDICATES IF CONVERGENCE IS COMPLETE.
 = 0--NOT DONE; = 1--OK.

NFWD = NODSF/5
IF (IUNSTB.EQ.1) GO TO 280
IF (IDELT.NE.1) GO TO 220
JNUMS = 0
IUNSTB = 0
NUMS = 0
JSGNCH=0
JOUTRG=0
DO 210 I = 1,2S
210 JSOLDS(I) = 0
CONTINUE

CRITF = .002
ICONV = 0
JSENS = JS - 2*(JS/2)
C******* (SUCTION - PRESSURE SIDE PRESSURES)/ SUCTION SIDE = EPSN
IF (ABS(EPSN).LT.CRITF) GO TO 280

C C******* IF EPSN < 0.0; NEED TO INCREASE FLOW TO PRESSURE SIDE
C******** EPSN > 0.0; NEED TO INCREASE FLOW TO SUCTION SIDE

C IF (JTIMES.EQ.0) GO TO 246
C
C********** JTIMES = 0, THIS IS FIRST CHECK AT THIS STATION,
C SO ROUGH ADJUST DELTAN;
C********** 1, HAVE BEEN HERE BEFORE, SO FINE TUNE DELTAN.
C IF ( JSGNCH.GT.0 ) GO TO 247
C
C********** JSGNCH = 0, THERE HAS NOT BEEN A PRIOR SIGN CHANGE IN EPSN;
C********** = 1, THERE HAS BEEN A SIGN CHANGE BEFORE,
C SO STAY AT THIS STATION
C

242 IF (EPSO/EPSN.LT.0.) JSGNCH = 1
C 247 CONTINUE
IF (EPLAST/EPSN.GE.0) GO TO 243
DELTAO = DELAST
EPSO = EPLAST

243 CONTINUE
IF (JSGNCH.EQ.0) GO TO 252
IF (NUMS.GT.0) GO TO 248
EPSMIN = ABS(EPSN)
DELTAO = DELTAN(ISLICE)

248 NUMS = NUMS + 1
IF (ABS(EPSN).GT.EPSMIN) GO TO 249
EPSMIN = ABS(EPSN)
DELTAO = DELTAN(ISLICE)

249 CONTINUE
IF (NUMS.LT.4) GO TO 252
IF (JNUMS.EQ.1) GO TO 250
NUMS = 0
JNUMS = 1
DELTAN(ISLICE) = DELTAO
JTIMES = 0
JSGNCH = 0
JOUTRG = 0
GO TO 290

250 CONTINUE
DELAST = DELTAN(ISLICE)
DELTAN(ISLICE) = DELTAO
JUNSTB = 1
GO TO 290

C 246 JTIMES = 1
EPSO = EPSN
DELTAO = DELTAN(ISLICE)
IF (EPSO.GT.0.0) DELTAN(ISLICE) = (1.0+DELTAN(ISLICE))/2.0
IF (EPSO.LT.0.0) DELTAN(ISLICE) = DELTAN(ISLICE)/2.
IF (DELTAN(ISLICE).EQ.DELTAO) DELTAN(ISLICE) = DELTAN(ISLICE) +
Z (.5-DELTAN(ISLICE))/5.
GO TO 290

C 252 CONTINUE
TERM = EPSN*(DELTAO-DELTAN(ISLICE))/(EPSO-EPSN)
IF (TERM.EQ.0.) TERM = .05
IF (JSGNCH.GT.0) GO TO 255
DELTAO = DELTAN(ISLICE)
EPSO = EPSN

255 CONTINUE
DELAST = DELTAN(ISLICE)
DELTAN(ISLICE) = DELTAN(ISLICE) -TERM
IF ( DELTAN(ISLICE).LT.1.0.AND.DELTAN(ISLICE).GT.0.0) GO TO 290
IF (JOUTRG.GT.0) GO TO 258
IF (DELTAN(ISLICE).LT.0.0) DELTAN(ISLICE) = .01
IF (DELTAN(ISLICE).GT.1.0) DELTAN(ISLICE) = .99
JOUTRG = 1
GO TO 290

C 258 CONTINUE
JOUTRG = 0
C
JSGNCH = 0
JTIMES = 0
JSOLDS(JS) = 1
IF (DELTAN(ISLICE).LT.1.) GO TO 265
C
C *** MOVE JS IN PRESSURE DIRECTION
IF (JSENS.EQ.0) GO TO 262
261 JS = JS + 2
IF (AJET(JS).LE.0.) GO TO 261
GO TO 285
C
C 262 CONTINUE
IF (JS.EQ.2) JS = 1
IF (JS.GT.2) JS = JS - 2
IF (AJET(JS).LE.0.) GO TO 262
GO TO 285
C
C 265 CONTINUE
C *** MOVE JS IN SUCTION DIRECTION
IF (JSENS.EQ.0) GO TO 267
IF (JS.EQ.1) JS = 2
IF (JS.GT.3) JS = JS - 2
IF (AJET(JS).GT.0.) GO TO '265
JSSENS = JS - 2*(JS/2)
GO TO 265
C
C 267 CONTINUE
JS = JS + 2
IF (AJET(JS).LE.0.) GO TO 267
GO TO 285
C
C******** Get Ready to Leave Subroutine
C
C THIS BLOCK IS EXECUTED IF CONVERGENCE WAS DETECTED
C
260 ICONV = 1
JTIMES = 0
IF (IUNSTB.EQ.1) WRITE(6,284) ISLICE,IDELT,JS,DELTAN(ISLICE)
IF (IUNSTB.EQ.1) WRITE(8,284) ISLICE,IDELT,JS,DELTAN(ISLICE)
IUNSTB = 0
EPLAST = EPSN
RETURN
284 FORMAT(1H2,40(***),40(***)/' SLICE ',I2,' POOR FLOW SPLIT ','
Z ',I3,' ITERATIONS, SPLIT AT STATION ','
Z I2,', BEST SPLIT IS AT DELTA = ',F6.4)
C
C THIS BLOCK IS EXECUTED FOR AN ABNORMAL EXIT---PROGRAM IS TERMINATED
C
769 CONTINUE
WRITE(6,792) DELTAN(ISLICE)
WRITE(8,792) DELTAN(ISLICE)
STOP
792 FORMAT(/5X, '****** FLOW SPLIT CANNOT BE MADE AS SPECIFIED,'
Z ',DELTA = ',F9.5)
C

285 IF (JSOLDS(JS).EQ.1) GO TO 789
DELTAN(ISLICE) = .50

C THIS BLOCK IS THE USUAL EXIT AFTER ADJUSTING THE FLOW SPLIT
C
290 CONTINUE
EPLAST = EPSN
IF (JSGNCH.EQ.0) DELAST = DELTA0
IDELT = IDELT + 1
RETURN
END

C---- SOURCE.NGASDAT

BLOCK DATA
C
C-- SOURCE.NGASDAT---
C
COMMON /GAAS/ GS(200), NG
DATA GS/620., 1160., 1700., 2240., 2780., 3320.,
Z .02564, .03580, .04548, .05467, .06435, .07475,
Z .2511, .2681, .2814, .2939, .3070, .3214,
Z .706, .706, .705, .703, .702, .699,
Z .07233, .09458, .11369, .13063, .14683, .16256,
Z 170*0.0/., NG /6/

C--GS IS TABLE OF AIR PROPERTIES VS TEMPERATURE AT CONSTANT PRESSURE
C--- PROPERTY VALUES ARE FROM POPEL & SVEHLA, TN D-7488, AT 20 ATM.
C--- NG IS THE NUMBER OF TEMPERATURE ENTRIES IN THE TABLE
C--- ENTRIES IN GS ARE:
C--- 1ST NG ARE TEMPERATURE (F)
C--- 2ND NG ARE THERMAL CONDUCTIVITY, (BTU/(HR*FT*R))
C--- 3RD NG ARE SPECIFIC HEAT, (BTU/(LBM*R))
C--- 4TH NG ARE PRANDTL NUMBER
C--- 5TH NG ARE VISCOSITY, (LBM/(FT*HR))
C
END

C---- SOURCE.NGASTB

SUBROUTINE GASTBL(TMP,C,CP,GAM,PD,R,XMU)

A SUBROUTINE TO LOOK UP GAS PROPERTIES IN AN INPUT TABLE (GS(200))
WHERE TMP = TEMPERATURE AT WHICH PROPERTIES ARE TO BE EVALUATED (R)
C = GAS THERMAL CONDUCTIVITY (BTU/(HR*FT*R))
CP = GAS SPECIFIC HEAT (BTU/(LBH*R))
GAM = RATIO OF SPECIFIC HEATS
PD = PRANDTL NUMBER
R = SPECIFIC GAS CONSTANT (FT*LB)/(LBH*R)
XMU = VISCOSITY (LBM/(FT*HR))

COMMON /GAAS/ GS(200), NG
DIMENSION AC(5)

C

TMP1=TMP - 460.
IF(TMP1.GT.GS(1)) GO TO 200
100 AP1=0.0
AP2=1.0
I1=2
I2=1
GO TO 500
200 DO 300 I=1,NG
   IF (GS(I).GT.TMP1) GO TO 400
   CONTINUE
   TMP1=GS(N)
400 I2=I1-1
   AP1=(TMP1-GS(I2))/(GS(I1)-GS(I2))
   AP2=1.0-AP1
500 DO 600 J=1,4
   I1=I1+NG
   12=12+NG
600 AC(J)=AP1*GS(I1)+AP2*GS(12)
   AC(5)=1.0/(1.0-R/(778.2*AC(2)))
C=C(1)
P=C(2)
D=C(3)
XMU=C(4)
GAM=C(5)
RETURN
END

C---- SOURCE.NGAUS
SUBROUTINE GAUSS(N,K)

C DEBUGGING OUTPUT:
C IWR CAN BE SET DYNAMICALLY IN ORDER TO GET DEBUG OUTPUT OF
C SELECTED ROWS OF THE MATRIX, BEFORE OR AFTER REDUCTION.
C
WRITE(8,57)
57 FORMAT(' ENTER NUMBER OF ROW TO BE DISPLAYED. USE I3 FORMAT')
58 READ(7,59) NROW
59 FORMAT(I3)
   IF (NROW.EQ.0) GO TO 63
   KP = K+1
   WRITE(8,60) NROW
60 FORMAT(' TCOP MATRIX, ROW NO. ',I3)
61 FORMAT(5(' ',I3,' ',D17.10,' '))
62 FORMAT(' ENTER ANOTHER ROW NO. OR 000 TO CONTINUE PROCESSING')
GO TO 58

C CONTINUE

63 JPIV = K/2 + 1
65 N1 = N-1
DO 100 I = 1,N1
JS = I+1
JF = I + K/2
IF (JF.GT.N) JF = N
PIVOT = TCOF(I,JPIV)
IF (PIVOT.EQ.0.0) GO TO 130
DO 90 J = JS,JF
JR = JPIV-J+I
IF (DABS(TCOP(J,JR)) .LT. 1.0D-30) GO TO 90
FB = TCOF(J,JR) /PIVOT
TCOF(J,JR) = 0.0
LS = JR + 1
LF = LS + K/2
IF (LF.LT.LS) GO TO 85
DO 80 L = LS,LF
LR = L+JPIV+1-LS
IF (LR.GT.K) GO TO 85
80 TCOF(J,L) = TCOF(J,L) - FM*TCOF(I,LR)
85 TCOF(J,K+l) = TCOF(J,K+1) - FM*TCOF(I,K+l)
90 CONTINUE
100 CONTINUE

C CONTINUE

155 IF (IWR.EQ.0) GO TO 163

C DEBUGGING OUTPUT:

158 WRITE(8,57)
READ(7,59) NROW
IF (NROW.EQ.0) GO TO 163
KP = K+1
WRITE(8,60) NROW
WRITE(8,61) (I,TCOP(NROW,I),I=1,KP)
WRITE(8,62)
GO TO 158

C CONTINUE

163 TCOF(N,K+1) = TCOF(N,K+1)/TCOF(N,JPIV)
DO 120 I = 1,N1
IIN = N-I
JP = I/2
SUM = TCOF(IIN,K+1)
DO 115 J = 1,JP
IJ = J+IIN
IF (IJ.GT.N) GO TO 117
115 SUM = SUM - TCOF(IJ,K+1)*TCOF(IIN,JP)
117 CONTINUE
120 TCOF(IIN,K+1) = SUM/TCOF(IIN,JPIV)
125 CONTINUE
RETURN
130 WRITE(7,135) I
135 FORMAT(/' DIAGONAL ELEMENT FOR ROW ',I2,' IS ZERO. NO ',
82x652}
GO TO 125
END

C-----SOURCE.NGETINT
SUBROUTINE GETIN
C
C
SOURCE.NGETINT----
C
COMMON /BOUND/ BCXS(400), BCXP(400), BCHGS(1000), BCHGP(1000), NGETINT 1209
C
Z BCTGS(1000), BCTGP(1000), BCQGS(1000), BCQGP(1000), NGETINT 1210
C
Z BCPGS(1000), BCPGP(1000), THUBIN(400), THUB(80), NGETINT 211
C
Z QHUBIN(400), QHUB(80), TTIPIN(400), TTIP(80), NGETINT 1212
C
Z QTIPIN(400), QTIP(80), ROHVG(400), PEX(400), NGETINT 1213
C
Z BTCIME(50), TTIO(50), PTIO(50), WPLEN, NGETINT 1214
C
Z WSVST(50), AKCTBL(20), AKWTBL(20), NBCS, NBCP NGETINT 1215
C
COMMON /PLMCOL/ ROHVGA(80), PG(80), XFC(80), FLMEFF(80), NGETINT 1216
C
Z XMUC(80), EMES(80), REFC(80), NFCSUP(80), NGETINT 1217
C
COMMON /IMPOR/ CIMP1, CIMP2, CIMP3, CIMP4, CIMP5, CIMP6, CIMP7, NGETINT 1220
C
Z DIMP1, DIMP2, DIMP3, DIMP4, DIMP5, DIMP6 NGETINT 1221
C
COMMON /RADL/ APLN(15), DPLN(15), RIN(15), ROUT(15), NGETINT 1223
C
Z PIN(15), TIN(15), W(15), WS NGETINT 1224
C
COMMON /SPECL/ CHANL(8000), TITLE(30), INDCHN(2000), NGETINT 1226
C
Z IPOINT, MD1, MD2, MD3, IADJIN, IWRITE NGETINT 1227
C
COMMON /TCO/ ADUMP, BTA, CD, CP, NGETINT 1230
C
Z GAM, PIM, E, SPAN, TOG, NGETINT 1230
C
Z WDUMP, WIM, AKC(15,80), AKW(15,80), NGETINT 1231
C
Z A(400), AJET(80), AM2(80), CNUM(80), NGETINT 1232
C
Z DH(80), DHF(80), DHJ(80), NGETINT 1233
C
Z DLX(400), FF(80), HC(80), HG(80), NGETINT 1234
C
Z P2(15,80), PEXIT(15), PUMP(80), QG(80), NGETINT 1235
C
Z QSNK(80), RR(80), S(15), T(2,15,400), NGETINT 1236
C
Z TG(80), TAU(400), WFC(80), NGETINT 1237
C
Z WJ(15,80), WCROS(2,15,80), XN(80), NGETINT 1238
C
Z ICOR, IPEL, IHUB, ITIP, NGETINT 1239
C
Z ISBLOK, ISLICE, NBLKSZ, NSLICE, NGETINT 1240
C
Z NFWD, NSTA, IHC(80), NGETINT 1241
C
COMMON /TRANS/ RHOC, RHOM, SPHTC, SFHTM, NGETINT 1242
C
Z DLTIME, DTME, TEP, TYM, NGETINT 1244
C
COMMON /UNITS/ CINCH(2), CHTC(2), CRFLX(2), CPBFR(2), CSMFL(2), NGETINT 1246
C
Z CTMPF(2), CTCIN(2), CDEN(2), CSPHT(2), CGASC(2), NGETINT 1247
C
Z CVISC(2), CRHOVG(2), IUNITS NGETINT 1248
C
DIMENSION THK(3), DTLX(5), TPLMLH(10) NGETINT 1249
C
NAMELIST /TITL/ TITLE NGETINT 1250
C
NAMELIST /CHANL/ NSLICE, NSTA, IADJIN, IWRITE, NGETINT 1254
C
NAMELIST /BC/ NBCS, NBCP, BCXS, BCXP, BCHGS, BCGP, NGETINT 1255
C
Z BCTGS, BCTGP, BCQGS, BCQGP, BCPGS, BCPGP, NBCS, NBCP NGETINT 1256
C
Z THUBIN, QHUBIN, TTIPIN, QTIPIN, ROHVG NGETINT 1257
C
Z PEX, BTCIME, TTIO, PTIO, WPLEN NGETINT 1258
C
73
**NAMELIST /CTRL/**

- **NFWD**: Not specified.
- **ICOR**: Not specified.
- **NGEO**: Not specified.

**NAMELIST /PROPS/**

- **CD**: Coefficient of Drag.
- **SPAN**: Span.
- **ADUMP**: Adiabatic Coefficient.
- **DHYD**: Hydraulic Diameter.
- **APLEN**: Adiabatic Length.
- **RO**: Density.
- **CIMP1**: Coefficient of Impingement 1.
- **CIMP2**: Coefficient of Impingement 2.
- **CIMP3**: Coefficient of Impingement 3.
- **CIMP4**: Coefficient of Impingement 4.
- **CIMP5**: Coefficient of Impingement 5.
- **CIMP6**: Coefficient of Impingement 6.
- **CIMP7**: Coefficient of Impingement 7.
- **DIMP1**: Diameter of Impingement 1.
- **DIMP2**: Diameter of Impingement 2.
- **DIMP3**: Diameter of Impingement 3.
- **DIMP4**: Diameter of Impingement 4.

**NAMELIST /GEO/**

- **ISTA**: Initial Station Number.
- **ISTB**: Final Station Number.
- **THK**: Thickness.
- **TDLX**: Distance from Upstream Node.
- **TDHJ**: Hydraulic Diameter of Impinging Jet Hole.
- **TXN**: Spanwise Spacing of Impinging JETs.
- **TRR**: Radial Location of This Station.
- **IHCT**: Type of Inside Heat Transfer.
- **TDP**: Pin Fin Diameter.
- **TSP**: Pin Fin Spacing.
- **TFLMHL**: Total FILM CoOLing Mass Flow Rate.

**DATA TILED/**

- **NSLICE**: Number of Slices of the Blade.
- **IHUB**: Indicates Temperature Distribution at Hub End.
  - **1**: Adiabatic Surface.
  - **2**: Heat Flux Specified.
- **ITIP**: Indicates Temperature Distribution at Tip End.
  - **1**: Adiabatic Surface.
  - **2**: Heat Flux Specified.
- **IADJIN**: Indicates Temperature Adj ustment.
  - **0**: Hold PTIO and Adjust WPLEN.
  - **> 0**: Fix WPLEN and Adjust PTIO.
- **ISTA**: First Station Number.
- **ISTB**: Final Station Number.
- **THK**: Thickness.
- **TDLX**: Distance from Upstream Node.
- **TDHJ**: Hydraulic Diameter of Impinging Jet Hole.
- **TXN**: Spanwise Spacing of Impinging JETS.
- **TRR**: Radial Location of This Station.
- **IHCT**: Type of Inside Heat Transfer.
- **TDP**: Pin Fin Diameter.
- **TSP**: Pin Fin Spacing.
- **AKCTBL**: Table of Cladding Thermal Conductivity.
- **AKWTBL**: Table of Wall Metal Thermal Conductivity.
- **RHOG**: Hot Gas Free Stream Mass Velocity.
- **COOLING**: Cooling System.
- **INPUT**: Input in (LBM/SEC FT**2), or (KG/SEC M**2) if UNITS=1.
- **RHOC**: Density of Outer Coating.
- **RHOM**: Density of Wall Metal.
C DLTYME = TIME STEP SIZE FOR TRANSIENT CALCULATIONS (SEC)
C TMMAX = MAX. TIME (SEC) TO WHICH TRANSIENT IS CARRIED.
C TEPS = FRACTION OF TIME STEP AT WHICH TEMP. IS EVALUATED. (NEW = OLD + TEPS*(NEW-OLD))
C WSVST = TABLE OF WHEEL SPEED VS TIME, (RPM VS SEC), ODD SUBSCRIPTS ARE SPEED, EVEN ARE TIME, WSVST(2)=0.0
C CIMP1 TO CIMP5 ARE EXPONENTS TO BE USED IN A GENERAL IMPINGEMENT WITH CROSSFLOW CORRELATION. IF NOT SPECIFIED, THEN THE BUILT IN KIRCHER-TABAKOFF CORRELATION IS USED.
C C See subroutine HCOOLT for description of general correlation.
C
C INITIALIZE:
C 100 CONTINUE
IEND = 0
IADJIN = 0
IHUB = 2
ITIP = 2
CIMP1 = 0.0
DIMP1 = 0.0
ADUMP = 0.0
IFILM = 0
IUNITS = 2
ALPHA = .04
BETA = -.16
DELTA = 16.
EPS = -1.
CD = .8
C
C GAS CONSTANT FOR AIR, FT LBF/LBM R
R = 53.35
C
C-- SET VALUES FOR UNITS CORRECTION FACTORS---
C-- ... (1) CONVERTS FROM SI TO ENGLISH, ... (2) MAKES NO CONVERSION--
C-- ALREADY IN ENGLISH
C
C--- CINCH(1) IS CONVERSION FACTOR FROM (CM) TO (IN)
CINCH(1) = .39370
CINCH(2) = 1.0
C--- CHTC(1) IS CONVERSION FACTOR FROM (WATTS/M**2 K) TO (BTU/HR FT**2R)
CHTC(1) = .17623
CHTC(2) = 1.0
C--- CHFLX(1) IS CONVERSION FACTOR FROM (WATTS/M**2) TO (BTU/HR FT**2)
CHFLX(1) = .31721
CHFLX(2) = 1.0
C--- CPBSR(1) IS CONVERSION FACTOR FROM (KILOPASCALS) TO (PSIA)
CPBSR(1) = .14503
CPBSR(2) = 1.0
C--- CMSPL(1) IS CONVERSION FACTOR FROM (KG/HR) TO (LBH/HR)
CMSPL(1) = 2.67924
CMSPL(2) = 1.0
C--- CTMPF(1) IS CONVERSION FACTOR FROM (K) TO (R)
CTMPF(1) = 1.8
CTMPF(2) = 1.0
C--- CTCON(1) IS CONVERSION FACTOR FROM (WATTS/M K) TO (BTU/HR FT R)
CTCON(1) = .57817
CTCON(2) = 1.0
C--- CDEN(1) IS CONVERSION FACTOR FROM (KG/M**3) TO (LBM/FT**3)
CDEN(1) = .06243
C--- CSPHT(1) IS CONVERSION FACTOR FROM (J/KG K) TO (BTU/LBPI)
CSPHT (1) = 0.000239
CSPHT (2) = 1.0
C--- CVISC(1) IS CONVERSION FACTOR FROM (PA SEC) TO (LBM/FT HR)
CVISC (1) = 2415.096
CVISC (2) = 1.0
C--- CGASC(1) IS CONVERSION FROM (J/KG K) TO (FT LBF/LBPI)
CGASC (1) = 11660.2
CGASC (2) = 1.0
C--- CRHOVG IS CONVERSION FROM (KG/SEC M**2) TO (LBM/SEC FT**2)
CRHOVG (1) = 0.0204823
CRHOVG (2) = 1.0

DO 105 I = 1,30
   TITLE (I) = TIKLE
C
DO 106 I = 1,1000
   BCHGS (I) = 0.0
   BCHGP (I) = 0.0
   BCTGS (I) = 0.0
   BCTGP (I) = 0.0
   BCQGS (I) = 0.0
   BCQGP (I) = 0.0
   BCFGP (I) = 0.0
   RHOC = 0.0
   RHOM = 0.0
   SPHTC = 0.0
   SPHTM = 0.0
   DO 107 I = 1,400
      THUBIN (I) = 0.0
      QHubin (I) = 0.0
      TTIIPIN (I) = 0.0
      QTIPIN (I) = 0.0
      RHOVG (I) = 0.0
C
   PEX (I) = 0.0
   DO 108 I = 1,50
      BCTIME (I) = 0.0
      TTIO (I) = 0.0
      PTIO (I) = 0.0
      WSVST (I) = 0.0
C
   RR (I) = 0.0
C
DO 110 I = 1,6000
   CHANL (I) = 0.0
C
   DO 112 I = 1,15
      PEXIT (I) = 0.0
   DO 112 J = 1,80
      AKC (I, J) = 0.0
   AKW (I, J) = 0.0
C
   DO 115 I = 1,2000
   INDCHN (I) = 0
   DO 115 J = 1,20
      AKCTBL (I) = 0.0
   AKWTBL (I) = 0.0
   IPILOT = 0
   IWRITE = 0
INEDIT = 0
TEPS = 1.0
DLTYME = 0.0
READ (5, TITL)
READ (5, CHANLS)
READ (5, BC)
IF (BCHGS (1) .EQ. 0.0) BTA = 1.0
IF (BCQGS (1) .EQ. 0.0) BTA = 0.0
IF (TTIPIN (1) .GT. 0.0) ITIP = 1
IF (THUBIN (1) .GT. 0.0) IHUB = 1
IF (ABS (TTIPIN (1)) .GT. 0.0) ITIP = 3
IF (ABS (THUBIN (1)) .GT. 0.0) IHUB = 3
WS = WSVST (1)
C
PEXIT (1) = PEX (1)
DO 175 ICHLNO = 1, NSLICE
C
ICHN0 IS THE CHANNEL NUMBER; = 1 AT THE HUB, = NSLICE AT THE TIP
C
READ (5, CONTL)
NODSF = 5 * NFWD
C-- NODSF IS THE NUMBER OF NODES IN THE FORWARD REGION
C
NODST = 5 * NSTA
C-- NODST IS THE TOTAL NUMBER OF NODES IN THE BLADE SLICE ICHLNO
C
NBLKSZ = (15 + 2 * NODST) + 8 * NSTA
C-- NBLKSZ IS THE SIZE OF THE DATA BLOCK RESERVED IN CHANL ARRAY FOR THIS SLICE ICHLNO
C
ISBLOK = IEND + 1
C-- ISBLOK IS THE STARTING POINT IN CHANL ARRAY FOR THIS BLOCK OF DATA
C
INSTRT = 15 + (ICHLNO - 1) * (15 + NSTA)
C-- INSTRT IS THE STARTING POINT IN INDCHN ARRAY FOR THIS BLOCK OF INTEGER DATA
C
INDCHN (ICHLNO) = INSTRT
INDCHN (INSTRT) = ICHLNO
INDCHN (INSTRT + 1) = IFILM
INDCHN (INSTRT + 2) = ICOR
INDCHN (INSTRT + 3) = NFWD
INDCHN (INSTRT + 4) = NSTA
INDCHN (INSTRT + 5) = ISBLOK
INDCHN (INSTRT + 6) = NBLKSZ
INDCHN (INSTRT + 7) = IPILOT
INDCHN (INSTRT + 8) = MD1
INDCHN (INSTRT + 9) = MD2
INDCHN (INSTRT + 10) = MD3
INDCHN (INSTRT + 12) = IHUB
INDCHN (INSTRT + 13) = ITIP
IHCTZ = INSTRT + 14
C-- IHCTZ IS THE RELATIVE ZERO POINT IN INDCHN FOR STORAGE OF THE
C INDICATOR IH
READ (5, PROPS)
S (ICHLNO) = SPAN * CINCH (IUNITS)
APLN (ICHLNO) = APLEN * CINCH (IUNITS) * CINCH (IUNITS)
DPLN (ICHLNO) = DHYD * CINCH (IUNITS)
BOUT (ICHLNO) = RO * CINCH (IUNITS)
RIN (ICHLNO) = HI * CINCH (IUNITS)
NOW, /GEO/ IS READ, N GEO TIMES, AND THE DATA STORED IN CHANL ARRAY. ISBLOK = THE STARTING POINT IN CHANL ARRAY FOR CHANNEL ICHLN0 DATA.

FIRST, STORE THE SINGLE VALUED DATA:

CHANL(ISBLOK) = CD
CHANL(ISBLOK+1) = ALPHA
CHANL(ISBLOK+2) = BETA
CHANL(ISBLOK+3) = DELTA
CHANL(ISBLOK+4) = EPS
CHANL(ISBLOK+6) = ADUMP*CINCH(IUNITS)**2
CHANL(ISBLOK+7) = SPAN*CINCH(IUNITS)
CHANL(ISBLOK+8) = BTA
CHANL(ISBLOK+9) = DLYME
CHANL(ISBLOK+10) = TEPS

THEN THE ARRAYS ARE STORED:

THE FOLLOWING ARE STORED BY NODE NUMBER:

THK (TAU), TDLX (DLX)

THE REST ARE STORED BY STATION NUMBER:

TDHJ (DHJ), TDHF (DHF), TXN (XN), TRR (RR),

(TDP (DP), TSP (SP),

(AKC), (AKW), IHCT (IHC).

IITHKZ = ISBLOK + 14
ITDLXZ = ISBLOK + 14 + NODST
ITDHJZ = ISBLOK + 14 + 2*NODST + NSTA
ITDXNZ = ISBLOK + 14 + 2*NODST + 3*NSTA
ITDPZ = ISBLOK + 14 + 2*NODST + 4*NSTA
ITSPZ = ISBLOK + 14 + 2*NODST + 5*NSTA
IEND = ISBLOK + 14 + 2*NODST + 8*NSTA

THK(1) = 0.0
DO 170 I = 1, N GEO
ISTB = 0
READ(5,GEO)
IF (THK(1).LE.0.0) THK(1) = .0001*THK(2)
IF (TDLX(1).GT.0.0*TDLX(4).OR.TDLX(4).GT.1.2*TDLX(1)) WRITE(8,136) ICHLN0,ISTA
136 FORMAT(/' CHANNEL ',I2,' STATION ',I3,'---TDLX VALUES DO NOT LOOK RIGHT')
IF (TDHJ.GT.0. .AND.TXN.LT. 1.1*TDHJ) WRITE(8,137) ICHLN0,ISTA
137 FORMAT(/' CHANNEL ',I2,' STATION ',I3,'---HOLE SPACING AND DIAMETER DO NOT LOOK RIGHT')
THK(1) = 0.0
DO 170 I = 1, N GEO
ISTB = 0
READ(5,GEO)
IF (THK(1).LE.0.0) THK(1) = .0001*THK(2)
IF (TDLX(1).GT.0.0*TDLX(4).OR.TDLX(4).GT.1.2*TDLX(1)) WRITE(8,136) ICHLN0,ISTA
136 FORMAT(/' CHANNEL ',I2,' STATION ',I3,'---TDLX VALUES DO NOT LOOK RIGHT')
IF (TDHJ.GT.0. .AND.TXN.LT. 1.1*TDHJ) WRITE(8,137) ICHLN0,ISTA
137 FORMAT(/' CHANNEL ',I2,' STATION ',I3,'---HOLE SPACING AND DIAMETER DO NOT LOOK RIGHT')

J REPRESENTS THE STATION NUMBER IN THIS CASE

JSENS = J - 2*(J/2)

JSENS = 0 INDICATES THAT STATION NO. IS EVEN AND STATION IS ON SUCTION SIDE
JSENS = 1 INDICATES THAT STATION NO. IS ODD AND STATION IS ON PRESSURE SIDE
C

IARG = ITDHz + J
CHANL(IARG) = TDHz*CINCH(IUNITS)
IARG = ITDHz + J
CHANL(IARG) = TDHz*CINCH(IUNITS)
IARG = ITXNZ + J
CHANL(IARG) = TXNZ*CINCH(IUNITS)
IARG = ITRRZ + J
CHANL(IARG) = TRR*CINCH(IUNITS)
IARG = ITSPZ + J
CHANL(IARG) = TSP*CINCH(IUNITS)
IARG = ITHKZ + J
INDCHN(IARG) = IHTCT
NODOUT = 5*J - 4

C

NODOUT IS THE NODE NO. ON THE OUTSIDE SURFACE AT STATION J
C 5 IS THE NUMBER OF NODES AT STATION J

C

145 CONTINUE
LOCA = ITDLXZ + NODOUT
IF (TDLX(3) .LE. 0.) GO TO 155
DO 150 L = 1, 5
LOCAL = LOCA + L - 1
150 CHANL(LOCAL) = TDLX(L)*CINCH(IUNITS)
GO TO 160
155 CHANL(LOCAL) = TDLX(1)*CINCH(IUNITS)
CHANL(LOCAL+3) = TDLX(4)*CINCH(IUNITS)
AA = TDLX(1)
B = (TDLX(4) - TDLX(1)) / (THK(1) + THK(2))
CHANL(LOCAL+1) = (AA + B*THK(1)) *CINCH(IUNITS)
CHANL(LOCAL+2) = (AA + B*(THK(1) + THK(2)/2.)) *CINCH(IUNITS)
CHANL(LOCAL+4) = (AA + B*(THK(1) + THK(2) + THK(3)/2.)) *CINCH(IUNITS)
CONTINUE

C

160 CONTINUE
LOCA = ITHKZ + NODOUT
CHANL(LOCAL) = THK(1)*CINCH(IUNITS)
CHANL(LOCAL+2) = THK(2)*CINCH(IUNITS)
CHANL(LOCAL+4) = THK(3)*CINCH(IUNITS)

C

165 CONTINUE
170 CONTINUE
175 CONTINUE
C

CONVEPT UNITS ON BC DATA

C

IF (IUNITS .EQ. 2) GO TO 300
NTBC = 1
DO 205 I = 2, 5
IF (BCTIME(I) .LE. 0.) GO TO 210
205 NTBC = NTBC + 1
210 NSETS = NBCS*NSLICE*NTBC
NSETP = NBCP*NSLICE*NTBC
C

DO 215 I = 1, NSETS
BCXS(I) = BCXS(I)*CINCH(1)
BCBGS(I) = BCBGS(I)*CHTC(1)
BCQGS(I) = BCQGS(I)*CHFLX(1)
BCPGS(I) = BCPGS(I)*CPRSR(1)

C

CONTINUE

79
DO 220 I = 1, NSETP
BCXP(I) = BCXP(I) * CINCH(I)
BCHGP(I) = BCHGP(I) * CHTC(I)
BCTGP(I) = BCTGP(I) * CTMPF(I) - 460.
BCQGP(I) = BCQGP(I) * CHFLX(I)
BCGP(I) = BCGP(I) * CPRSR(I)

C
220

C
NSET = NSTA * NTBC
DO 225 I = 1, NSET
RHOVG(I) = RHOVG(I) * CROVG(I)
THUBIN(I) = THUBIN(I) * CTMPF(I) - 460.
QHUBIN(I) = QHUBIN(I) * CHFLX(I)
TTIPIN(I) = TTIPIN(I) * CTMPF(I) - 460.

C
225

C
NSET = NSLICE * NTBC
DO 230 I = 1, NSET
PEX(I) = PEX(I) * CPRSR(IUNITS)

C
230

C
DO 235 I = 1, 49, 2
WPLEN = WPLEN * CMSFL(IUNITS)
RHOC = RHOC * CDEN(IUNITS)
RHOM = RHOM * CDEN(IUNITS)

C
235

C
RTIFIN(I) = RTIFIN(I) * CTMPP(I) - 460.
THIFIN(I) = THIFIN(I) * CTMPP(I) - 460.

C
280

C
DO 280 I = 1, 19, 2
AKCTBL(I) = AKCTBL(I) * CTMPP(IUNITS) - 460.
AKWTBL(I) = AKWTBL(I) * CTMPP(IUNITS) - 460.

C
280

C
CONTINUE
IF (IFILM .NE. 2) GO TO 320
DO 310 I = 1, NSTA
RHOVGA(I) = RHOVGA(I)

C
310

C
CONTINUE
IF (INEDIT .GT. 0), PRINT AN INPUT EDIT
DO 180 I = 1, NSLICE
CALL INPRF(I, INEDIT)
185 CONTINUE
RETURN
END

C-----SOURCE.NHCFRCT
FUNCTION HCFECC(IS, LCOOL, LIN)

C
C- SOURCE.NHCFRCT-----
C
COMMON /TCO/ ADUMP, BTA, CD, CP, SPAN, TOG,
Z G, RO, K, NAJET, DH (80), DH (80), DHJ (80),
Z W, WM, AKC(15, 80), AKW(15, 80),
Z BH(80), BH(80), A(400), A(400),
Z

NHCFRCT 1671
NHCFRCT 1672
NHCFRCT 1673
NHCFRCT 1674
NHCFRCT 1675
NHCFRCT 1676
NHCFRCT 1677
NHCFRCT 1678
NHCFRCT 1679
NHCFRCT 1680

80
I Z DLX (400), FF (80), HC (80), HG (80), NHCFRCT 1681
Z P (2, 15, 80), PEXIT (15), PUMP (80), QG (80), NHCFRCT 1682
Z QSNK (80), RR (80), S (15), T (2, 15, 400), NHCFRCT 1683
Z TG (80), TAU (400), WFC (80), NHCFRCT 1684
Z WJ (15, 80), WCROS (2, 15, 80), XN (80), NHCFRCT 1685
Z ICOR, IFILM, IHUB, ITIP, NHCFRCT 1686
Z ISBLOK, ISLICE, NBLKSZ, NSLICE, NHCFRCT 1687
Z NFWD, NSTA, IHC (80) NHCFRCT 1688

C COMPUTE TURBULENT HEAT TRANSFER COEFFICIENT IN CHANNEL FLOW:
C NU = .023*(RE**.8)*(PD**.333)
C
100 CONTINUE
C
TMP = (T(2, ISLICE, LCOOL) + T(2, ISLICE, LIN))/2.
CALL GASTBL (TMP, C, CP, GAM, PD, R, XMU)
EE = 12.3600.*ABS(WCROS(2, ISLICE, IS))*DH(IS)/(A(LCOOL)*XMU)
HCFRCD = .023*12.**(C/DH(IS))*(RE**.8)*(PD**.333)

200 CONTINUE
RETURN
END

C----SOURCE.NHCOOLT
SUBROUTINE HCOOLT(JS)

COMMON /IMPCOR/ CIMP1, CIMP2, CIMP3, CIMP4, CIMP5, CIMP6, CIMP7,
DIM1, DIM2, DIM3, DIM4, DIM5, DIM6
COMMON /PRPS/ CPO, GAMO, DP(80), SP(80), RE(80),
CPC(80), GAMC(80), DMR1(80), DMR2(80)
COMMON /TCO/ ADUMP, BTA, CD, CP,
GAM, EIM, R, SPAN, TOG,
WDUMP, WIM, AKC(15, 80), AKW(15, 80),
A(400), AJET(80), AM2(80), CNUM(80),
DH(80), DHF(80), DHJ(80),
DLX(400), FF(80), HC(80), HG(80),
QSNK(80), RR(80), S(15), T(2, 15, 400),
TG(80), TAU(400), WFC(80),
WJ(15, 80), WCROS(2, 15, 80), XN(80),
ICOR, IFILM, IHUB, ITIP,
ISBLOK, ISLICE, NBLKSZ, NSLICE,
NFWD, NSTA, IHC(80)

DIMENSION IGG(80), IRE(80), REJ(80), REJOVR(80)

1 CONTINUE
TMP=TOG
CALL GASTBL (TMP, C, CP, GAM, PD, R, XMU)
CONDCT =
XMUTOG = XMU
PI=3.14159
IF (JS.GT.1) GO TO 101
IF (ICOR.EQ.1) GO TO 101
IF (WJ(ISLICE, JS).LE.0.0) GO TO 101

C--- LEADING EDGE HEAT TRANSFER CORRELATION FOR STATIONS FORWARD OF ICORNHCOOLT
C--- CORRELATION OF METZGER ET AL, J. ENG. POWER, JULY 1969, PP 149-158 NHCOOLT 1740

81
NEND=ICOR+1
XS = 0.0
XP = 0.0
DO 50 J = 3,NEND,2
LINS = 5*(J-1) - 1
LINP = 5*J - 1
XS = XS + DLX(LINS)
XP = XP + DLX(LINP)
50 CONTINUE

XL = (XS + XP)/2.
IF (AJET(JS).GT.0.) GMASS = WJ(ISLICE,JS)/AJET(JS)
BES = PI*DHJ(JS)**2/(4.*XN(JS))
DEH = 2.*BES
REJ(JS) = 12.*3600.*GMASS*DLH/XMU
PROD = REJ(JS)**.27*(XL/BES)**.52
STANMX = .355/PROD
HC(JS) = STANMX*CP*GMASS*144.*3600.
C IF (REJ(JS).LT.1150..OR.REJ(JS).GT.6300.) WRITE(6,75) REJ(JS)
75 FORMAT(1H /****WARNING**** LEADING EDGE IMPINGEMENT JET REYNOLDS 'N
Z 'NUMBER IS ',F8.1/' RANGE OF THE CORRELATION IS 1150',-N
Z ' < REJ < 6300')
ILLED = ICOR - 1
IF (ILEAD.LT.2) GO TO 85
DO 80 I = 2,ILEAD
80 HC(I) = HC(JS)
85 CONTINUE
GO TO 101
90 WRITE(8,95) ICOR
95 FORMAT(/' SOLUTION TERMINATED*** TOO MANY ROWS OF IMPINGEMENT ',N
Z ' HOLEs FORWARD OF STATION',I3,'. HOLES ARE ',N
Z 'ALLOWED ONLY AT STATION 1. ')
STOP
C--KIRCHER-TABAKOFF CORRELATION, IMPINGEMENT WITH CROSS FLOW
C--ICOR = STATION NUMBER APPLICATION OF THIS CORRELATION BEGINS
C
101 IGGC = 0
IREC = 0
ISTRT=ICOR
IF (JS.GT.1) ISTRT= 1
C IF (CIMP1.NE.0.0) GO TO 400
DO 130 I = ISTRT,NFWD
WC = ABS(WCRS(2,ISLICE,I))
II = 5*I
REJ(I) = 0.0
IF (HIC(I).EQ.1) GO TO 103
LCOOL = 5*I
LIN = LCOOL - 1
HC(I) = HCFRCD(I,LCOOL,LIN)
GO TO 130
103 CONTINUE
IF (AJET(I).EQ.0.0) GO TO 128
IF (WJ(ISLICE,I).LE.0.) GO TO 128
TMP=(T(2,ISLICE,LIN)+T0)/2.
CALL GASTBL (TMP,C,CP,GAM,PD,R,XMU)
CONDCT = C

CONTINUE


105 REJ(I)=WJ(ISLICE,I)/AJET(I)*DHJ(I)/(XMUTOG/3600.)*12.0
GG=(WC(A(I)))/(WJ(ISLICE,I)/AJET(I))
IF (GG.LE.2.0) GO TO 110
IGGC = IGGC + 1
IGG(IGGC) = I
110 CONTINUE
IF (REJ(I).GE.300.0.AND.REJ(I).LE.3.04) GO TO 115
IREC = IREC + 1
IRE(IREC) = I
115 CONTINUE
IF (REJ(I).LT.3000.) GO TO 112
AM=-.002517*(XN(I)/DHJ(I))**2+.068485*XN(I)/DHJ(I)+.506994
HC(I) =REJ(I)**AM
HC(I) =HC(I)*EXP(.02596*(XN(I)/DHJ(I))**2-.8259*XN(I)/DHJ(I)+.3985)
HC(I) =HC(I)/(1.+4696)*
Z = ((WC(A(I))/WJ(ISLICE,I)/AJET(I)))*TAU(I)/DHJ(I)**.965)
GO TO 125
120 AM=.001852*(XN(I)/DHJ(I))**2+.042838*(XN(I)/DHJ(I))+.516548
HC(I) =REJ(I)**AM
HC(I) =HC(I)*EXP(.0126*(XN(I)/DHJ(I))**2-.5106*XN(I)/DHJ(I)-.2057)
HC(I) =HC(I)/(1.+4215)
Z = ((WC(A(I))/WJ(ISLICE,I)/AJET(I)))*TAU(I)/DHJ(I)**.58)
125 CONTINUE
HC(I) =HC(I)*CONDCT/DHJ(I)*1.2*PDTOG**.33*(TAU(I)/DHJ(I))**.091
GO TO 130
128 CONTINUE
IF (I.GT.2) HC(I) = HC(I-2)
IF (I.EQ.2) HC(I) = HC(1)
IF (I.EQ.1) HC(I) = HC(3)
130 CONTINUE
IST = NFWD + 1
DO 150 I = IST,NSTA,2
IF (IH(I).NE.1) GO TO 155
150 HC(I) = HC(I-2)
155 IST = NFWD*2
DO 160 I = IST,NSTA,2
IF (IH(I).NE.1) GO TO 165
160 HC(I) = HC(I-2)
165 CONTINUE
IF (IGGC.GT.0) WRITE(6,140) (IGG(I),I=1,IGGC)
DO 132 I = 1,IREC
ISTATN = IRE(I)
REJVR(I) = REJ(ISTATN)
132 CONTINUE
IF (IREC.GT.0) WRITE(6,145) (IRE(I),REJVR(I),I=1,IREC)
135 CONTINUE
140 FORMAT(1H" "***** WARNING ***** RATIO OF CROSSFLOW TO ",
Z "JET-FLOW IS OUT OF THE RANGE OF ",
Z "THE CORRELATION AT THE FOLLOWING STATIONS: '/23X,20(I4,','))")
145 FORMAT(1H" "***** WARNING ***** JET REYNOLDS NUMBER IS ",
Z "OUT OF THE RANGE OF THE CORRELATION ",
Z "AT THE FOLLOWING STATIONS: '/1X,8('**','I2,'--','-','F8.1,('*))")
DO 301 I = 1,NFWD
DUM22(I) = REJ(I)
301 CONTINUE
RETURN
83
C--- GENERAL CORRELATION FOR IMPINGEMENT WITH CROSSFLOW IS EVALUATED HERE.

C--- FORM OF CORRELATION IS:

\[ ST = \text{CIMP1} \times (\text{GG} \times \text{CIMP2}) \times (\text{GI} \times \text{CIMP3}) \times ((Z/D) \times \text{CIMP4}) \]

C--- WHERE GG IS THE MASS FLUX RATIO, FREE STREAM TO JET, AND

C--- GI IS THE MOMENTUM FLUX RATIO.

C

400 CONTINUE
DO 450 I = ISTRT,NFWD
WC = ABS(WCHOS(2,ISLICE,I))
II = 5*I
B0INV = P*TOG/(144.*P(2,ISLICE,I))
B0INV = B0*(2,ISLICE,I)/(144.*P(2,ISLICE,I))
REJ(I) = 0.0
IF (IHC(I) .EQ. 1) GO TO 403
LC00L = 5*I
LIN = LC00L - 1
HC(I) = HCFRCD(I,LC00L,LIN)
GO TO 450

403 CONTINUE
IF (AJET(I) .EQ. 0.0) GO TO 445
IF (WJ(ISLICE,I) .LE. 0.) GO TO 445
TMP = (T(2,ISLICE,LIN) + TOG)/2.
CALL GASTBL (TMP,C,CP,GAN,PD,R,XXU)
CONDCT = C
XMUTOG = XMU
PDTOG = PD

405 REJ(I) = WJ(ISLICE,I)/AJET(I) * DHJ(I) / (XMUTOG/3600.)*12.?
GG = (WC/A(I)) / (WJ(ISLICE,I)/AJET(I))
GI = (WC/A(I)) ** 2 * B0INV / (WJ(ISLICE,I)/AJET(I)) ** 2 * K0INVJ
Z0VERD = TAU(I)/DHJ(I)
Z0XV = XN(I)/DHI(J)
ST = CIMP1 * (GG ** CIMP2) * (GI ** CIMP3) * (Z0VERD ** CIMP4)
Z = (X0VERD ** CIMP5) * (REJ(I) ** CIMP6) * (PDTOG ** CIMP7)
HC(I) = 144.*3600.*ST*CP*WJ(ISLICE,I)/AJET(I)
GO TO 450

C

445 CONTINUE
IF (I.GT.2) HC(I) = HC(I-2)
IF (I.EQ.2) HC(I) = HC(I-1)
IF (I.EQ.1) HC(I) = HC(3)

450 CONTINUE
IST = NFWD+1
DO 460 I = IST,NSTA,2
IF (IHC(I) .NE. 1) GO TO 465
HC(I) = HC(I-2)

465 IST = NFWD+2
DO 470 I = IST,NSTA,2
IF (IHC(I) .NE. 1) GO TO 475
HC(I) = HC(I-2)

475 CONTINUE
C

485 CONTINUE
RETURN
END

C--- SOURCE, NHCPINT
SUBROUTINE HCPPINS(IS, DELTBN, LC00L, LCUP, LIN, LC00LP, PINS, ZFAREA)

84
COMPUTE THE HEAT TRANSFER COEFFICIENT AND EFFECTIVENESS FOR A TRIANGULAR ARRAY OF PIN PINS

WHERE DP IS PIN DIAMETER IN INCHES AND SP IS PIN SPACING IN INCHES

VDP = DP(IS)
VSP = SP(IS)

DIMENSION EFAREA(80), DELTAN(15)

VDP = RP(IS)
VSP = SP(IS)

CONTINUE

100 CONTINUE

TMP = (T(2,ISLICE,LCOOL) + T(2,ISLICE,LIN)) / 2.

CALL GASTBL(TMP,C,CP,GAO,P,GAM,P,MU)

C-- NO. OF PINS AT THIS STATION IS:

PINS = SPAN*DLX(LCOOL) / (3.6606*VSP**2)

C-- AVERAGE LENGTH OF PINS:

SLF = (TAU(LCOOL) + TAU(LCUP)) / 2.

IF (IS.GT.NFDU.AND.IS.LE.NFUD+1) SLP = TAU(LCOOL)

C-- MINIMUM FLOW AREA:

AMIN = SLP*SPAN*(VSP-VDP) / VSP

C-- TOTAL SURFACE AREA:

AHTR = 2.*DLX(LCOUL)*SPAN + 3.14159*PINS*(VDP*SLP-VDP**2/4.)

C-- CHANNEL HYDRAULIC DIAMETER:

DH(IS) = 4.*AMIN*DLX(LCOUL)/AHTR

REDH = 12.*3600.*ABS(WCROS(2,ISLICE,IS)) * DH(IS) / AMIN*MU

TERM1 = -0.89*(VSP/SLP)**0.5075

TERM2 = -3.094*VDP/VSP

TERM3 = 4.143*EXP(TERM1 + TERM2) /(REDH**.2946)

105 CONTINUE

HC(IS) = (12.*C/DH(IS)) * (.023 + TERM3) *(REDH**.5) *(PD**.333)

EML = SQRT((4.*HC(IS)*SLP**2)/(AKW(ISLICE,IS)*VDP))

EFTVNS = TANH(EML)/EML

C-- CHECK LOCATION OF HEAT FLOW SPLIT POINT IF THIS IS A TRAILING EDGE REGION STATION

C

IF (IS.LE.NFWD) GO TO 160

TBAR = (T(2,ISLICE,LCOOLP) - T(2,ISLICE,LCOUL)) / (T(2,ISLICE,LIN) - T(2,ISLICE,LCOUL))

HYCOS = COSH(EML)

HYSIN = SINH(EML)

IF (HYCOS-TBAR.LT.HYSIN) GO TO 120
WRITE(6,110) LCOOLP,LIN,DELTAN(ISLICE)
FORMAT(1H// ** WARNING ** NODE',I3, '
Z IS RECEIVING HEAT FROM NODE',I3,' THROUGH THE PINS.', 
Z ** RESULTS ARE INVALID. DELTAN = ',F7.4) 
GO TO 140
CONTINUE
IF (HYCOS-TEAR .GT. 0.) GO TO 130
WRITE(6,110) LCOOLP,LIN,DELTAN(ISLICE)
GO TO 140
CONTINUE
XOVRL = (HYCOS-TEAR)/BYSIN
XOVRL = ALOG((1.+XOVRL)/(1.-XOVRL))/(2.*EML)
CONTINUE
EFAREA(IS) = DLX(LIN)*SPAN
Z - 3.14159*PINS*(VDP**2/4.- EFTVNS*VDP*SLP*XOVRL) 
IF (IS.GT.NPWD) EFAREA(IS+l) = DLX(LCOOLP)*SPAN 
Z - 3.14159*PINS*(VDP**Z/4.- EFTVNS*VDP*SLP) 
CONTINUE
RETURN
END
C----SOURCE.NINPRTT
SUBROUTINE INPRT(IC HLNL,INEDIT)
COMMON /BOUND/ BCXS(400), BCXP(400), BCHGS(1000), BChGP(100C), NINPRTT 2009
Z BCTG5(1000), BCTGP(1000), BCGGS(1000), BCGGP(1000), NINPRTT 2010
Z BCPSG(1000), BCPSGP(1000), THUBIN(400), THUB(80), NINPRTT 2011
Z QHUBIN(400), QHUB(80), TTIPIN(400), TTIP(80), NINPRTT 2012
Z QTIPIN(400), QTIP(80), PHTVS(400), PEX(400), NINPRTT 2013
Z BCTIME(5C), TTO(50), PTO(50), WPLEN, NINPRTT 2014
Z WSYST(50), AKCTBL(20), AKWTBL(20), NBCS, NBCP NINPRTT 2015
COMMON /GAAS/ GS(200), NG NINPRTT 2016
COMMON /PRPS/ CPO, GAMA, DP(80), SP(80), RE(80), NINPRTT 2019
Z CPC(80), GAC(80), DUMR1(80), DUMR2(80) NINPRTT 2020
COMMON /RADL/ APLN(15), DPLN(15), RIN(15), ROUT(15), NINPRTT 2021
Z PIN(15), TIN(15), W(15), NS NINPRTT 2022
COMMON /SPECL/ CHANL(8000), TITLE(30), INDCMN(2000), NINPRTT 2024
Z IPLOT, MD1, MD2, MD3, IADJIN, IWRITE NINPRTT 2025
COMMON /TCO/ ADUMP, HTA, CD, CP, tumor, TOG, NINPRTT 2027
Z GAM, PIN, R, SPAN, NINPRTT 2028
Z WDUMP, WIR, AKC(15,80), AKW(15,80), NINPRTT 2029
Z A(400), AJET(80), AM(280), CNMU(80), NINPRTT 2030
Z DH(80), DHF(80), DHJ(80), NINPRTT 2031
Z DLX(400), FF(80), HC(80), HG(80), NINPRTT 2032
Z P(2,15,80), PENTX(15), PUMP(80), QG(80), NINPRTT 2033
Z QSNK(80), RR(80), S(15), T(2,15,400), NINPRTT 2034
Z TG(80), TAU(400), WPC(80), NINPRTT 2035
Z WJ(15,80), WC(2,15,80), XN(80), NINPRTT 2036
Z ICOR, IFILM, IUHUB, ITIP, NINPRTT 2037
Z ISBLOK, ISLICE, NBLKSZ, NSLICE, NINPRTT 2038
Z NFWD, NSTA, IHC(80), NINPRTT 2039
Z NFWD, NTRA, NINPRTT 2040
86
C INITIALIZE TEMPERATURE DISTRIBUTION (DEGREES R)

I = ICHNL
NODSF = 5*NPWD
NODSTM = 5*NSTA - 4
NODST = 5*NSTA
DO 830 I1 = 5,NODSF,5
IS = I1/5
LO = I1-4
LJ = I1-3
L = I1-2
LI = I1-1
T(2,I,LO) = .9*TG(IS)
T(2,I,LI) = T(2,I,LO)/1.08
T(2,I,LJ) = T(2,I,LO) - (T(2,I,LO)-T(2,I,LI))/TAU(LO)/(TAU(LO)+
Z TAUL) )

T(2,I,L) = T(2,I,LO) - (T(2,I,LO)-T(2,I,LI))*(TAU(LO)+TAUL)/2.)

930 T(2,I,1) = TTI0(1) + 460.

DO 860 I1 = ISTRT,NODSTM,10
T(2,I,1) = T(2,I,NODSF)
T(2,I,1+5) = T(2,I,1)
DO 960 J = 1,4
IPJ = I1 + J
IMJ = I1 + J - 5
IUFP = NODSF + J - 5
T(2,I,IPJ) = T(2,I,IUFP)

DO 965 J = 1,NODST
T(1,I,J) = T(2,I,J)

C IF (ICHNL.GT.1) GO TO 94
WRITE(6,408)
FORMAT(1H1,///,20X,'PROPERTY TABLES'///)
WRITE(6,410)
FORMAT(1H1,'OUTER COATING EFFECTIVE THERMAL CONDUCTIVITY')

C IF (IUNIT2.EQ.1) GO TO 420
WRITE(6,412) (AKCTBL(I),I=1,19,2)
WRITE(6,414) (AKCTBL(I),I=2,20,2)
412 FORMAT(//5X,'T, (F) ',10X,10(F9.1))
414 FORMAT(5X,'K, (BTU/HR/FT/R) ',10(F9.3))
416 FORMAT(//5X,'WALL METAL THERMAL CONDUCTIVITY */)
      WRITE(6,412) (AKCTBL(I),I=1,19,2)
      WRITE(6,414) (AKWTBL(I),I=2,20,2)
      GO TO 445
C
420 CONTINUE
      DO 418 I = 1,19,2
         AKCTBL(I) = (AKCTBL(I)+460.)/1.8
         AKWTBL(I) = (AKWTBL(I)+460.)/1.8
         AKCTBL(I+1) = AKCTBL(I+1)/CTCON(1)
      AKWTBL(I+1) = AKWTBL(I+1)/CTCON(1)
      WRITE(6,422) (AKCTBL(I),I=1,19,2)
      WRITE(6,424) (AKWTBL(I),I=2,20,2)
      CONTINUE
C
      DO 448 I = 1,19,2
         AKCTBL(I) = 1.8*AKCTBL(I) - 460.
         AKWTBL(I) = 1.8*AKWTBL(I) - 460.
         AKCTBL(I+1) = AKCTBL(I+1)*CTCON(1)
      AKWTBL(I+1) = AKWTBL(I+1)*CTCON(1)
      WRITE(6,450) (GS(J),J=1,NGS)
      WRITE(6,454) (GS(J),J=1,NGS)
      GO TO 470
C
      WRITE(6,452) (GS(J),J=1,NGS)
      FORMAT(//5X,'TEMPERATURE (F),10 (F9.1))
      L = NG + 1
      LE = NG + NGS
      WRITE(6,454) (GS(J),J=L,LE)
      L = 2*NG + 1
      LE = 2*NG + NGS
      WRITE(6,456) (GS(J),J=L,LE)
      L = 3*NG + 1
      LE = 3*NG + NGS
      WRITE(6,458) (GS(J),J=L,LE)
      L = 4*NG + 1
      LE = 4*NG + NGS
      WRITE(6,460) (GS(J),J=L,LE)
      GO TO 90
C
NINPRTT 2101
NINPRTT 2102
NINPRTT 2103
NINPRTT 2104
NINPRTT 2105
NINPRTT 2106
NINPRTT 2107
NINPRTT 2108
NINPRTT 2109
NINPRTT 2110
NINPRTT 2111
NINPRTT 2112
NINPRTT 2113
NINPRTT 2114
NINPRTT 2115
NINPRTT 2116
NINPRTT 2117
NINPRTT 2118
NINPRTT 2119
NINPRTT 2120
NINPRTT 2121
NINPRTT 2122
NINPRTT 2123
NINPRTT 2124
NINPRTT 2125
NINPRTT 2126
NINPRTT 2127
NINPRTT 2128
NINPRTT 2129
NINPRTT 2130
NINPRTT 2131
NINPRTT 2132
NINPRTT 2133
NINPRTT 2134
NINPRTT 2135
NINPRTT 2136
NINPRTT 2137
NINPRTT 2138
NINPRTT 2139
NINPRTT 2140
NINPRTT 2141
NINPRTT 2142
NINPRTT 2143
NINPRTT 2144
NINPRTT 2145
NINPRTT 2146
NINPRTT 2147
NINPRTT 2148
NINPRTT 2149
NINPRTT 2150
NINPRTT 2151
NINPRTT 2152
NINPRTT 2153
NINPRTT 2154
NINPRTT 2155
NINPRTT 2156
NINPRTT 2157
NINPRTT 2158
NINPRTT 2159
NINPRTT 2160
CONTINUE
DO 471 J = 1, NGS
WRITE(6,472) (DUM1(J), J=1, NGS)
FORMAT(//5X,'TEMPERATURE (K) ',10 (F9.1))
L = NG + 1
LE = NG + NGS
JI = 0
DO 473 J = L, LE
JI = JI + 1
DUM1(JI) = GS(J)/CTCON(1)
WRITE(6,474) (DUM1(JI), J=1, JI)
FORMAT(5X,'K, (W/M/K)',10 (F9.5))
L = 2*NG + 1
LE = 2*NG + NGS
JI = 0
DO 475 J = L, LE
JI = JI + 1
DUM1(JI) = GS(J)/CSPHT(1)
WRITE(6,476) (DUM1(JI), J=1, JI)
FORMAT(5X,'CP, (J/KG/K)',10 (F9.2))
L = 3*NG + 1
LE = 3*NG + NGS
WRITE(6,478) (GS(J), J=L, LE)
FORMAT(5X,'PRANDTL NUMBER',10 (F9.5))
L = 4*NG + 1
LE = 4*NG + NGS
JI = 0
DO 479 J = L, LE
JI = JI + 1
DUM1(JI) = GS(J)/CVISC(1)
WRITE(6,480) (DUM1(JI), J=1, JI)
FORMAT(5X,'VIS. (N S/M**2)',10 (F9.5))
C
CONTINUE
IF (INEDIT.EQ.0) GO TO 350
C-- LIST OUT THE INPUT HOT GAS BOUNDARY CONDITIONS--
C-- MNBC IS THE MAX OF (NBCS & NBCP)
C
MNBC = NBCS
IF (MNBC.LT.NBCP) MNBC=NBCP
NTIMES = 1
C-- NTIMES IS THE NUMBER OF TIME STEPS IN BC TABLES
481 IF (BCTIME(NTIMES+1).LE.0.0) GO TO 482
NTIMES = NTIMES + 1
GO TO 481
C
CONTINUE
WRITE(6,4820)
FORMAT(1H1,40X,'HOT GAS BOUNDARY CONDITIONS')/
WRITE(6,4830)
FORMAT(1H50X,'HOT GAS BOUNDARY CONDITIONS'/)
Z
'*************SUCTION SIDE**********************'
'*************PRESSURE SIDE**********************'
C
C-- SET THE NO. OF POINTS PER TIME STEP IN SEP BC ARRAYS
NPRTS = NSLICE*NBCS
NPRTP = NSLICE*NBCP

C
NL = 3
DO 499 IT = 1,NTIMES
C--SET THE NO. OF POINTS THAT PRECEDED TIME STEP 'IT'
NPCS = NPRCS*(IT-1)
NPRCP = NPRTP*(IT-1)
C--START THE LOOP THROUGH ALL SLICES
DO 499 IT = 1,NTIMES
C--SET THE NO. OF POINTS PRECEDING THIS SLICE
NBFRS = NPRCS + NBCS*(ISL-1)
NBFRP = NPRCP + NBCP*(ISL-1)

C
NL = NL + 3 + MNBC
IF (NL.LT.60) GO TO 4860
NL = 3 + MNBC
WRITE(6,4820)
WRITE(6,483)
CONTINUE
C
IF (IT.EQ.1) WRITE(6,484) ISL
484 FORMAT(45X,'INITIAL STEADY STATE',/49X,'SLICE NO.',I2)
IF (IT.GT.1) WRITE(6,485) BCTIME(IT), ISL
485 FORMAT(45X,'BCTIME = ',F8.3,' SEC',/47X,'SLICE NO.',I2)
WRITE(6,486)
486 FORMAT(45X,'Z',26X,'X',7X,'HG','6X','TG','10X','QG','6X','PG','
Z',26X,'X',7X,'HG','6X','TG','10X','QG','6X','FG')
C
C--HERE WE LOOP WITHIN A SLICE
C
DO 499 IBC = 1,MNBC
IF (IBC.GT.NBCS) GO TO 487
J = NBFRS + IBC
JXS = (ISL-1)*NBCS + IBC
TBCXS = BCXS(JXS)/CINCH(IUNITS)
TBCHGS = BCHGS(J)/CHTC(IUNITS)
TBCPGS = BCPG(J)/CPRSR(IUNITS)
WRITE(6,489) TBCXS,TBCHGS,TBCPGS
489 FORMAT(2X,F6.2,2F8.1,F12.1,F8.1)
CONTINUE
C
IF (IBC.GT.NBCP) GO TO 499
J = NBFRP + IBC
JXP = (ISL-1)*NBCP + IBC
TBCXP = BCXP(JXP)/CINCH(IUNITS)
TBCXG = BCHX(J)/CHTC(IUNITS)
TBCXG = BCPX(J)/CPRSR(IUNITS)
WRITE(6,490) TBCXG
490 FORMAT(66X,F6.2,2F8.1,F12.1,F8.1)
CONTINUE
C
IF (IBC.LE.NBCS) WRITE(6,488) TBCXP,TBCXG,TBCXP,TBCXG,TBCXP
IF (IBC.GT.NBCS) WRITE(6,490) TBCXP,TBCXG,TBCXP,TBCXG,TBCXP

CONTINUE
CONTINUE
DO 95 I = 1, 200
NFLUID(I) = 0
WRITE(6,150) ICHNL
IF (ICHNL.GT.1) GO TO 101
IF (IHUB.EQ.1) WRITE(6,142)
IF (IHUB.EQ.2) WRITE(6,144)
IF (IHUB.EQ.3) WRITE(6,146)
CONTINUE
IF (ICHNL.LT.NSLICE) GO TO 102
IF (ITIP.EQ.1) WRITE(6,147)
IF (ITIP.EQ.2) WRITE(6,148)
IF (ITIP.EQ.3) WRITE(6,149)
CONTINUE
TRIN = RIN(ICHNL)/CINCH(IUNITS)
TROUT = ROUT(ICHNL)/CINCH(IUNITS)
TAPLN = APLN(ICHNL)/(CINCH(IUNITS)*CINCH(IUNITS))
WRITE(6,103) TRIN, UL(IUNITS), TROUT, UL(IUNITS),
Z
TDPLN, UL(IUNITS), TAPLN, UA(IUNITS)
IF (IUNITS.EQ.1) GO TO 500
WRITE(6,153) NFWD, NSTB, SPAN
WRITE(6,155) CD, ADUMP
TEM = TTI0(I) + 460.
WRITE(6,157) TEM, PTO(I), PEX(ICHNL), WPLEN
ITFBG = NFWD + 2
WRITE(6,154) ICHNL, ITRBG
DO 118 I = 1, NSTA, 20
IF (IP18.GT.NSTA) IP18 = NSTA
IF (I.EQ.1) WRITE(6,156) (J,J=I,IP18,2)
IF (I.GT.1) WRITE(6,159) (J,J=I,IP18,2)
ID = 0
DO 104 J = I, IP18, 2
ID = ID + 1
DUM1(ID) = RF(J)
NFLUID(J) = 5*J
CONTINUE
WRITE(6,158) (NFLUID(J), J=I,IP18,2)
ID = 0
DO 116 J = I, IP18, 2
ID = ID + 1
NOS = NFLUID(J) - 4
IF (NOS.GT.7) GO TO 106
XOS = 0.0
XJN = 0.0
XMM = 0.0
XIS = 0.0
XCC = 0.0
GO TO 108
XOS = XOS + DLX(NOS)
XJN = XJN + DLX(NOS+1)
XMM = XMM + DLX(NOS+2)
XIS = XIS + DLX(NOS+3)
XCC = XCC + DLX(NOS+4)
CONTINUE
DUM2(ID) = XOS
DUM25(ID) = XJN
DUM3(ID) = XMM
DUM4(ID) = XIS
DUM5(ID) = XCC
DUM55(ID) = TAU(NOS)
DUM6(ID) = TAU(NOS+2)
NOS = NFLUID(J)
DUM7(ID) = TAU(NOS)
DUM8(ID) = A(NOS)
DUM9(ID) = DH(J)
DUM10(ID) = DHJ(J)
DUM11(ID) = CNUM(J)
DUM12(ID) = AJET(J)
DUM16(ID) = THUBIN(J) - 460.
DUM17(ID) = QHUBIN(J)
DUM18(ID) = TTIPTIN(J) - 460.
DUM19(ID) = QTIPIN(J)
NOS = NFLUID(J) - 4
DUM13(ID) = TG(J) - 460.
DUM14(ID) = HG(J)
JHCAL = IHC(J)
DUM15(ID) = HCAL(JHCAL)
IF (BTA.GT..Ol) WRITE(6,160) (DUM1(J),J=1,ID)
WRITE(6,162) (DUM2(J),J=1,ID)
WRITE(6,163) (DUM25(J),J=1,ID)
WRITE(6,164) (DUM3(J),J=1,ID)
WRITE(6,166) (DUM4(J),J=1,ID)
WRITE(6,168) (DUM5(J),J=1,ID)
WRITE(6,169) (DUM55(J),J=1,ID)
WRITE(6,170) (DUM6(J),J=1,ID)
WRITE(6,172) (DUM7(J),J=1,ID)
WRITE(6,174) (DUM8(J),J=1,ID)
WRITE(6,176) (DUM9(J),J=1,ID)
WRITE(6,178) (DUM10(J),J=1,ID)
WRITE(6,180) (DUM11(J),J=1,ID)
WRITE(6,182) (DUM12(J),J=1,ID)
WRITE(6,183) (DUM15(J),J=1,ID)
WRITE(6,184) (DUM13(J),J=1,ID)
IF (BTA.LT..Ol) WRITE(6,186) (DUM14(J),J=1,ID)
IF (BTA.GT..Ol) WRITE(6,188) (DUM14(J),J=1,ID)
IF (ICHNL.GT.1) GO TO 118
IF (IHB.EQ.1) WRITE(6,196) (DUM16(J),J=1,ID)
IF (IHB.EQ.3) WRITE(6,198) (DUM17(J),J=1,ID)
IF (ITIP.EQ.1) WRITE(6,202) (DUM18(J),J=1,ID)
IF (ITIP.EQ.3) WRITE(6,204) (DUM19(J),J=1,ID)
CONTINUE
ITRBG = NFWD + 1
WRITE(6,190) ICHNL,ITRBG
XOS = 0.0
XJN = 0.0
XMM = 0.0
XIS = 0.0
XCC = 0.0
DO 140 I = 2,NSTA,20
IP19 = I * 19
IF (IP18.GT.NSTA) IP18 = NSTA-1
IF (I.EQ.2) WRITE(6,156) (J,J=I,IP18,2)
IF (I.GT.2) WRITE(6,159) (J,J=I,IP18,2)
  ID = 0
  DO 122 J = I,IP18,2
  ID = ID + 1
  DUM1(ID) = RR(J)
  NFLUID(J) = 5*J
  CONTINUE
  WRITE(6,158) (NFLUID(J),J=I,IP18,2)
  ID = 0
  DO 130 J = I,IP18,2
  ID = ID + 1
  NOS = NFLUID(J) - 4
  XOS = XOS + DLX(NOS)
  XJN = XJN + DLX(NOS+1)
  XM = XM + DLX(NOS+2)
  XIS = XIS + DLX(NOS+3)
  XCC = XCC + DLX(NOS+4)
  DUM2(ID) = XOS
  DUM3(ID) = XM
  DUM4(ID) = XIS
  DUM5(ID) = XCC
  DUM55(ID) = TAU(NOS)
  DUM6(ID) = TAU(NOS+2)
  NOS = NFLUID(J)
  DUM7(ID) = TAU(NOS)
  DUM8(ID) = A(NOS)
  DUM9(ID) = DH(J)
  DUM10(ID) = DHJ(J)
  DUM11(ID) = CNUM(J)
  DUM12(ID) = AJet(J)
  DUM16(ID) = THUSBIN(J) - 460.
  DUM17(ID) = QHUBIN(J)
  DUM18(ID) = TTIPIN(J) - 460.
  DUM19(ID) = QTIPIN(J)
  NOS = NFLUID(J) - 4
  DUM13(ID) = TG(J) - 460.
  DUM14(ID) = HG(J)
  JHCAL = IHC(J)
  DUM15(ID) = HHCAL(JHCAL)
  IF (BTA.GT.0.1) DUM14(ID) = QG(J)
  CONTINUE
  WRITE(6,160) (DUM1(J),J=1,ID)
  WRITE(6,162) (DUM2(J),J=1,ID)
  WRITE(6,163) (DUM25(J),J=1,ID)
  WRITE(6,164) (DUM3(J),J=1,ID)
  WRITE(6,166) (DUM4(J),J=1,ID)
  WRITE(6,168) (DUM5(J),J=1,ID)
  WRITE(6,169) (DUM55(J),J=1,ID)
  WRITE(6,170) (DUM6(J),J=1,ID)
  WRITE(6,172) (DUM7(J),J=1,ID)
  WRITE(6,174) (DUM8(J),J=1,ID)
  WRITE(6,176) (DUM9(J),J=1,ID)
  WRITE(6,178) (DUM10(J),J=1,ID)
  WRITE(6,180) (DUM11(J),J=1,ID)
  WRITE(6,182) (DUM12(J),J=1,ID)
  WRITE(6,183) (DUM15(J),J=1,ID)
  WRITE(6,184) (DUM13(J),J=1,ID)
  IF (BTA.LT.0.1) WRITE(6,186) (DUM1(J),J=1,ID)
  IF (BTA.GT.0.1) WRITE(6,188) (DUM14(J),J=1,ID)
IF (ICHNL.GT.1) GO TO 140
IF (IHUB.EQ.1) WRITE(6,196) (DUM16(J),J=1,ID)
IF (IHUB.EQ.3) WRITE(6,198) (DUM17(J),J=1,ID)
IF (ITIP.EQ.1) WRITE(6,202) (DUM18(J),J=1,ID)
IF (ITIP.EQ.3) WRITE(6,204) (DUM19(J),J=1,ID)
CONTINUE

FORMAT(1H1,46X,'INPUT FOR SLICE NUMBER',I3)
FORMAT(21X,' HUB TEMPERATURES ARE SPECIFIED')
FORMAT(21X,' ADIABATIC HUB SPECIFIED')
FORMAT(21X,' HUB HEAT FLUX IS SPECIFIED')
FORMAT(21X,' TIP TEMPERATURES ARE SPECIFIED')
FORMAT(21X,' ADIABATIC TIP SPECIFIED')
FORMAT(21X,' TIP HEAT FLUX IS SPECIFIED')
FORMAT(1H2//' NUMBER OF STATIONS IN IMPINGEMENT REGION IS',I3)
FORMAT(' IMPINGEMENT HOLE DISCHARGE COEF.=',F6.3)
FORMAT(' AREA OF DUMP TO TRAILING EDGE =',F8.5,' in**2')
FORMAT(' COOLANT INLET TEMP.=',F7.1,' R, COOLANT INLET PRESSURE=',
FORMAT(' EXIT PRESSURE =',F6.1,' psia, COOLANT FLOW =',F6.1,' lbm/hr)'
FORMAT(1H2//' PRESSURE SIDE, SLICE ',I2,' TRAILING EDGE REGION ',
FORMAT(' BEGINNING AT STATION-',I3)
FORMAT(1H2// STATION NUMBER'//5X,10 (6X, I4))
FORMAT(' COOLANT NODE NUMBER'//5X,10 (6X, I4))
FORMAT(1H2// RADIAL LOCATION(IN)',10F10.3)
FORMAT(1H2// X, OUTSIDE SUB.(IN)',10F10.5)
FORMAT(1H2// X, INTERFACE (IN)',10F10.5)
FORMAT(1H2// X, MID-METAL (IN)',10F10.5)
FORMAT(1H2// X, INSIDE SURF. (IN)',10F10.5)
FORMAT(1H2// X, MID.COO.L.CH. (IN)',10F10.5)
FORMAT(' COATING THKNESS (IN)',10F10.3)
FORMAT(' WALL THICKNESS (IN)',10F10.5)
FORMAT(' CHANNEL WIDTH (IN)',10F10.5)
FORMAT(' CHANNEL AREA(IN**2)',10F10.5)
FORMAT(' CHANNEL HYD.DIA(IN)',10F10.5)
FORMAT(' NO. OF IMP. JETS',10F10.2)
FORMAT(' TOT.JET AREA(IN**2)',10F10.5)
FORMAT(' TYPE OF HC CALC.',10F6.4)
FORMAT(' OUTSIDE BC: TG,(F)',10F10.1)
FORMAT(' HG (BTU/HR/FT**2/B)',10F10.1)
FORMAT(' GG (BTU/HR/FT**2)',10F10.1)
FORMAT(1H1// SUCTION SIDE, SLICE ',I2,' TRAILING EDGE REGION ',
FORMAT(' BEGINS AT STATION-',I3)
FORMAT(1H2// CLAD K(BTU/HR/FT)**',10F10.3)
FORMAT(1H2// METL K(BTU/HR/FT)**',10F10.3)
FORMAT(1H2// GIVEN HUB TEMP. (F)',10F10.1)
FORMAT(1H2// QUHUB (BTU/HR/FT**2)',10F10.1)
FORMAT(1H2// GIVEN TIP TEMP. (F)',10F10.1)
FORMAT(1H2// QTIP (BTU/HR/FT**2)',10F10.1)
FORMAT(1H2//' TOTAL NUMBER OF STATIONS IS',I3)
FORMAT(' SPANC = SPAN/CINCH(1)
ADUMP = ADUMP/(CINCH(1)**2)
WRITE(6,553) NFWD,NSTA,SPANC
WRITE(6,555) CD,ADUMP
TEM = (TPIO(1) + 460.)/1.8
PTIOC = PTIO(1)/CPBSPR(1)
PEXC = PEX(ICHNL)/CPBSPR(1)
WPLENC = WPLEN/CMSFL(1)
WRITE(6,557) TEN,PTIOC,PEXC,WPLENC
ITRBG = NFWD + 2
WRITE(6,154) ICHNL,ITRBG
DO 518 I = 1,NSTA,20
IP18 = I + 18
IF (IP18.GT.NSTA) IP18 = NSTA
IF (I.EQ.1) WRITE(6,556) (J,J=I,IP18,2)
IF (I.GT.1) WRITE(6,559) (J,J=I,IP18,2)
ID = 0
DO 504 J = I,IP18,2
ID = ID + 1
DUM1(ID) = RR(J)/CINCH(1)
NFLUID(J) = 5*J
504 CONTINUE
WRITE(6,558) (NFLUID(J),J=I,IP18,2)
ID = 0
DO 516 J = I,IP18,2
ID = ID + 1
NOS = NFLUID(J) - 4
IF (NOS.GT.1) GO TO 506
XOS = 0.0
XJN = 0.0
XMM = 0.0
XIS = 0.0
XCC = 0.0
GO TO 508
506 XOS = XOS + DLX(NOS)/CINCH(1)
XJN = XJN + DLX(NOS+1)/CINCH(1)
XMM = XMM + DLX(NOS+2)/CINCH(1)
XIS = XIS + DLX(NOS+3)/CINCH(1)
XCC = XCC + DLX(NOS+4)/CINCH(1)
508 CONTINUE
DUM2(ID) = XOS
DUM25(ID) = XJN
DUM3(ID) = XMM
DUM4(ID) = XIS
DUM5(ID) = XCC
DUM55(ID) = TAU(NOS)/CINCH(1)
DUM6(ID) = TAU(NOS+2)/CINCH(1)
NOS = NFLUID(J)
DUM7(ID) = TAU(NOS)/CINCH(1)
DUM8(ID) = A(NOS)/(CINCH(1)**2)
DUM9(ID) = DH(J)/CINCH(1)
DUM10(ID) = DHJ(J)/CINCH(1)
DUM11(ID) = CNUM(J)
DUM12(ID) = AJET(J)/(CINCH(1)**2)
DUM16(ID) = THUBIN(J)/1.8
DUM17(ID) = QHUBIN(J)/CHFLX(1)
DUM18(ID) = TIPIN(J)/1.8
DUM19(ID) = QTIPIN(J)/CHFLX(1)
NOS = NFLUID(J) - 4
DUM13(ID) = TG(J)/1.8
DUM14(ID) = HG(J)/CHFLX(1)
JHCAL = IHC(J)
DUM15(ID) = HCAL(JHCAL)
IF (BTA.GT.0.01) DUM14(ID) = QG(J)/CHFLX(1)
516 CONTINUE
WRITE (6,560) (DUM1(J), J=1,ID)
WRITE (6,562) (DUM2(J), J=1,ID)
WRITE (6,563) (DUM25(J), J=1,ID)
WRITE (6,564) (DUM3(J), J=1,ID)
WRITE (6,566) (DUM4(J), J=1,ID)
WRITE (6,568) (DUM5(J), J=1,ID)
WRITE (6,569) (DUM55(J), J=1,ID)
WRITE (6,570) (DUM6(J), J=1,ID)
WRITE (6,572) (DUM7(J), J=1,ID)
WRITE (6,574) (DUM8(J), J=1,ID)
WRITE (6,576) (DUM9(J), J=1,ID)
WRITE (6,578) (DUM10(J), J=1,ID)
WRITE (6,580) (DUM11(J), J=1,ID)
WRITE (6,582) (DUM12(J), J=1,ID)
WRITE (6,583) (DUM13(J), J=1,ID)
WRITE (6,584) (DUM14(J), J=1,ID)
IF (BTA.LT.0.1) WRITE(6,586) (DUM14(J), J=1,ID)
IF (BTA.GT.0.1) WRITE(6,588) (DUM14(J), J=1,ID)
IF (ICHNL.GT.1) GO TO 518
IF (IHUB.EQ.1) WRITE (6,596) (DUM16(J), J=1,ID)
IF (IHUB.EQ.3) WRITE (6,598) (DUM17(J), J=1,ID)
IF (ITIP.EQ.1) WRITE (6,602) (DUM18(J), J=1,ID)
IF (ITIP.EQ.3) WRITE (6,604) (DUM19(J), J=1,ID)
CONTINUE

ITRBG = NFWD + 1
WRITE (6,190) ICHNL, ITRBG
XOS = 0.0
XJN = 0.0
XMM = 0.0
XIS = 0.0
XCC = 0.0
DO 540 I = 2,NSTA,20
IP18 = I + 18
IF (IP18.GT.NSTA) IP18 = NSTA-1
IF (I.EQ.2) WRITE (6,556) (J, J=I,IP18,2)
IF (I.GT.2) WRITE (6,559) (J, J=I,IP18,2)
ID = 0
DO 522 J = I,IP18,2
ID = ID + 1
DUM1(ID) = RR(J)/CINCH(1)
NFLUID(J) = 5*J
CONTINUE
WRITE (6,558) (NFLUID(J), J=I,IP18,2)
ID = 0
DO 530 J = I,IP18,2
ID = ID + 1
NOS = NFLUID(J) - 4
XOS = XOS + DLX(NOS)/CINCH(1)
XJN = XJN + DLX(NOS+1)/CINCH(1)
XMM = XMM + DLX(NOS+2)/CINCH(1)
XIS = XIS + DLX(NOS+3)/CINCH(1)
XCC = XCC + DLX(NOS+4)/CINCH(1)
DUM2(ID) = XOS
DUM25(ID) = XJN
DUM3(ID) = XMM
DUM4(ID) = XIS
DUM5(ID) = XCC
DUM55(ID) = TAU(NOS)/CINCH(1)
DUM6(ID) = TAU(NOS+2)/CINCH(1)
NOS = NFLUID(J)
DUM7(ID) = TAU(NOS)/CINCH(1)
DUM8(ID) = A(NOS)/(CINCH(1)**2)
DUM9(ID) = DH(J)/CINCH(1)
DUM10(ID) = DJ(J)/CINCH(1)
DUM11(ID) = CNUM(J)
DUM12(ID) = A(JET(J))/CINCH(1)**2
DUM13(ID) = TG(J)/CINCH(1)
DUM14(ID) = DJ(J)/CINCH(1)
DUM15(ID) = HCAL(JHCAL)
DUM16(ID) = THUBIN(J)/1.8
DUM17(ID) = QHUBIN(J)/CHFLX(I)
DUM18(ID) = TTIPIN(J)/1.8
DUM19(ID) = QTIPIN(J)/CHFLX(1)
NOS = NFLUID(J) - 4

CONTINUE

WRITE(6,560) (DUM1(J),J=1,ID)
WRITE(6,562) (DUM2(J),J=1,ID)
WRITE(6,563) (DUM25(J),J=1,ID)
WRITE(6,564) (DUM3(J),J=1,ID)
WRITE(6,566) (DUM4(J),J=1,ID)
WRITE(6,568) (DUM5(J),J=1,ID)
WRITE(6,569) (DUM55(J),J=1,ID)
WRITE(6,570) (DUM6(J),J=1,ID)
WRITE(6,572) (DUM7(J),J=1,ID)
WRITE(6,574) (DUM9(J),J=1,ID)
WRITE(6,576) (DUM9(J),J=1,ID)
WRITE(6,578) (DUM10(J),J=1,ID)
WRITE(6,580) (DUM11(J),J=1,ID)
WRITE(6,582) (DUM12(J),J=1,ID)
WRITE(6,583) (DUM15(J),J=1,ID)
WRITE(6,584) (DUM13(J),J=1,ID)
IF (BTA.LT.0.1) WRITE(6,586) (DUM14(J),J=1,ID)
IF (BTA.GT.0.1) WRITE(6,588) (DUM14(J),J=1,ID)
IF (ICHNL.GT.1) GO TO 540
IF (IHUB.EQ.1) WRITE(6,596) (DUM16(J),J=1,ID)
IF (ITIP.EQ.1) WRITE(6,602) (DUM18(J),J=1,ID)
IF (ITIP.EQ.3) WRITE(6,604) (DUM19(J),J=1,ID)
CONTINUE

FORMAT('/NUMBER OF STATIONS IN IMPINGEMENT REGION IS',I3)
Z ' ', TOTAL NUMBER OF STATIONS IS',I3,
Z ' ', SPAN OF THIS SLICE IS',F6.3,' CM',
Z ' ', AREA OF DUMP TO TRAILING EDGE =',F6.3,' CM**2',
Z ' ', COOLANT INLET TEMP.=',F7.1,' K, COOLANT INLET',
Z ' ', PRESSURE =',F7.1,' KPA, EXIT PRESSURE =',
Z ' ', COATING THKNESS =',F6.1,' KG/HR',
Z ' ', WALL THICKNESS (CM)',10P10.3
Z ' ', RADIAL LOCATION(CM)',10P10.3
Z ' ', OUTSIDE SUR. (CM)',10P10.5
Z ' ', INTERFACE (CM)',10P10.5
Z ' ', MID-METAL (CM)',10P10.5
Z ' ', INSIDE SURF. (CM)',10P10.5
Z ' ', MID.COOI.CH. (CM)',10P10.5
Z ' ', COATING THKNESS (CM)',10P10.5
Z ' ', WALL THICKNESS (CM)',10P10.5

553 FORMAT(' NUMBER OF STATIONS IN IMPINGEMENT REGION IS',I3)
554 FORMAT(' SPAN OF THIS SLICE IS',F6.3,' CM',
555 FORMAT(' AREA OF DUMP TO TRAILING EDGE =',F6.3,' CM**2',
556 FORMAT(' COOLANT INLET TEMP.=',F7.1,' K, COOLANT INLET',
557 FORMAT(' PRESSURE =',F7.1,' KPA, EXIT PRESSURE =',
558 FORMAT(' COATING THKNESS =',F6.1,' KG/HR',
559 FORMAT(' WALL THICKNESS (CM)',10P10.3
560 FORMAT(' RADIAL LOCATION(CM)',10P10.3
561 FORMAT(' OUTSIDE SUR. (CM)',10P10.5
562 FORMAT(' INTERFACE (CM)',10P10.5
563 FORMAT(' MID-METAL (CM)',10P10.5
564 FORMAT(' INSIDE SURF. (CM)',10P10.5
565 FORMAT(' MID.COOI.CH. (CM)',10P10.5
566 FORMAT(' COATING THKNESS (CM)',10P10.5
567 FORMAT(' WALL THICKNESS (CM)',10P10.5

97
C----SOURCE,NPARAYT
SUBROUTINE PARRAY(JS,JSENS,ICHOKE)
C COMMON/MATRX/ TCOF(400,30)
C COMMON/PRPS/ CPC(80), GAMC(80), DP(80), SP(80), RE(80)
Z CPC(80), GAMC(80), DUMR1(80), DUMR2(80)
C COMMON/TCO/ ADUMP, BTA, CD, CP, SPAN, TOG
Z GAM, PIM, R, CPC(15,80), AKW(15,80)
Z A(400), AJET(80), AN2(80), CNUM(80)
Z DH(80), DHP(80), DHJ(80)
Z DLX(400), FF(80), HC(80), HG(80)
Z P(2,15,80), PEXIT(15), PUMP(80), QG(80)
Z QSNK(80), RR(80), S(15), T(2,15,400)
Z TG(60), TAU(400), WFC(80)
Z WJ(15,80), WCROS(2,15,80), DX(80)
Z ICR, IFILM, IHUB, ITIP
Z ISBLOK, ISLICE, NBLKSZ, NSLICE
Z NFWD, NSTA, IHC(80)
C COMMON/TRNSNT/ RHOC, RHOM, SPHTC, SPHTM
Z DLTIME, TME, TEPS, TYMXAX
C DIMENSION POLD(80), PSAV(5)
C COMPUTE NEW PRESSURES
C IFNL = THE NUMBER OF FLOW CHANNEL NODES
C TREPS = 1.0
IF (TIME.GE.0.) TREPS = TEPS
800 IFNL = NSTA - 3
NODST = 5*NSTA
NODSF = 5*NPFD
C INITIALIZE COEFFICIENT ARRAY TO 0.0
C DO 810 I = 1,IPNL
DO 810 J = 1,30
810 TCOF(I,J) = 0.0
C COMPUTE THE COEFFICIENT VALUES
C DO 900 I = 1,IPNL
FILM = 0.0
820 TCOF(1,J) = 0.0
C FOR THE IMPINGEMENT REGION:
C ICHK = 0 IMPLIES I IS EVEN AND STATION IS ON SUCTION SIDE
C = 1 IMPLIES THAT I IS ODD AND STATION IS ON PRESSURE SIDE
C DEFINE THE REAL NODE NUMBER IN TERMS OF I
C WHERE IRL IS THE PIVOTAL ELEMENT = COOLANT NODE NUMBER, LC00L
C IDN = DOWNSTREAM COOLANT NODE
C IUP = UPSTREAM COOLANT NODE
C IF ( I.LT.NFWD ) GO TO 840
IF ( I.EQ.NFWD ) GO TO 890
830 CONTINUE
GO TO 860
C FOR I=NFWD, THE NODE IS THE ENTRANCE TO THE TRAILING EDGE AND IS
C TREATED SEPARATELY AT (890)
C FOR I>NFWD, THE NODE IS IN THE TRAILING EDGE AND IRL IS DEFINED AS:
C IF (ICHK.GT.0) GO TO 855
I1L = 5*I
IDN = IRL + 10
IDNS = I+2
IUP = IRL
IUPS = IDNS - 2
ITC = 10
ITCP = 12
830 CONTINUE
GO TO 860
C 840 CONTINUE
IRL = 5*I
IF (I.GT.JS) GO TO 843
C IF (I.LT.JS) GO TO 852
C IF (ICHK.GT.0) GO TO 849
C GO TO 855
C 843 IF (ICHK.GT.0) GO TO 849
C STATION I IS SUCTION SIDE, DOWNSTREAM OF SPLIT POINT
C IUPS = I - 2
IDNS = I
IUP = IRL - 10
IDN = IRL
ITC = 8
ITCP = 10
IF (I.GT.2) GO TO 860
IUPS = 1
IUP = 5
ITC = 9
GO TO 860

C C STATION I IS PRESSURE SIDE, DOWNSTREAM OF SPLIT POINT
C
849 CONTINUE
IUPS = I
IDNS = I + 2
IUP = IRL
IDN = IRL + 10
ITC = 10
ITCP = 8
IF (I.GT.2) GO TO 860
IDNS = 1
IDN = 5
ITCP = 9
GO TO 860

852 CONTINUE
IF (ICHK.GT.0) GO TO 858
IF (1CHK.NE.JSENS) GO TO 846
855 CONTINUE
C C I IS ON SUCTION SIDE, FORWARD OF SPLIT POINT
C
IUPS = I
IDNS = I - 2
IUP = IRL
IDN = IRL - 10
IDX = IUP
ITC = 10
ITCP = 8
IF (I.GT.2) GO TO 860
IDNS = 1
IDN = 5
ITCP = 9
GO TO 860

858 CONTINUE
IF (ICHK.NE.JSENS) GO TO 849
IDNS = I
IUPS = I + 2
IUP = IRL
IDN = IRL + 10
IDX = IUP
ITC = 12
ITCP = 10

860 CONTINUE
C
TTRTM = 0.0
IF (DLTYME.GT.0.0.AND.TYME.GE.0.) TTRTM = 12.*DLX(IDN) *
Z (WCROS(2,ISLICE,IDNS)-WCROS(1,ISLICE,IDNS))/(DLTYME*A(IUP)*32.2)
WPCDUM = WFC(IDNS)
IF (I.GT.NFWD) WPCDUM = WPCDUM + WFC(IDNS+1)
IF(WCROS(2,ISLICE,IDNS).NE.0.0) FILM = WPCDUM/WCROS(2,ISLICE,IDNS)
TCOP(I,ITC) = TREPS*
Z (((1.0 + GAMC(IUPS)*AM2(IUPS)) + (A(IDN)-A(IUP))/(2.*A(IUP)))* 
TCOP(I,ITC) = TREPS*(-(1.0 + .5*gamc(IDNS)*AM2(IDNS))* 
Z (4.*FF(IDNS)*DLX(IDX)/DH(IDNS)+2.*FILM)*A(IDN)/A(IUP) + 
Z + (A(IDN)-A(IUP))/(2.*A(IUP))))

100
ROOT = SQRT(32.2*GAMC(IDNS)*R*T(2,ISLICE,IDNS)*AM2(IDNS))

PUMTRM = 0,0
IF (ROOT.NE.0.0) PUMTRM = (3.14159265*WS/30.)**2*
Z = RR(IDNS)*/(R(RR(IDNS)-RR(IUPS))*WCROS(2,ISLICE,IDNS))
Z = (A(IUP)*ROOT*144.*32.2)
TCOF(I,20) = PUMTRM + TRTRM - (1.-TREPS)*
Z = (P(1,ISLICE,IUPS)*TCOF(I,ITC) + P(1,ISLICE,IDNS)*TCOF(I,ITCP))
CONTINUE
IF (IDS.NE.ICHOKE) GO TO 880

TCOF(I,20) = -PEXIT(ISLICE)*TCOF(I,12) + TCOF(I,20)
TCOF(I,12) = 0.0
CONTINUE
C FOR TRAILING EDGE CHANNELS:
C IF (I.LT.IFNL) GO TO 900
C TCOF(I,20) IS NON-ZERO ONLY FOR I=IFNL
C IF (ICHOKE.EQ.NSTA-1) GO TO 900
C TCOF(I,20) = -PEXIT(ISLICE)*TCOF(I,12) + TCOF(I,20)
C TCOF(I,12) = 0.0
C CONTINUE
C FOR A PRESSURE SIDE, TRAILING EDGE REGION STATION, COOLANT NODE IS IDENTICAL TO SUCTION SIDE NODE.
C TCOF(I,10) = 1.0
TCOF(I,9) =-1.0
TCOF(I,20) = 0.0
CONTINUE
C FOR THE SPECIAL NODE AT THE ENTRANCE TO THE TRAILING EDGE:
C ALLOWING FOR THE POSSIBILITY OF ADDITION OF EXTRA COOLING AIR INTO TRAILING EDGE,
C TRTRM = 0.0
IF (DLTYN.E.GT.0.0.AND.TYME.GE.0.) TRTRM=12.*DLX(NFWD+1) *
Z = (WCROS(2,ISLICE,NFWD+1)-WCROS(1,ISLICE,NFWD+1))/
Z = (DLTYN-A(NODSF+5)*32.2)
AVRGA = (A(NODSF-5) + A(NODSF) - A(NODSF+5))/(3.*A(NODSF+5))
TCOF(I,9) = TREPS*((1. + GAMC(NFWD-1)*AM2(NFWD-1)) *
Z = A(NODSF-5)/A(NODSF+5) - AVRGA)
TCOF(I,10) = TREPS*((1. + GAMC(NFWD)*AM2(NFWD)) *
Z = A(NODSF)/A(NODSF+5) - AVRGA)
IF (WCROS(2,ISLICE,NFWD+1).NE.0.0) PILM =
Z = (WFC(NFWD+1)+WFC(NFWD+2))/WCROS(2,ISLICE,NFWD+1)
TCOF(I,11) = TREPS*(-1. - GAMC(NFWD+1)*AM2(NFWD+1)) *
Z = (1. + 2.*FF(NFWD+1)*DLX(NODSF+5)/DH(NFWD+1)*PILM - AVRGA)
C PUMP(NFWD+1) = (3.14159265*WS/30.)**2*
Z = RR(NFWD+1)*/(RR(NFWD+1)-RR(NFWD))
ROOT = SQRT(32.2*GAMC(NFWD+1)*R*
Z = T(2,ISLICE,NODSF+5)*AM2(NFWD+1))
PUMTRM = 0.0
IF (ROOT.NE.0.0) PUMTRM = (3.14159265*WS/30.)**2*RR(NFWD+1)*
Z = (RR(NFWD+1)-RR(NFWD))*WCROS(2,ISLICE,NFWD+1)
Z \cdot \frac{A(NODSF+5) \cdot \text{ROOT} \cdot 144. \cdot 32.2}{(A(NODSF+5) \cdot \text{ROOT} \cdot 144. \cdot 32.2)}

C PUMP HAS UNITS OF (IN**2/SEC**2): ROOT HAS UNITS OF (FT/SEC)

C DUMT = 0.0
Z IF (ADUMP.GT.0.) DUMT = - WDUMP**2*R*
Z \frac{(P(1,ISLICE,NFWD-5)+P(1,ISLICE,NFWD)) \cdot ADUMP \cdot A(NODSF+5) \cdot 32.2}{(P(1,ISLICE,NFWD-1)+P(l,ISLICE,NFWD)) \cdot ADUMP \cdot A(NODSF+5) \cdot 32.2}

C CONTINUE
Z RETURN
C ENDC---- SOURCE.NPLENMP
SUBROUTINE PLNUH(WXX,PXX,PTEXIT, TXX,TTEXIT)

A SUBROUTINE TO COMPUTE PRESSURE DROP IN THE CENTRAL COOLANT PLENUM

COMMON /RADL/ APLN(15), DPLN(15), RIN(15), ROUT(15),
Z FIN(15), TIN(15), W(15), WS
C COMMON /TCO/ ADUMP, ETA, CD, CP,
Z GAM, PIM, R, SPAN, TUG,
Z GAM, PIM, R, SPAN, TUG,
Z A(400), AJET(80), AM2(80), CNUM(80),
Z DH(80), DHF(80), DJH(80),
Z DLX(400), FP(8C), HC(80), HG(80),
Z P(2,15,80), PEXIT(15), PUMP(80), QG(80),
Z QSNK(80), RR(80), S(15), T(2,15,400),
Z TG(80), TAU(400), WFC(80),
Z WJ(15,80), WCR(2,15,80), XH(80),
Z ITCR, IFILM, IHUB, ITIP,
Z ISBLOK, ISLICE, NSLKSZ, NSLICE,
Z NFWD, NSTA, IHG(80)
C COMMON /TRNSNT/ RHOC, RHON, SPHTC, SPHTN,
Z DLYME, TME, TEE, TTMAX
C COMMON /UNITS/ CINCH(2), CHTC(2), CHFLX(2), CPRSR(2), CMSFL(2)
**CTUPF (2), CTCNP (2), CDEN (2), CSPHT (2), CGASC (2), CTCON (2), CVISC (2), CRHOVG (2), IUNITS**

**Z**

**DIMENSION BETTA (20), B (20), AMC (20), SIGMA (20), T1 (20), P1 (20), Z1 (15), CH (15)**

**DIMENSION SV (3), XK (4), XL (4)**

--- **PUNP (PP) IS THE EQUATION FOR DELTA P OVER LENGTH DX**

**PUNP (PP) =**

\[
\frac{D1*RRP*(PP/R/TP-V1)}{144.0-2.*PP*G2*TP/PP/AA/DD} Z
\]

\[
(1.-TP*G2/(PP*AA)**2+G1*CP)
\]

\[
Z *778.161*TP*TP*V1/(PP*AA)**2/(PP)**DX
\]

--- **PUN' (XK) IS THE EQUATION FOR DELTA T OVER LENGTH DX**

**PUN' (XK) =**

\[
\]

**DX**

--- **SAVE INLET TOTAL PRESSURE (PSIA) IN PIN AND INLET TOTAL TEMPERATURE (F) IN TIN**

**PIN (ISLICE) =**

**TIN (ISLICE) =**

**CH (ISLICE) =**

**ZED =**

**IF (ZED.EQ.0.) ZED = 0.001**

**IF (TIN (ISLICE) .GT.-430.0) GO TO 5**

**TIN (ISLICE) = 50.0**

**SIGB = 0.0**

**NSTNS = 4**

**SEGMTS = NSTNS - 1**

**T1 = TIN (ISLICE) + 460.0**

**B (1) = T1**

**BETA1 = PIN (ISLICE)**

**DX = S (ISLICE) / SEGMTS**

**XXN = NSTNS**

**DR = (ROUT (ISLICE) - RIN (ISLICE)) / SEGMTS**

--- **COMPUTE CONSTANT TERMS-C1-C8**

**CONTINUE**

**TW = B (1)**
CALL GASTBL(TTX,C,CP,GAM,PD,R,XMU)
J=1
C6=.5*(GAM-1.0)
C1=GAM/C6
C
IF (WS) 21,19,21
C NO PUMPING
19 C3=0.0
GO TO 23
C
C PUMPING
C
21 C3=2.36695E-6*(WS**2)/(C1*R)
23 C8=32.17*GAM*R
C5=1.0/SQRT(C8)
C7=1.0/(32.17*C1*R)
IF (J.GT.1) GO TO 33
25 CONTINUE
C
C COMPUTE CHANNEL REYNOLDS NO. IF J = 1
C
REY = 12.0*W(ISLICE)/XMU*DPLN(ISLICE)/APLN(ISLICE)
C
C COMPUTE FRICTION FACTOR
C
C COMPUTE 2 TERMS
33 CONTINUE
Z3=12.0*W(ISLICE)/XMU
Z4=(R*W(ISLICE)/3600.0)**2
IF (J.GT.1) GO TO 77
C
C DETERMINE INLET CONDITIONS
C
C INITIAL STATION COMPUTATIONS -
C
C BALANCING ON TOTAL PRESSURE -
C
39 NAG=-1
41 SIGC=(B(J)/APLN(ISLICE))**2*Z4/BETTA(J)
IF (ABS(SIGB-SIGC) .LE. 0.00001*SIGC) GO TO 57
C SIGMA NOT CONVERGED
43 IS=IS+1
SV(IS)=SIGC
IF (IS.EQ.3) GO TO 135
45 B(J)=T1-C7*SIGC
IF (B(J).LT.50.0) GO TO 159
C TEMP OK
47 SIGB=SIGC
BETTA(J)=BETAL*(B(J)/T1)**C1
GO TO 41
C SIGMA CONVERGED
57 B(J)=T1-C7*SIGC
AMC(1)=SQRT(SIGC/B(J))*C5
IF (B(J).LE.0.0) GO TO 159
KTRBZ=0
63 BETTA(J)=BETAL*(1.0+C6*AMC(1)**2)**C1
IF (BETTA(J).LE.0.0) GO TO 159
65 B(1)=T1/(1.0+C6*AMC(1)**2)
```plaintext
SIGMA(1) = (B(J)/APLN(ISLICE))**2*Z4/BETTA(J)
SIGC = SQRT(SIGMA(1)/B(1)) * C5
IF (ABS(SIGC-AMC(1)) .LE. .01) GO TO 71

AMC(1) = SIGC
KTRBZ = KTRBZ + 1
IF (KTRBZ .LE. 20) GO TO 63

TT1(1) = TIN(ISLICE)
C CHANNEL PRESSURE DROP -

73
NAG = 1
F1(1) = REY
SIGMA(2) = SIGMA(1) * .95
J = 2
IS = 0
CONTINUE
AJ = J - 1
DR2 = DR * (2.0*(RIN(ISLICE) * AJ * DB) - DB)

75
TT1(J) = TT1(J-1) + C3 * DR2
KSIG = 0
AZ = DPLN(ISLICE)/APLN(ISLICE)
REY = Z3 * AZ
PP = SQRT(BETTA(1))
SIGMA(1) = SQRT(SIGMA(1))
BETTA(1) = PP
TP = B(1)
G2 = (W(ISLICE)/3600.0)**2*R/32.174
G1 = G2/CP/778.16
D1 = (WS*3.1415927/30.0)**2*DR/DX/32.174
D2 = D1/778.16/144.0
RRP = RIN(ISLICE)
DO 97 J = 2, NSTNS
AZ = DPLN(ISLICE)/APLN(ISLICE)
REY = Z3 * AZ
F1(J) = 0.079*REY**(-.25)
IF (REY.LT.2300.) F1(J) = 16.0/REY
PTEMP = BETTA(J-1)
TTEMP = B(J-1)
RTEMP = RRP
MACH1 = 1
XNN = 2.0
DD = DPLN(ISLICE)
AA = APLN(ISLICE)
GO TO 85

81
MACH1 = XNN
WRITE(6,83) ISLICE, J
83 FORMAT(5X,9H****,50HDREEASED INCREMENT DERIVATIVE CHANGING
1 TOO FAST,3X,'BRANCH NO. ',I2,*, STATION NO. ',I2/)
XNN = XNN*2.0
DX = DX/2.0
PTEMP = BETTA(J-1)
TTEMP = B(J-1)
PP = PTEMP
TP = TTEMP
RRP = RTEMP
DD = DPLN(ISLICE)
AA = APLN(ISLICE)
DO 91 L = 1,4
V1 = G1/PP/AA**2/(1.0+G1*R*TP/(PP*AA)**2)
TERM1 = TP*G2/(PP*AA)**2
```
TERM2 = G1 * CP * 778.16 * R * TP * TP * V1 / (PP * AA) ** 2 / PP

TESTMA = 1.0 - TERM1 + TERM2

IF (TESTMA .LE. 0.0) GO TO 159

XK(L) = FUNP(P1(J))

IF (L .EQ. 1) GO TO 89

DO 87 LL = 2, L

XTEST = ABS((XK(L) - XK(LL-1)) / PP)

IF (XTEST .GT. DIFTOL) GO TO 81

CONTINUE

XL(L) = PUNP(XK(L))

IF (L .EQ. 4) GO TO 93

PP = PTEMP + XK(L) / 2.0

TP = TTEMP + XL(L) / 2.0

IF (L .EQ. 2) GO TO 91

RRP = RRP + DR / XNN

IF (L .NE. 3) GO TO 91

PP = PTEMP + XK(L)

TP = TTEMP + XL(L)

IF (PP .LE. 0.0 OR. TP .LE. 0.0) GO TO 159

V1 = G1 / PP / AA ** 2 / ((1.0 + G1 * R * TP) / (PP * AA) ** 2)

TERM1 = PP * G2 / (PP * AA) ** 2

TERM2 = G1 * CP * 778.16 * R * TP * TP * V1 / (PP * AA) ** 2 / PP

TESTMA = 1.0 - TERM1 + TERM2

IF (TESTMA .LE. 0.0) GO TO 159

BETTA(J) = PTEMP + (XK(1) + 2.0 * (XK(2) + XK(3)) + XK(4)) / 6.0

B(J) = TTEMP + (XL(1) + 2.0 * (XL(2) + XL(3)) + XL(4)) / 6.0

PP = BETTA(J)

IF (B(J) .LE. 0.0 OR. BETTA(J) .LE. 0.0) GO TO 159

MACH1 = MACH1 - 1

PTEMP = PP

TTEMP = TP

GO TO 85

CONTINUE

C ALL STATIONS COMPUTED

C RESTART CHOKE BRANCH IF M.LT. 6

IF (CH(ISLICE) .EQ. 0.0 OR. CH(ISLICE) .EQ. 1.0) GO TO 113

IF (ACH .EQ. (-1.0)) GO TO 113

ACH = -1

AB = 0

DO 109 J = 1, NSTNS

IF (ACH(J) .LT. AB) GO TO 109

AB = ACH(J)

CONTINUE
CR = 0

AJ = (GAM + 1.0) / 2.0

CX = - (GAM + 1.00) / (GAM - 1.0) / 2.0

AZ = .95 / AB * (1.0 + C6/90) ** (C1) * (1.0 + C6/AB ** 2) ** (-CX)

WCOK=E=W(ISLICE)

W(ISLICE) = AZ * W(ISLICE)

WRITE(6, 111) ISLICE, W(ISLICE), WCOK

111 FORMAT (8X, 6H* ****, 23H RESTART CHOKED BRANCH , 15, 24H FLOW RATE IN)

GO TO 177

113 BETA = PIN(ISLICE) ** 2

C CALCULATE THE CHOKING FLOW RATE

AB = 0.0

DO 115 J = 1, NSTNS

IF (AHC(J) LT AB) GO TO 115

AB = AUC(J)

115 CONTINUE

AJ = (GAM + 1.0) / 2.0

CX = - (GAM + 1.0) / (GAM - 1.0) / 2.0

AZ = .95 / AB * (1.0 + C6/90) ** (C1) * (1.0 + C6/AB ** 2) ** (-CX)

C COMPUTE RESISTANCE EQUATION FOR BALANCE

PT1 = (1.0 + C6 * AHC(1) ** 2) ** (C1/2.0) * BETA(1)

IF (C3 NE 0.0) GO TO 117

GO TO 119

117 DFR2 = ROUT(ISLICE) ** 2 - RIN(ISLICE) ** 2

119 PEXIT = PT1 * (1.0 + C3/T1 + D2R) ** (C1/2.0)

121 CONTINUE

Z1(ISLICE) = (PEXTX2 - BETA(NSTNS) ** 2) / W(ISLICE) ** 2

IF (Z1(ISLICE).GT.0.0) GO TO 129

WRITE(6, 125) ISLICE, Z1(ISLICE)

125 FORMAT (8X, 6H* *****, 23H PASSAGE *' , 13, 5X,' HAS NEGATIVE OR NO RESISTANCE'

Z = F12.4/)

Z1(ISLICE) = ZED

129 CONTINUE

PP = BETA(NSTNS) * (1.0 + C6/AMC(NSTNS) ** 2) ** (C1/2.0)

PEXIT = PP

DIFTO = 0.005

KTR1 = 0

C COMPUTE AVERAGE STATIC PRESSURE AND STATIC TEMPERATURE

C

PXX = 0.0

TXX = 0.0

DO 134 I = 1, NSTNS

TXX = TXX + B(I)

134 PXX = PXX + BETA(I)

TXX = TXX / XXN - 460.0

PXX = PXX / XXN

TTEXIT = T1(NSTNS)

RETURN

C COMPUTE ACCELERATION

C

135 D = SV(2) - SV(1)

137 D = (SV(3) - SV(2)) / D

E = ABS(D) - 1.0

IF (ABS(D) .GT. 0.6) GO TO 143

139 E = D / (D - 1.0)

141 SIGB = E * SV(2) + (1.0 - E) * SV(3)

143 TS = 0

107
145 SIGC=SIGB
147 KSIG=KSIG+1
149 WRITE (6,151) ISLICE, J
151 FORMAT(7X,28H PROGRAM IS LOOPING IN BRANCH, I6, 9H STATION, I4//)
153 ISLICE=220
155 CONTINUE
157 STOP

C

CHOKING ADJUSTMENT
C

159 WCHOKE=W(ISLICE)
161 IF (CH(ISLICE).EQ.0.) GO TO 165
163 W(ISLICE)=-.98*W(ISLICE)
165 AB=0.
167 CONTINUE
169 CONTINUE
W(ISLICE)=1600.*APLN(ISLICE)*SQRT((32.17*BLTA1*GAM)/(RT1))
171 CH(ISLICE)=1.0
173 WRITE(6,175) ISLICE
175 FORMAT(3X,12H*** PASSAGE ,I6, 23H HAS CHOKED AT STATION ,I5,
177 134H AND THE FLOW HAS BEEN REDUCED TO ,F10.4, 6H PEOM ,F10.4, 4H ***)
179 GO TO 17
181 WRITE (6,183) ISLICE
183 FORMAT(/2X,16H***FLOW IN BRANCH, I6,
185 Z HAS BEEN REDUCED 50 TIMES BECAUSE OF CHOKING*)
END

C----SOURCE.NPLOTIT
SUBROUTINE PLOTMF(ALPH2)
C
C SOURCE.NPLOTIT
C

108
COMMON /PRPS/ CPO, GAMO, DP(80), SP(80), RE(80), NPL0T1T 3361
C
COMMON /SPEC1/ CHANL(6000), TITLE(30), INDCHN(2000), NPL0T1T 3362
C
COMMON /TCO/ ADUMP, ETA, CD, CP, NPL0T1T 3363
Z
GAM, PIM, R, SPAN, TOG, NPL0T1T 3364
Z
WDUMP, WM, AKC(15, 80), AKW(15, 80), NPL0T1T 3365
Z
A(400), AJET(80), AM2(80), CNUM(80), NPL0T1T 3366
Z
DH(80), DHP(80), DHJ(80), NPL0T1T 3367
Z
DLX(400), FF(80), HC(80), HG(80), NPL0T1T 3368
Z
P(2, 15, 80), PEXIT(15), PUMP(80), QG(80), NPL0T1T 3369
Z
QSMK(80), RR(80), S(15), T(2, 15, 400), NPL0T1T 3370
Z
IG(80), TAU(400), WFC(80), NPL0T1T 3371
Z
WE(15, 80), WCROS(2, 15, 80), XN(80), NPL0T1T 3372
Z
ICOR, IFILM, NPL0T1T 3373
Z
ICOR, IHUB, ITP, NPL0T1T 3374
Z
ICOR, IFILM, NPL0T1T 3375
Z
ICOR, IHUB, ITP, NPL0T1T 3376
Z
ICOR, IFILM, NPL0T1T 3377
Z
ICOR, IHUB, ITP, NPL0T1T 3378
Z
ICOR, IFILM, NPL0T1T 3379
Z
ICOR, IHUB, ITP, NPL0T1T 3380
Z
ICOR, IFILM, NPL0T1T 3381
Z
ICOR, IHUB, ITP, NPL0T1T 3382
Z
ICOR, IFILM, NPL0T1T 3383
Z
ICOR, IHUB, ITP, NPL0T1T 3384
C
COMMON /INNSNT/ RHOC, RHOM, SPHTC, SPHTM, NPL0T1T 3385
Z
DLTME, TYME, TPS, NPL0T1T 3386
Z
DLTME, TYME, TPS, NPL0T1T 3387
Z
DLTME, TYME, TPS, NPL0T1T 3388
C
DIMENSION Y (320), XLABL (29), YLABL (7), TLABL1 (21), TLABL2 (9), NPL0T1T 3389
C
DIMENSION XLABL2 (15), VARIB (15), YPLAB1 (10), ALPH2 (4), XLABL (7), NPL0T1T 3390
C
DIMENSION XS (80), XP (60), TSO (500), TPM (500), TPO (500), NPL0T1T 3391
C
DIMENSION YLABL2 (15), YPLAB1 (10), XLABL2 (20), NPL0T1T 3392
C
DIMENSION YTEM (80), SYPLAB (2), YLABL2 (80), NPL0T1T 3393
C
LOGICAL*1 I XAX/. TRUE./, IYF. X/. PALS E./, NPL0T1T 3394
C
INTEGER*2 NPTS, NPL0T1T 3395
C
DIMENSION RTNARR (2), VARS (12), SYPIBL (10), YLABL (20), NPL0T1T 3396
C
DIMENSION PLEGN (5), SLEGY (5), SYPLAB (10), YLABL (20), NPL0T1T 3397
C
DATA PLEGN/'X') ', 'PRES', 'SURE', 'SID', 'E'/, NPL0T1T 3398
C
DATA SLEGY/'B') ', 'SUCT', 'ION', 'SID', 'E'/, NPL0T1T 3399
C
DATA XLABL2/15*, NPL0T1T 3400
C
DATA SYMBOL/'O'/, NPL0T1T 3401
C
DATA XLABL/'YID-', NPL0T1T 3402
C
DATA VARIB/'PRES', 'SURE', 'SUCT', 'ION', 'SURE', 'ACE', 'MID-', NPL0T1T 3403
C
109
ALABL(7) = PLT Yale (2)

C

MD2 > 0 INDICATES JOB IS COMPLETE.--NOW DO SUMMARY PLOTS.
C

10 CONTINUE

IF (MD2.GT.0) GO TO 80

20 NSTAPS = NSTA/2 + 1

NLBL5 = NSTAPS/5

C

C SET UP TIME AND DATE LABEL FOR PLOT IDENTIFICATION
C

ALABL(1) = ALPH2(3)
ALABL(2) = ALPH2(4)
ALABL(3) = ALPHZ(1)
ALABL(4) = ALPH2(2)

C

C SET UP TITLE
C

DO 45 I = 1,30

IF (I.LE.21) TLABL1(I) = TITLE(I)

IF (I.GT.21) TLABL2(I-21) = TITLE(I)

45 CONTINUE

C

C PRESSURE SIDE
C

46 IF (MD3.GT.1) GO TO 55

47 XP(1) = 0.0

IX = 1

DO 50 I = 3,NSTA,2

50 XP(IX) = XP(IX-1) + DLX(NMM)/CINCH(IUNITS)

XPL = XP(NSTAPS)

DO 51 I = 2,NSTAPS

51 XP(1) = XP(I)/XPL

55 CONTINUE

IY = 0

ITP = NSTAPS*(ISLICE-1)

DO 60 I = 1,NSTA,2

IY = IY + 1

NOS = 5*I - 4

Y(IY) = T(2,ISLICE,NOS)/CTMPF(IUNITS)

IF (IUNITS.EQ.2) Y(IY) = Y(IY) - 460.

ITP = ITP + 1

TPO(ITP) = Y(IY)

IYP = IY + NSTAPS

Y(IYP) = T(2,ISLICE,NOS+1)/CTMPF(IUNITS)

IF (IUNITS.EQ.2) Y(IYP) = Y(IYP) - 460.

IYP = IYP + 2*NSTAPS

Y(IYP) = T(2,ISLICE,NOS+2)/CTMPF(IUNITS)

IF (IUNITS.EQ.2) Y(IYP) = Y(IYP) - 460.

TPH(ITP) = Y(IYP)

IYP = IY + 3*NSTAPS

Y(IYP) = T(2,ISLICE,NOS+3)/CTMPF(IUNITS)

IF (IUNITS.EQ.2) Y(IYP) = Y(IYP) - 460.

IYP = IY + 4*NSTAPS

NCOOL = NOS + 4

110
\[ Y(IYP) = \frac{T(2, ISLICE, NC0OL)}{CTMPF(IUNITS)} \]

\[ \text{IF (IUNITS.EQ.2)} \ Y(IYP) = Y(IYP) - 460. \]

\[ \text{IF (IUNITS.EQ.2)} \ XLABL(16) = \text{VARIB(1)} \]

\[ \text{XLABL(17) = VARIB(2)} \]

\[ \text{IF (IUNITS.EQ.2)} \ G3 \text{ TO 601} \]

\[ \text{DO 611 I = 1,15} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]

\[ \text{C} \]
IVARS(4) = ISYM(I)
YST = 1 + (I-1)*NSTAPS
YEN = I*NSTAPS
III = 0
DO 602 II = YST,YEN
III = III + 1
602 YTEM(III) = Y(II)
CALL GLOT(XP,YTEM,IVARS)
603 CONTINUE
CALL DISPLA(1)

PRESSURE SIDE COOLANT PRESSURE DISTRIBUTION

IY = 0
DO 61 I = 1,NSTA,2
IY = IY + 1
61 Y(IY) = F(I+1,SLICE,I) * Z (1. + (AMC(I) - 1.)*AM2(I)/2.)* (AMC(I)/(AMC(I) - 1.))/CS (IUNITS)
CALL CHARS(84,TLABL1,0.0,0.15,9.85,12)
CALL CHARS(36,TLABL2,0.0,0.15,9.85,12)
CALL CHARS(60,XLABL,0.0,1.5,25,12)
CALL CHARS(56,XLABL2,0.0,1.5,0.05,12)
CALL CHARS(40,YLABL,90,2,8,12)
MD3 = MD3 + 1
CALL NUMER1(1,MD3,4,0,ALABL(5))
CALL CHARS(28,ALABL1,0.0,6.2,1.25,12)
CALL SCALE (IXAX,NPTS,XP)
CALL SCLBAK(IYBX,NPTS,Y,KTNARF)
CALL GINTVL(RTNARR(1),RTNARR(2),YMIN,YMAX)
VARS(1) = YMIN
VARS(2) = YMAX
VARS(3) = 1.0
VARS(4) = 1.0
VARS(5) = 1.0
VARS(6) = 1.0
VARS(7) = 1.0
CALL XAXIS(.8,.6,VARS)
VAX(2) = 8.9
VAX(3) = 90.
VAX(4) = YMIN
VAX(5) = YMAX
VAX(6) = YMIN
VAX(7) = 1.0
C--- AXES ARE SET, NOW PLOT THE PRESSURE
IVARS(1) = 4
IVARS(2) = NSTAPS
IVARS(3) = 2
IVARS(4) = 65
CALL GLOT(XP,Y,IVARS)
CALL DISPLA(1)

SUCTION SIDE

IF (MD3.GT.2) GO TO 69
XS(1) = 0.0
IX = 1
DO 65 I = 2,NSTA,2
CALL CHARS(28,YLABL,90.,25,3.3,12)
MD3 = MD3 + 1
CALL NUMBER(1,MD3,4,0,YLABL(5))
CALL CHARS(28,YLABL,0.0,6.2,9.3,12)

C----- TITLES ARE DONE, NOW SET UP AXES FOR TEMPERATURE PLOTS
C
NPTS = NSTAPS
CALL SCALE(IYAX,NPTS,XS)
NPTS = 5*NSTAPS
CALL SCBLAK(IYAX,NPTS,Y,RTNARR)
CALL GINTVL(RTNARR(1),RTNARR(2),10,1,YMIN,YMAX)
VARS(1) = 7.0
VARS(2) = 9.0
VARS(3) = 0.0
VARS(4) = 0.0
VARS(5) = 1.0
VARS(6) = .5
VARS(7) = 1.0
CALL XAXIS(.8,.6,VARS)
VARS(2) = 8.9
VARS(3) = 90.
VARS(4) = YMIN
VARS(5) = YMAX
CALL YAXIS(.8,.6,VARS)

C----- AXES ARE SET. NOW PLOT THE FIVE TEMPERATURE CURVES, USING
C DIFFERENT SYMBOLS FOR EACH.
C
IVARS(1) = 4
IVARS(2) = NSTAPS
IVARS(3) = 2
DO 703 I = 1,5
IVARS(4) = ISYM(I)
IYST = 1 + (I-1)*NSTAPS
IYEN = I*NSTAPS
III = 0
DO 702 II = IYST,IYEN
III = III + 1
702 YTEM(III) = Y(II)
CALL GPLOT(XS,YTEM,IVARS)
703 CONTINUE
CALL DISPLA(1)

C
C SUCTION SIDE COOLANT PRESSURE DISTRIBUTION
C
Y(1) = P(2,ISLICE,1)*
Z(1+(GAMC(1)-1.)*AM2(1)/2.)*GAMC(1)/(GAMC(1)-1.))/CPRSR(IUNITS)
IY = 1
DO 75 I = 2,NSTA,2
IY = IY + 1
75 Y(IY) = P(2,ISLICE,I)*(1+(GAMC(I)-1.)
Z*AM2(I)/2.)*GAMC(I)/(GAMC(I)-1.))/CPRSR(IUNITS)
CALL CHARS(84,XLABL,.0,.0,.15,9.85,12)
CALL CHARS(36,XLABL,.0,.0,.15,9.65,12)
CALL CHARS(60,XLABL,.0,.1,.5,.25,12)
CALL CHARS(56,XLABL,.0,.1,.5,.05,12)
CALL CHARS(40,YPLABL,.9,.25,2.8,12)
MD3 = MD3 + 1
CALL NUMBER (1, MD, 4, 0, ABL (5))
CALL CHARs (28, ABL, 0, 0, 6, 2, 9, 9, 12)
NPTS = NSTAPS
CALL SCALE (IXAX, NPTS, XS)
CALL SCLBAK (IXAX, NPTS, Y, RTNARR)
CALL GINTVL (RTNARR (1), RTNARR (2), 10, 1, 1MIN, YMAX)
VARS (1) = 7.0
VARS (2) = 9.0
VARS (3) = 0.0
VARS (4) = 0.0
VARS (5) = 1.0
VARS (6) = .5
VARS (7) = 1.0
CALL XAXIS (.8, .6, VARS)
VARS (2) = 8.9
VARS (3) = 90.
VARS (4) = YMIN
VARS (5) = YMAX
CALL YAXIS (-8, 1.6, 10)

C --- AXES ARE SET, NOW PLOT THE PRESSURE

IVARS (1) = 4
IVARS (2) = NSTAPS
IVARS (3) = 2
IVARS (4) = 65
CALL GPLOT (XS, Y, IVARS)
CALL DISPLA (1)
GO TO 150

C THE FOLLOWING SECTION PUTS OUT PLOTS CONTAINING TEMPERATURES FROM
C ALL SLICES ON ONE FRAME
C
80 CONTINUE
   IF (ISLICE. LT. NSLICE) GO TO 150
C
C THE FOLLOWING PUTS TWO PLOTS ON ONE FRAME OF FILM
C
C FIRST PLOT THE OUTSIDE SURFACE TEMPERATURES FOR EACH
C SLICE ON THE SAME PLOT
C
NPTS = NSTAPS * NSLICE
CALL SCLBAK (IXAX, NPTS, TPC, RTNARR)
TMAXP = RTNARR (2)
TMNP = RTNARR (1)
CALL SCLBAK (IXAX, NPTS, TSO, RTNARR)
TMAXS = RTNARR (2)
TMINS = RTNARR (1)
IF (TMAXS .GT. TMAXP) TMAXP = TMAXS
IF (TMINS .LT. TMNP) TMNP = TMINS
NINTRV = (TMAXP - TMNP) / 100. + 2
CALL GINTVL (TMNP, TMAXP, NINTRV, 0, ATMINP, ATMAXP)
AINTRV = NINTRV
CALL CHARS (84, TLABL1, 0.0, 0.15, 9.85, 12)
CALL CHARS (36, TLABL2, 0.0, 0.15, 9.65, 12)
YLABL2 (5) = VARIb (5)
YLABL2 (6) = VARIb (6)
CALL CHARS (44, YLABL2, 90., 25, 1.6, 12)
VARS (1) = 8.
VARS (2) = 8.5
VARS(3) = 0.
VARS(4) = 0.0
VARS(5) = 1.0
VARS(6) = .5
VARS(7) = 1.0
VARS(8) = 0.0
CALL XAXIS(1.2,5.5,VARS)
MD3 = MD3 + 1
CALL NUMBER(1,MD3,4,0,ALABL(5))
CALL CHARS(28,ALABL,0.0,1.3,9.5,12)
VARS(1) = 7.
VARS(2) = 3.8
VARS(3) = 90.
VARS(4) = ATMINP
VARS(5) = ATMAXP
VARS(6) = AINTRV
VARS(7) = 1.
CALL YAXIS(1.2,5.5,VARS)
DO 100 I = 1,NSLICE
JST = NSTAPS*(I-1)
DO 95 J = 1,NSTAPS
Y(J) = TPO(JST+J)
SYMBOL = SYMBL(1)
KS = 0
C
C--- LABEL EVERY 10TH POINT WITH THE SLICE NUMBER, TO
C IDENTIFY THE CURVES.
KSTART = I + 1
DO 98 K = KSTART,NSTAPS,10
KS = KS + 1
XLBL(KS) = XP(K)
YLBL(KS) = Y(K)
IVARS(1) = 6
IVARS(2) = NLBLS
IVARS(3) = 3
IVARS(4) = 240*I
IVARS(5) = 1
IVARS(6) = 8
CALL G PLOT(XLBL,YLBL,IVARS)
IVARS(1) = 3
IVARS(2) = NSTAPS
IVARS(3) = 0
100 CALL G PLOT(XP,Y,IVARS)
C
VARS(1) = 7.
VARS(2) = 8.5
VARS(3) = 0.
VARS(4) = 0.0
VARS(5) = 1.0
VARS(6) = .5
VARS(7) = 1.0
CALL XAXIS(1.2,5.5,VARS)
CALL CHARS(42,XLABL,0.0,3.0,4.0,5.12)
VARS(1) = 7.
VARS(2) = 3.8
VARS(3) = 90.
VARS(4) = ATMINP
VARS(5) = ATMAXP

116
VAES(6) = ALNTEV
VARS(7) = 1.
CALL YAXIS(1.2, .5, VARS)
CALL CHARS(20, SLEGN, 0.0, 6.0, 4.5, 12)
DO 110 I = 1, NSLICE
JST = NSTAPS*(I-1)
DO 105 J = 1, NSTAPS
Y(J) = TSO(JST+J)
SYMBOL = SYMBOL(I)
KS = 0

C--- LABEL EVERY 10TH POINT WITH THE SLICE NUMBER, TO
C IDENTIFY THE CURVES.
KSTART = I + 1
DO 108 K = KSTART, NSTAPS, 10
KS = KS + 1
XLBL(KS) = XS(K)
YLBL(KS) = Y(K)
IVARS(1) = 6
IVARS(2) = NLBLS
IVARS(3) = 3
IVARS(4) = 240+I
IVARS(5) = 1
IVARS(6) = 8
CALL GPLOT(XLBL, YLBL, IVARS)
IVARS(1) = 3
IVARS(2) = NSTAPS
IVARS(3) = 0
110 CALL GPLOT(XS, Y, IVARS)
CALL DISPLAY(1)

C CONTINUE
CALL SCLBAK(IYAX, NPTS, TFM, RTNARR)
TMAXP = RTNARR(2)
TMINP = RTNARR(1)
CALL SCLBAK(IYAX, NPTS, TSM, RTNARR)
TMAXS = RTNARR(2)
TMINS = RTNARR(1)
IF (TMAXS.GT.TMAXP) TMAXP = TMAXS
IF (TMINS.LT.TMINP) TMINP = TMINS
NINTRV = (TMAXP-TMINP)/100. + 2
CALL GINTVL(TMINP, TMAXP, NINTRV, 0, ATMINP, ATMAXP)
AINTRV = NINTRV
CALL CHARS(84, TLBL1, 0.0, 0.15, 9.85, 12)
CALL CHARS(36, TLBL2, 0.0, 0.15, 3.65, 12)
YLABL2(5) = VARIB(7)
YLABL2(6) = VARIB(8)
CALL CHARS(44, YLABL2, 90, .25, 1.6, 12)
VARS(1) = 8.
VARS(2) = 8.5
VARS(3) = 0.
VARS(4) = 0.0
VARS(5) = 1.0
VARS(6) = 0.5
VARS(7) = 1.0
VARS(8) = 0.0
CALL XAXIS(1.2,5.5,VARS)
MD3 = MD3 + 1
CALL NUMBER(1,MD3,4,0,ALABL(5))
CALL CHARS(28,ALABL,0.0,1.3,9.5,12)
CALL CHARS(20,PLEGN,0.0,6.0,9.5,12)
VARS(1) = 7.
VARS(2) = 3.8
VARS(3) = 90.
VARS(4) = ATMINP
VARS(5) = ATMAXP
VARS(6) = AINTRV
VARS(7) = 1.
CALL YAXIS(1.2,5.5,VARS)
CONTINUE
DO 120 I = 1,NSLICE
JST = NSTAPS*(I-1)
DO 115 J = 1,NSTAPS
Y(J) = TPFL(JST+J)
SYRBOL = SYMBL(1)
KS = 0
LABEL EVERY 10TH POINT WITH THE SLICE NUHBEB, TO IDENTIFY THE CURVES.
KSTART = I + 1
DO 118 K = KSTART,NSTAPS,10
KS = KS + 1
XLBL(KS) = XP(K)
YLBL(KS) = Y(K)
IVARS(1) = 6
IVARS(2) = NLBLS
IVARS(3) = 3
IVARS(4) = 240*I
IVARS(5) = 1
IVARS(6) = 8
CALL GPLOT(XLBL,YLBL,IVARS)
IVARS(1) = 3
IVARS(2) = NSTAPS
IVARS(3) = 0
CALL GPLOT(XP,Y,IVARS)
VARS(1) = 7.
VARS(2) = 8.5
VARS(3) = 0.
VARS(4) = 0.0
VARS(5) = 1.0
VARS(6) = .5
VARS(7) = 1.0
CALL XAXIS(1.2,5.5,VARS)
CALL CHARS(42,XLABL,0.0,3.5,.05,15)
VARS(1) = 7.
VARS(2) = 3.8
VARS(3) = 90.
VARS(4) = ATMINP
VARS(5) = ATMAXP
VARS(6) = AINTRV
VARS(7) = 1.
CALL YAXIS(1.2,5.5,VARS)
CALL CHARS(28,PLEGN,0.0,6.0,4.5,12)
DO 130 I = 1,NSLICE
JST = NSTAPS*(I-1)
DO 125 J = 1, NSTAPS

125 Y(J) = TSM(JST+J)
SYMBOL = SYMBOL(I)
KS = 0
C
C --- LABEL EVERY 10TH POINT WITH THE SLICE NUMBER, TO
C IDENTIFY THE CURVES.
KSTART = I + 1
DO 128 K = KSTART, NSTAPS, 10
KS = KS + 1
XLBL(KS) = XS(K)
IVARS(1) = 6
IVARS(2) = NLBLS
IVARS(3) = 3
IVARS(4) = 240+I
IVARS(5) = 8
IVARS(6) = 0
CALL GPLOT(XLBL, YLBL, IVARS)
IVARS(1) = 3
IVARS(2) = NSTAPS
IVARS(3) = 0
CALL GPLOT(XS, Y, IVARS)
C
C
CONTINUE
RETURN
END

C---- SOURCE.NPREPAT
SUBROUTINE NPREPAT (ICHNL, NTTG)
C
COMMON /BOUND/ BCXS(400), BCXP(400), BCHGS(1000), BCHGP(1000),
BCPS(1000), BCPGP(1000), THUBIN(400), THUB(80),
QHUBIN(400), QHUB(80), QTIPIN(400), QTIP(80),
BCTIME(50), TTI0(50), PT10(50), WPLEN,
BTIME(50), AKCTBL(20), AKWTBL(20), NBCS, NBCP
COMMON /FRIC/ ALPHA, BETA, DELTA, EPS
COMMON /PRPS/ CPO, GAMO, DP(80), SP(80), RE(80),
CPC(80), GACC(80), DUMB1(80), DUMB2(80)
COMMON /SPECL/ CHANL(8000), TITLE(30), INDCHN(2000),
IPLOT, MD1, MD2, MD3, IADJX, IWRITE
COMMON /TCC/ ADUMP, ETA, CD, CP,
GAM, PIM, R, SPAN, TOG,
WDUMP, WIM, ACK(15,80), AKM(15,80),
A(80), AJST(80), AM2(80), CNUM(80),
DH(80), DHF(80), DHJ(80),
DLX(400), FF(80), HC(80), HG(80),
I. P(2,15,80), PEXIT(15), PUMP(80), QG(80),
QSNK(80), RR(80), S(15), T(2,15,400),
WJ(15,80), WCR(2,15,80),
ICOR, IPILM, IHub, ITP,
ISBLOK, ISLICE, NBLKSZ, NSLICE,
NFWD, NSTA, IHC(80)

C COMMON /TRNSNT/ RHOC, RHOM, SPHTN,
DLTYME, TME, TEP, TYMMA

C ICHNL IS THE CHANNEL NUMBER; = 1 FOR THE HUB REGION,
= NSLICE AT THE TIP

C-- LOCATE INPUT DATA FOR THIS CHANNEL AND STORE IT IN WORKING ARRAYS.
C I1 IS THE STARTING POINT IN THE INDCHN ARRAY FOR THIS CHANNEL
C
I1 = INDCHN(ICHNL)
IF ( ICHNL.NE.INDCHN(I1)) GO TO 290
C-- IF ABOVE TEST IS TRUE, THEN THE DATA IS NOT STORED WHERE EXPECTED
C
10 CONTINUE
IFILM = INDCHN(I1+1)
ICOR = INDCHN(I1+2)
NFWD = INDCHN(I1+3)
NSTA = INDCHN(I1+4)
ISBLOK = INDCHN(I1+5)
NBLKSZ = INDCHN(I1+6)
IPLT = INDCHN(I1+7)
MD1 = INDCHN(I1+8)
MD2 = INDCHN(I1+9)
IHub = INDCHN(I1+12)
ITIP = INDCHN(I1+13)
I11 = I1 + 14
CD = CHANL(ISBLOK)
ALPHA = CHANL(ISBLOK+1)
BETA = CHANL(ISBLOK+2)
DELTA = CHANL(ISBLOK+3)
EPS = CHANL(ISBLOK+4)
ADUMP = CHANL(ISBLOK+6)
SPAN = CHANL(ISBLOK+7)
S(ICHNL) = SPAN
BTA = CHANL(ISBLOK+6)
DLTYME = CHANL(ISBLOK+9)
TEPS = CHANL(ISBLOK+10)
NODSP = 5*NFWD
NODST = 5*NSTAA
I1 = ISBLOK + 14
I3 = ISBLOK + 10 + 2*NODST
C*****
12 CONTINUE
DO 205 I = 1,NODST
IM = I1 + I
TAU(I) = CHANL(IM)
A(I) = TAU(I)*SPAN

COP/NTRNSNT/ RHOC, RHOM, SPHTC,
2DLTYME, TME, TEPS,
C ICHNL IS THE CHANNEL NUMBER;
C = 1 FOR THE HUB REGION,
C = NSLICE AT THE TIP
C-- LOCATE INPUT DATA FOR THIS CHANNEL AND STORE IT IN WORKING ARRAYS.
C I1 IS THE STARTING POINT IN THE INDCHN ARRAY FOR THIS CHANNEL
C
I1 = INDCHN(ICHNL)
IF ( ICHNL.NE.INDCHN(I1)) GO TO 290
C-- IF ABOVE TEST IS TRUE, THEN THE DATA IS NOT STORED WHERE EXPECTED
C
10 CONTINUE
IFILM = INDCHN(I1+1)
ICOR = INDCHN(I1+2)
NFWD = INDCHN(I1+3)
NSTA = INDCHN(I1+4)
ISBLOK = INDCHN(I1+5)
NBLKSZ = INDCHN(I1+6)
IPLT = INDCHN(I1+7)
MD1 = INDCHN(I1+8)
MD2 = INDCHN(I1+9)
IHub = INDCHN(I1+12)
ITIP = INDCHN(I1+13)
I11 = I1 + 14
CD = CHANL(ISBLOK)
ALPHA = CHANL(ISBLOK+1)
BETA = CHANL(ISBLOK+2)
DELTA = CHANL(ISBLOK+3)
EPS = CHANL(ISBLOK+4)
ADUMP = CHANL(ISBLOK+6)
SPAN = CHANL(ISBLOK+7)
S(ICHNL) = SPAN
BTA = CHANL(ISBLOK+6)
DLTYME = CHANL(ISBLOK+9)
TEPS = CHANL(ISBLOK+10)
NODSP = 5*NFWD
NODST = 5*NSTAA
I1 = ISBLOK + 14
I3 = ISBLOK + 10 + 2*NODST
C*****
12 CONTINUE
DO 205 I = 1,NODST
IM = I1 + I
TAU(I) = CHANL(IM)
A(I) = TAU(I)*SPAN

COP/NTRNSNT/ RHOC, RHOM, SPHTC,
2DLTYME, TME, TEPS,
IM = IM + NODST
DLX(I) = CHANL(IM)
DO 215 I = 1,NSTA
IFLU = 5*I
DH(I) = 4.0*A(IFLU)/(2.*(SPAN+TAU(IFLU)))
IM = IM + NODST
DHJ(I) = CHANL(IM)
IM = IM + NODST
XN(I) = CHANL(IM)
IF (DHJ(I) .GT. 0.0 .AND. XN(I) .GT. 0.0) GO TO 202
AJet(I) = 0.0
GO TO 212
C-- NOW, GIVEN SLICE, ICHNL, EVALUATE B.C. AT METAL NODE POINTS. IN THE
C-- FOLLOWING:
C-- XS & XP ARE DISTANCE FROM LEADING EDGE, ALONG OUTSIDE SURFACE
C-- (INCHES), FOR SUCTION & PRESSURE SIDES.
C-- THE CONVENTIONS USED IN THE FOLLOWING ARE: INDEX BEGINNING WITH -I-
C-- IS A SLICE INDEX, INDEX BEGINING WITH -N- IS A TIME INDEX, INDEX
C-- BEGINING WITH -L- IS A N X INDEX, AND AN INDEX BEGINING WITH -J- IS
C-- A PROPERTY INDEX I.E. HG,QG,TG,PG.
C-- FIRST, CHECK THAT THIS IS A TRANSIENT CASE, AND DETERMINE THE MAX.
C-- BCTIME INDEX, NMX.
C-- NEXT, EAT THE LOCATION IN THE BCTIME ARRAY OF THE CURRENT TIME, AND CALCULATE THE VALUE OF THE INTERPOLATING
C-- PARAMETER, TMPFRAC.
NMX = 1
310 IF (BCTIME(NMX+1).LE.0.0) GO TO 315
NMX = NMX + 1
GO TO 310
315 CONTINUE
C--
TMFRAC = 0.0
NLST = 1
IF (NMIEQ.1.OR.TYME.LE.0.0) GO TO 330
C-- THE ABOVE TRANSFER OCCURS IF THIS IS A STEADY STATE PROBLEM
C-- THE FOLLOWING TRANSFER OCCURS IF WE ARE BEYOND THE LAST BCTIME ENTRY
NLST = NMIE
IF (TYME.GE.BCTIME(NMIE)) GO TO 330
NMIE1 = NMIE - 1
DO 320 N = 1,NMIE1
NLST = N
IF (TYME.GE.BCTIME(N).AND.TYME.LT.BCTIME(N+1)) GO TO 325
320 CONTINUE
325 TMFRAC = ((TYME-BCTIME(NLST))/(BCTIME(NLST+1)-BCTIME(NLST)))
C-- NEXT, SEARCH THE BCXS & BCXP ARRAYS TO FIND THE X INTERPOLATING FACTORS, XSF & XPP, FOR POSITIONS XS & XP, SLICE ICHNL.
C-- THE BRACKETING INDICES ARE LBLOWS & LABOVS & LBLOWP & LABOVP.
C-- THE STARTING POINTS IN THE BCXS & BCXP ARRAYS FOR THIS SLICE ARE:
330 LSS = (ICHNL-1)*NBCS
LSP = (ICHNL-1)*NBCP
XS = 0.0
XP = 0.0
C-- THE STARTING POINTS IN THE PROPERTY ARRAYS, FOR THE LATEST TIME STEP ARE:
JS1S = NSLICE*(NLST-1)*NBCS + LSS
JS1P = NSLICE*(NLST-1)*NBCP + LSP
IF (NMIE.EQ.NLST) GO TO 335
JS2S = JS1S + NSLICE*NBCS
JS2P = JS1P + NSLICE*NBCP
335 HG(1) = BCHGS(JS1S+1) + TMFRAC*(BCHGS(JS2S+1)-BCHGS(JS1S+1))
TG(1) = BCTGS(JS1S+1) + TMFRAC*(BCTGS(JS2S+1)-BCTGS(JS1S+1)) +460
QG(1) = BCQGS(JS1S+1) + TMFRAC*(BCQGS(JS2S+1)-BCQGS(JS1S+1))
PG(1) = BCPGS(JS1S+1) + TMFRAC*(BCPGS(JS2S+1)-BCPGS(JS1S+1))
C-- THE OUTSIDE SURFACE NODE NUMBERS FOR S & P SIDES ARE:
NNS = 5*K - 4
NNP = 5*K + 1
XS = XS + DLX(NNS)
XP = XP + DLX(NNP)
C-- INSERT ERROR MESSAGE HERE-- EXTRAPOLATING BEYOND THE BCXS TABLE
C-- DO 340 L = 1,NBCS
LBLOWS = LSS + L -1
LABOVS = LSS + L
IF (BCXS(LABOVS).GT.XS) GO TO 342
340 CONTINUE
C-- DO 345 L = 1,NBCP
LBLOWP = LSP + L -1
LABOVP = LSP + L
IF (BCXP(LABOVP).GT.XP) GO TO 347
345 CONTINUE
\[ XP = \frac{(XP - BCXP(LBLOWP)) \times (BCXP(LBLOWP) - BCXP(LBLOWP))}{(BCXP(LBLOWP) - BCXP(LBLOWP))} \]

Now the fractions are known, calculate the interpolated properties.

First, for the steady state or for times beyond the last BCTIME:

\[ JS = NSLICE \times (NLST - 1) \times NBCS + LBLOWS \]

\[ JBP = NSLICE \times (NLST - 1) \times NBCP + LBLOWP \]

\[ HG(K) = BCQGS(JBIS) + XSF \times (BCQGS(JBIS + 1) - BCQGS(JBIS)) \]

\[ QG(K) = BCQGP(JBP) + XPF \times (BCQGP(JBP + 1) - BCQGP(JBP)) \]

\[ TG(K) = BCTGS(JBIS) + XSF \times (BCTGS(JBIS + 1) - BCTGS(JBIS)) + 460. \]

\[ PG(K) = BCPGS(JBIS) + XSF \times (BCPGS(JBIS + 1) - BCPGS(JBIS)) \]

If \( NMX \neq 1 \) or \( TYPE \geq BCTIME \) or \( TYLE \leq 0.0 \) go to 350

\[ JBS = NSLICE \times (NLST - 1) \times NBCS + LBLOWS \]

\[ JB2P = NSLICE \times (NLST - 1) \times NBCP + LBLOWP \]

IF (NMX.EQ.1.OR.TYME.GE.BCTIME(NNX).OR.TYLIE.LE.O.O) GO TO 350

\[ AHG = BCQGS(JB2S) + XSF \times (BCQGS(JB2S + 1) - BCQGS(JB2S)) \]

\[ AQG = BCQGP(JB2P) + XPF \times (BCQGP(JB2P + 1) - BCQGP(JB2P)) \]

\[ ATG = BCTGS(JB2S) + XSF \times (BCTGS(JB2S + 1) - BCTGS(JB2S)) + 460. \]

\[ APG = BCPGS(JB2S) + XSF \times (BCPGS(JB2S + 1) - BCPGS(JB2S)) \]

\[ JBS = JS - NSLICE \times (NLST - 1) \times NBCS + LBLOWS \]

\[ QG(K) = BCQGP(JB2P) + XPF \times (BCQGP(JB2P + 1) - BCQGP(JB2P)) \]

\[ TG(K) = BCTGS(JB2S) + XSF \times (BCTGS(JB2S + 1) - BCTGS(JB2S)) + 460. \]

\[ PG(K) = BCPGP(JB2P) + XPF \times (BCPGP(JB2P + 1) - BCPGP(JB2P)) \]

IF (NMX.EQ.1.OR.TYME.GE.BCTIME(NNX).OR.TYLIE.LE.O.O) GO TO 350

Data storage is messed up.

END
C
C NODE NAMES FOR STATION IS ARE:
C LCOOL = COOLANT NODE AT IS.
C LIN = INSIDE WALL NODE.
C L = MID WALL NODE.
C LOUT = OUTSIDE WALL NODE.
C LCUP = ADJACENT UPSTREAM COOLANT NODE.
C LCUPS = ADJACENT INSIDE WALL NODE
C LUP = ADJACENT UPSTREAM MID-WALL NODE.
C LDN = ADJACENT DOWNSTREAM MID-WALL NODE.
C LJ = JUNCTION OF COATING AND WALL METAL

IS

L
C

-----------LOUT*-----------
C

WALL
C

---LUP--*------LDS*-----
C

FLOW
C

---&gt; LCUP-* LCOOL*
C

-LCUPP-**LCOOLP-*********
C

PRESSURE SIDE WALL OR PLENUM
C

********************************************************************
C

REAL*8 TCOF
C

COM/MOON /BOOII/ BCXS(400), BCXP(400), BCHG(1000), BCHGP(1000),
C

Z

COM BCTGS(1000), BCTGP(1000), BCQGS(1000), BCQGP(1000),
C

Z

COM BCPGS(1000), BCPGP(1000), THUBIN(400), THUB(80),
C

Z

COM QHUBIN(400), QHUB(80), TTPIN(400), TTP(80),
C

Z

COM QTIPI(400), QTIP(80), BHOVGA(400), PEX(400),
C

Z

COM BTCIME(50), TTIO(50), PTIO(50), WPLEN,
C

Z

COM WSVST(50), AKCTSL(2C), AKWTEB(20), NBCS, NBCP
C

*************************************************************************
C

COM/MOON /PMCOOL/ BHOVGA(80), PG(80), XFC(80), FLMEFF(80),
C

Z

COM XMUC(80), EMES(80), REFC(80), NFCSUP(80),
C

*************************************************************************
C

COM/MOON /MATRX/ TCOF(400,30)
C

*************************************************************************
C

COM/PEPS/ CP0, GAMO, DP(80), SP(80), BE(80),
C

Z

COM CPC(80), GAMC(80), DUMR1(80), DUMR2(80),
C

*************************************************************************
C

COM/TCO/ ADUMP, BTA, CD, CP,
C

Z

COM GAM, FLM, R, SPAN, TOG,
C

Z

COM WDUMP, WIT, AKC(15,80), AKE(15,80),
C

Z

COM A(400), AJET(80), AM2(80), CNMU(80),
C

Z

COM DH(80), DHP(80), DJH(80),
C

Z

COM DLX(400), FP(80), HC(80), HG(60),
C

Z

COM P(2,15,80), PEXIT(15), PUMP(80), QG(80),
C

Z

COM QSNK(80), S(15), T(2,15,400),
C

Z

COM TG(80), TAU(400), WFC(80),
C

Z

COM WJ(15,80), WCRS(2,15,80), XN(80),
C

Z

COM ICOR, IFILH, IHUB, ITIP,
C

*************************************************************************
C** BEGIN OVERALL LOOP, WHERE LOOP VARIABLE (IS) IS THE STATION NUMBER

DO 440 IS = 1,NSTA
ISUP = IS - 2
YIMP = 0.0
YFINS = 0.0
YCONV = 0.0
YIMPU = 0.0
YFINSU = 0.0
YCONVU = 0.0
YIMPUU = 0.0
YFNSUU = 0.0
YCNVUU = 0.0
IF (IHC(IS).EQ.1) YIMP=1.0
IF (IHC(IS).EQ.2) YCONV= (1.0+RCVRY*AM2(IS)*(GAMC(IS)-1.)/2.)
IF (IHC(IS).EQ.3) YFINS=1.0
FACTOR = 1.0
IF (IS.EQ.ICOMS.OR.IS.EQ.ICOMP) FACTOR = .5
ISENS = 0
ISEW = IS - 2*(IS/2)

C IF (IS.GT.NFWD+1)-- IN TRAILING EDGE REGION, GO TO 380

C IF(IS.GT.NFWD+1) GO TO 380

C J9 = 16
LCOOL = 5*IS
LIN =LCOOL-1
L =LCOOL-2
LJ = LCOOL - 3
LOUT = LCOOL - 4
IF (IHC(IS) .NE. 2) GO TO 320
C
C FORCED CONVECTION HC:
TMP = (T(2, ISLICE, LCOOL) + T(2, ISLICE, LIN))/2.
CALL GASTBL(TMP, C, CP, GAM, PD, R, XMU)
RE(IS) = 12.*3600.*ABS(WCROS(2, ISLICE, IS))*DH(IS)/(A(LCOOL)*XMU)
HC(IS) = .023*12.*((C/DH(IS))*(RE(IS)**.8)*(PD**.333)
C 320 CONTINUE
IF (IS.GE.JS) GO TO 322
C
C SPECIAL CASE FOR STATION NUMBER 1:
IF(IS.EQ.1) ISUP = 2
C IF STATION IS IS FORWARD OF FLOW SPLIT, AND ON SAME SIDE, GO TO 370
IF (ISEN.EQ.JSENS) GO TO 370
IF (ISUP.LT.1) ISUP = 1
IF (IHC(ISUP).EQ.1) YIHPU = 1.0
IF (IHC(ISUP).XQ.3) YFINSU = 1.0
LCUP = LCOOL - 10
LCUPS = LCOOL - 11
LUP = LCOOL - 12
LDN = LCOOL + 8
IF (IS.NE.2) GO TO 324
C
C IF THIS IS STATION NUMBER 2:
LCUP = 5
LCUPS = 4
LUP = 3
324 CONTINUE
IF (IS.GT.1) GO TO 326
C
C IF THIS IS STATION NUMBER 1:
LCUP = 10
LCUPS = 9
LUP = 8
326 CONTINUE
C
C IS = 1, STATION NO. 1, LEADING EDGE NODES
C IF (IS.NE.JS) GO TO 330
C
C*******************************************************************************
C*********** THIS BLOCK COMPUTES TCOF ELEMENTS FOR THE STATION AT WHICH
C THE FLOW SPLITS
C*********** IS = JS
C*******************************************************************************
C 328 CONTINUE
DX1 = DLX(LUP)
IF (DX1.EQ.0.0) DX1 = DLX(L)
DX2 = DLX(LDN)
IF (DX2.EQ.0.0) DX2 = DLX(L)
DX3 = DLX(LUP-2)
IF (DX3.EQ.0.0) DX3 = DLX(LOUT)
DX4 = DLX(LDN-2)
IF (DX4.EQ.0.0) DX4 = DLX(LOUT)
DX5 = DLX(LUP+1)
IF (DX5.EQ.0.0) DX5 = DLX(LIN)
DX6 = DLX(LDN+1)
IF (DX6.EQ.0.0) DX6 = DLX(LIN)
DX9 = DLX(LUP-1)
IF (DX9.EQ.0.0) DX9 = DLX(LIN)
DX10 = DLX(LDN-1)
IF (DX10.EQ.0.0) DX10 = DLX(LIN)
CURV = 1.0 + (DX9+DX10)/(DX3+DX4)

NTARAYT 4441
NTARAYT 4442
NTARAYT 4443
NTARAYT 4444
NTARAYT 4445
NTARAYT 4446
NTARAYT 4447
NTARAYT 4448
NTARAYT 4449
NTARAYT 4450
NTARAYT 4451
NTARAYT 4452
NTARAYT 4453
NTARAYT 4454
NTARAYT 4455
NTARAYT 4456
NTARAYT 4457
NTARAYT 4458
NTARAYT 4459
NTARAYT 4460
NTARAYT 4461
NTARAYT 4462
NTARAYT 4463
NTARAYT 4464
NTARAYT 4465
NTARAYT 4466
NTARAYT 4467
NTARAYT 4468
NTARAYT 4469
NTARAYT 4470
NTARAYT 4471
NTARAYT 4472
NTARAYT 4473
NTARAYT 4474
NTARAYT 4475
NTARAYT 4476
NTARAYT 4477
NTARAYT 4478
NTARAYT 4479
NTARAYT 4480
NTARAYT 4481
NTARAYT 4482
NTARAYT 4483
NTARAYT 4484
NTARAYT 4485
NTARAYT 4486
NTARAYT 4487
NTARAYT 4488
NTARAYT 4489
NTARAYT 4490
NTARAYT 4491
NTARAYT 4492
NTARAYT 4493
NTARAYT 4494
NTARAYT 4495
NTARAYT 4496
NTARAYT 4497
NTARAYT 4498
NTARAYT 4499
NTARAYT 4500

TRTRMC = 0.0
IF (TLIME.GT.0.0.AND.TYME.GE.0.) TRTRMC=(3600./144.)*RHOC*SPHTC*
Z (TAU(LOUT)**2)/(4.*AKC(ISLICE,IS)*DLTYME)
TCOF(LOUT,13) = -TREPS + TRTRMC*CURV
TCOF(LOUT,12) = TRTRMC*CURV + TREPS*(1.0* (1.0-BTA)*HG(IS) *TAU(LOUT)/
Z (12.*AKC(ISLICE,IS)))
TCOF(LOUT,J9) = -(1.0).*TREPS*FLMEPF(IS) *HG(IS) *TAU(LOUT)/
Z (12.*AKC(ISLICE,IS))
TCOF(LOUT,24) = (BTA*OG(IS) ) + (1.0-BTA)*HG(IS) *TG(IS) *
Z (1.0-FLMEPF(IS)) *TAU(LOUT)/(12.*AKC(ISLICE,IS))
Z - T(1.ISLICE,LOUT)*( (1.0-TREPS)*(1.0-BTA)*HG(IS) *TAU(LOUT)/
Z (12.*AKC(ISLICE,IS)) + 1.) - TRTRMC*CURV
Z + T(1.ISLICE,LJ)*(1.0-TREPS*TRTRMC*CURV)
Z + T(1.ISLICE,LCOOL)* (1.0-TREPS)*FLMEPF(IS) *(1.0-BTA)*HG(IS) *
Z TAU(LOUT)/(12.*AKC(ISLICE,IS))

TCOF(LJ,11) = TREPS
TCOF(LJ,13) = TREPS*(AKW(ISLICE,IS)/AKC(ISLICE,IS)) *
Z (2.*TAU(LOUT)/TAU(L))*(DX1+DX2+DX3+DX4)/(DX9+DX10+DX3+DX4)
TCOF(LJ,12) = - TCOF(LJ,11) - TCOF(LJ,13)
TCOF(LJ,24) = (1.0-TREPS)* (T(1.ISLICE,LJ) -T(1.ISLICE,LOUT)) *
Z (T(1.ISLICE,LJ) -T(1.ISLICE,L))* TCOF(LJ,13)/TREPS)

J1 = 12 - L + LUP
J2 = 12 - L + LDN

THETA1 = (DX1+DX2+DX5+DX6)/(DX1+DX2+DX9+DX10)
THETA2 = ((TAU(L) +TAU(LUP))/ (2.*DX1) ) *2.*TAU(L) /(DX1+DX2+DX9+DX10)
THETA3 = ((TAU(L) +TAU(LDN))/ (2.*DX2) ) *2.*TAU(L) /(DX1+DX2+DX9+DX10)
THETA6 = 24.*TAU(L)/(AKW(ISLICE,IS) *S(ISLICE) *(DX1+DX2+DX9+DX10))
THETA4 = 0.0
THETA5 = 0.0
HUB1 = 0.0
HUB3 = 0.0
TIP1 = 0.0
TIP3 = 0.0
IF (ISLICE.EQ.1) GO TO 3290

FOR A SLICE THAT IS NOT AT THE HUB OF THE BLADE:

THETA4 = (TAU(L)/S(ISLICE) ) * (TAU(L)/(S(ISLICE) +S(ISLICE-1) ) ) *
\[ 2 \cdot \frac{(DX_1 + DX_2)}{(DX_1 + DX_2 + DX_9 + DX_{10})} \]

IF (ISLICE.EQ.NSLICE) GO TO 3292
THETA5 = THETA4*(S(ISLICE)+S(ISLICE-1))/(S(ISLICE)+S(ISLICE+1))
TBELOW = T(1,ISLICE-1,L)
TABOVE = T(1,ISLICE+1,L)
GO TO 3294
C
C FOR THE SLICE AT THE HUB END OF THE BLADE:
C
3290 CONTINUE
IF (IHUB.EQ.1) HUB1 = ((DX1+DX2)/(DX1+DX2+DX_9+DX_{10}))*
\[ 2 \cdot (TAU(L)/S(1))^{**2} \]
THETA5 = 0.0
IF (NSLICE.GT.1) THETA5 = (TAU(L)/S(ISLICE))*(TAU(L)/(S(ISLICE)+S(2)))*
\[ 2 \cdot (DX_1+DX_2)/(DX_1+DX_2+DX_9+DX_{10}) \]
Z
IF (IHUB.EQ.3) HUB3 = ((DX1+DX2)/(DX1+DX2+DX_9+DX_{10}))*\( (TAU(L)**2)/(\text{AKW (1,ISLICE) * 12. * S(NSLICE)}) \)
C
IHUB = 3, THE HEAT FLUX AT THE HUB END IS SPECIFIED (BTU/HR FT**2) ****NTARAYT 4519
TBELOW = T(1,1,L)
IF (IHUB.EQ.1) TBELOW = THUB(ISLICE)
TABOVE = T(1,1,L)
IF (IHUB.EQ.1) GO TO 3294
C
C FOR THE SLICE AT THE BLADE TIP, (IF THERE ARE MORE THAN 1 SLICES BEING CONSIDERED):
C
3292 CONTINUE
IF (ITIP.EQ.1) TIP1 = ((DX1+DX2)/(DX1+DX2+DX_9+DX_{10}))*
\[ 2 \cdot (TAU(L)/S(NSLICE))^{**2} \]
IF (NSLICE.GT.1) TBELOW = (1+(ISLICE-1,L)
IF (ITIP.EQ.3) TIP3 = ((DX1+DX2)/(DX1+DX2+DX_9+DX_{10}))*\( (TAU(L)**2)/(\text{AKW (NSLICE,IS) * 12. * S(NSLICE)}) \)
Z
TABOVE = T(1,ISLICE,L)
IF (ITIP.EQ.1) TABOVE = TIP(ISLICE)
C
3294 CONTINUE
THETA9 = 0.0
IF (DLTMY.EQ.0.0.AND.TYME.GE.0.) THETA9 = 2.3600.*RHOM*SPHTM*
\[ (DX_1+DX_2)*(TAU(L)**2)/(144.*\text{AKW (ISLICE,IS)}*\text{AKW (ISLICE,IS)}) \]
Z
TCOF(L,11) = 1.0*TREPS
TCOF(L,13) = THETA1*TREPS
TCOF(L,14) = THETA2*TREPS
TCOF(L,12) = THETA3*TREPS
TCOF(L,24) = QSNK(ISLICE)*THETA6 - (THETA4+HUB1)*TBELOW
Z
\[ -T(1,ISLICE,L)*THETA1+T(1,ISLICE,L) \]
Z
\[ +T(1,ISLICE,L)*(1.0-TREPS)*(1.0+THETA1+THETA2+THETA3+THETA4)*HUB1+TIP1) - THETA9 \]
C
AHTRN1 = (DX5 + DX6)*S(ISLICE)/2.
THETA8 = 2.*HX*HC(ISLICE)*AHTRN1*TAU(L)/(12.*AKW (ISLICE,IS)*S(ISLICE))

128
**Z**

\[ \text{TCOF}(\text{LIN},11) = \text{TREPS} \]

\[ \text{TCOF}(\text{LIN},24) = - \text{THETA8} \cdot \text{TOG} - (1.0 - \text{TREPS}) \cdot (\text{T}(1,\text{ISLICE},\text{L}) - \text{T}(1,\text{ISLICE},\text{LIN}) \cdot (1.0 - \text{THETA8})) \]

\[ \text{Z} \]

\[ \text{TCOF}(\text{LIN},12) = (-1.0 - \text{THETA8}) \cdot \text{TREPS} \]

\[ \text{TCOF}(\text{LCOOL},12) = 1.0 \]

\[ \text{TCOF}(\text{LCOOL},24) = \text{TOG} \]

\[ \text{IF (TYME.GT.0.)} \]

\[ \text{RCHRD} = \frac{(144. / 3600.) \cdot \text{AKW}(\text{ISLICE}, \text{IS}) \cdot \text{DLTYME}}{\text{RHOH} \cdot \text{SPHTM} \cdot \left( \left( \frac{\text{DXI} + \text{DXZ}}{2} \right)^2 \right)} \]

\[ \text{Z} \]

\[ \text{IF (TYME.GT.0.)} \]

\[ \text{RTRNV} = \frac{(144. / 3600.) \cdot \text{AKW}(\text{ISLICE}, \text{IS}) \cdot \text{DLTYME} - \text{RHOH} \cdot \text{SPHTM} \cdot (\text{T}(\text{L})^2)}{\text{RHOH} \cdot \text{SPHTM} \cdot (\text{TAU}(\text{L})^2)} \]

\[ \text{IF (RCHRD.GT.RCHRDH)} \]

\[ \text{RCHRDH} = \text{RCHRD} \]

\[ \text{IF (RTRNV.GT.RTRNVH)} \]

\[ \text{RTRNVH} = \text{RTRNV} \]

\[ \text{GO TO 440} \]

---

**C**

\[ \text{CONTINUE} \]

\[ \text{DX1} = \text{DLX} \cdot (\text{L}) \]

\[ \text{DX2} = \text{DLX} \cdot (\text{LDN}) \]

\[ \text{DX3} = \text{DLX} \cdot (\text{LOUT}) \]

\[ \text{DX4} = \text{DLX} \cdot (\text{LDN-2}) \]

\[ \text{DX5} = \text{DLX} \cdot (\text{LIN}) \]

\[ \text{DX6} = \text{DLX} \cdot (\text{LDN+1}) \]

\[ \text{DX7} = \text{DLX} \cdot (\text{LCOOL}) \]

\[ \text{DX9} = \text{DLX} \cdot (\text{LJ}) \]

\[ \text{DX10} = \text{DLX} \cdot (\text{LDN-1}) \]

\[ \text{IF (IS.GT.1)} \]

\[ \text{GO TO 332} \]

**C**

\[ \text{DX1} = \text{DLX} \cdot (\text{LUP}) \]

\[ \text{DX9} = \text{DLX} \cdot (\text{LUP-1}) \]

\[ \text{DX3} = \text{DLX} \cdot (\text{LUP-2}) \]

\[ \text{DX5} = \text{DLX} \cdot (\text{LUP+1}) \]

\[ \text{DX7} = \text{DLX} \cdot (\text{LUP+2}) \]

---

**C**

\[ \text{CONTINUE} \]

\[ \text{LCU}{\text{PP}} = \text{LCOOL} - 5 \]

\[ \text{LCOO}{\text{LP}} = \text{LCOOL} + 4 \]

\[ \text{J1} = 12 - \text{L} + \text{LUP} \]

\[ \text{J2} = 12 - \text{L} + \text{LDN} \]

\[ \text{J4} = 12 - \text{LCOOL} + \text{LC}{\text{UP}} \]

\[ \text{J5} = 12 - \text{LCOOL} + \text{L}{\text{UP}} \]

\[ \text{J6} = 12 - \text{LCOOL} + \text{L}{\text{CP}} \]

\[ \text{J8} = 12 - \text{LCOOL} + \text{L}{\text{COOL}} \]

---

**C**

\[ \text{CONTINUE} \]
\[ A_{1} = \text{SPAN}\times\text{DX5}/2. \]

\[ A_{2} = A_{1} \]

\[ A_{3} = 0.0 \]

\[ A_{4} = 0.0 \]

\[ C \]

\[ \text{IF (IHC(IS).EQ.3.AND.IS.LE.NFWD)} \]

\[ \text{CALL HCPINS(IS,DELTAN,LCOO\L,C\L\LUP,\L\LIN,LC\LOOLP,PINS,EFAREA)} \]

\[ \text{IF (IHC(IS-2).EQ.3) A1 = EF\AREA(IS-2)/2.} \]

\[ \text{IF (IHC(IS).EQ.3) A2 = EF\AREA(IS)/2.} \]

\[ \text{IF (IHC(IS).EQ.3) A4TRN1 = EF\AREA(IS).} \]

\[ I_{3} = 12 - \text{LIN} + \text{LCOOL} \]

\[ \text{C \]

\[ \text{CURV} = 1.0 + (\text{DX9+DX10})/(\text{DX3+DX4}) \]

\[ \text{C \]

\[ \text{STOP} \]

\[ \text{C FOR MID-METAL NODE (L)}: \]

\[ \text{TCOF(L,12), 12 REFERS TO NODE L} \]

\[ \text{J1 = 2} \]

\[ \text{TCOF(L,13) = -TREPS} \]

\[ \text{J2 = 22} \]

\[ \text{Z (12.*AKC(ISLICE,IS)) \]

\[ \text{TCOF(L,14) = (BTA*QG(IS) + (1.0-BTA) *HG(IS)*TG(IS))} \]

\[ \text{T \]

\[ \text{TAU(LOUT)/(12.*AKC(ISLICE,IS))} \]

\[ \text{AT JUNCTION OF COATING AND METAL, NODE LJ:} \]

\[ \text{TCOF(LJ,11) = TREPS} \]

\[ \text{TCOF(LJ,12) = TREPS} \]

\[ \text{Z (2.*TAU(LOUT)/TAU(L))*(DX1+DX2+DX3+DX4)/(DX9+DX10+DX3+DX4)} \]

\[ \text{TCOF(LJ,12) = -TCOF(LJ,11) - TCOF(LJ,13)} \]
TCOP(LJ,24) = (1.-TREPS)*((T(1,ISLICE,LJ)-T(1,ISLICE,LOUT)) +
(T(1,ISLICE,LJ)-T(1,ISLICE,L))*TCOP(LJ,13)/TREPS)

C

C FOR MID-METAL NODE:

C

THETA1 = (DX1+DX2+DX5+DX6)/(DX1+DX2+DX9+DX10)
THETA2 = ((TAU(L)+TAU(LUP))/(2.*DX1))*Z
THETA3 = 0.0
THETA6 = 24.*TAU(L)/(AKW(ISLICE,IS)*S(ISLICE)*(DX1+DX2+DX9+DX10))

C

IF (ISLICE.LT.NSTA-1) THETA3 = ((TAU(L)+TAU(LDN))/(2.*DX2))*Z

C

C FOR A SLICE THAT IS NOT AT THE HUB OF THE BLADE:

C

THETA4 = (TAU(L)/S(ISLICE))*(TAU(L)/(S(ISLICE)+S(ISLICE-1)))*Z

C

IF (ISLICE.EQ.NSLICE) THETA5 = (S(ISLICE)+S(ISLICE-1))/(S(ISLICE)+S(ISLICE+1))

C

C FOR THE SLICE AT THE HUB END OF THE BLADE:

C

HUB1 = ((DX1+DX2)/(DX1+DX2+DX9+DX10))**(Z)

C

C FOR IHUB = 1, HUB TEMPERATURE IS SPECIFIED

C

THETA5 = 0.0

C

IF (NSLICE.GT.1) THETA5 = (TAU(L)/S(1))*(TAU(L)/(S(1)+S(2)))*Z

C

IF (IHUB.EQ.3) HUB3 = ((DX1+DX2)/(DX1+DX2+DX9+DX10))*(TAU(L)**2)/Z

C

IHUB = 3, THE HEAT FLUX AT THE HUB END IS SPECIFIED (BTU/HR FT**2)

C

TBELOW = T(1,1,L)

C

TABOVE = T(1,1,L)

C

IF (NSLICE.GT.1) GO TO 3414

C

C FOR THE SLICE AT THE BLADE TIP, (IF THERE ARE MORE THAN 1 SLICES

C BEING CONSIDERED) :

C

IF (ITIP.EQ.1) TIP1 = ((DX1+DX2)/(DX1+DX2+DX9+DX10))*(2.*TAU(L)/Z

C

IF (NSLICE.GT.1) TBELOW = T(1,ISLICE-1,L)
IF (ITIP.EQ.3) TIP3 = ((DX1+DX2)/(DX1+DX2+DX9+DX10))*(TAU(L)**2)/ (AKW(NSLICE,IS)*12.*S(NSLICE))

TABOVE = T(1,ISLICE,L)

IF (ITIP.EQ.1) TABOVE = Ttip(IS)

C

3414 CONTINUE

THETA9 = 0.0

IF (DLTYME.GT.0.0.AND.TIME.GE.0.) THETA9 = 2.*3600.*RHOH*SPHTM*

Z (DX1+DX2)* (TAU(L)**2)/ (14Y.*ARY(ISLICE,IS)*

2 (DX1+DX2+DX9+DX10)*DLTYME)

C

C ENDEFF IS EFFECT OF HEAT TRANSFER FROM THE GAS TO THE REAR EDGE OF THE BLADE:

C

ENDEFF = 0.0

ENDPLX = 0.0

IF (IS.GE.NSTA-1.AND.BTA.EQ.0.D) ENDEFF = 2.*HG(IS)*(TAU(L)**2)/ (12.*AKY(ISLICE,IS)*(DX1+DX9))

Z (DX1+DX2)* (TAU(L)**2)/ (14Y.*ARY(ISLICE,IS)*

2 (DX1+DX2+DX9+DX10)*DLTYRE)

IF (IS.GE.NSTA-1.AND.BTA.GT.0.0) ENDPLX = QG(IS)*(TAU(L)**2)/ (12.*AKY(ISLICE,IS)*(DX1+DX9))

C CONTINUE

TCOF(L,11) = 1.0*TREPS

TCOF(L,13) = THETA1*TREPS

TCOF(L,J1) = THETA2*TREPS

TCOF(L,J2) = 0.0

TCOF(L,12) = (-1.0 - THETA1 - THETA2 - THETA3 - THETA4 - THETA5 - ENDEFF - HUB1 - TIP1)*TREPS - THETA9

TCOF(L,24) = QSNK(IS)*THETA6 - (THETA4+HUB1)*TBELOW - THETA6* (TCOF(L,24) - HUB1)*TBELOW - Z (THETA5+TIP1)*TABOVE - QHUB(IS)*HUB3 + QTIP(IS)*TIP3

THETA5 = THETA5*(1.0-TREPS)*(1.+THETA1)*TREPS + (1.0-TREPS)*(THETA1+THETA2+THETA3+THETA4+HUB1+TIP1)*TBELOW - Z (THETA5+TIP1)*TABOVE - QHUB(IS)*HUB3 + QTIP(IS)*TIP3

THETA9 = 0.0

C FOR INNER SURFACE NODE:

C

342 THETA8 = 2.*HX*HC(IS)*AHTRN1*TAU(L)/(12.*AKW(ISLICE,IS)*

Z S(ISLICE)*(DX1+DX2+DX5+DX6))

C

TCOF(LIN,11) = TREPS

TCOF(LIN,12) = (-1.0 - THETA8)*TREPS

TCOF(LIN,I3) = THETA8*(YCONV+YFINS)*TREPS

TCOF(LIN,24) = -YIMP*THETA8*T0G

Z (1.0-THETA8)*(T(1,ISLICE,L)-T(1,ISLICE,LIN)*(1.+THETA8)

Z THETA8*(YCONV+YFINS)*T(1,ISLICE,LCOOL))

C

IF THIS IS A TRAILING EDGE, PRESSURE SIDE, STATION, COOLANT NODE COINCIDES WITH SUCTION SIDE COOLANT NODE:

C

IF (ISENS.EQ.0) GO TO 343

TCOF(LCOOL,7) = -1.0

TCOF(LCOOL,12) = 1.0

TCOF(LCOOL,24) = 0.0

IF (ISENS.EQ.1) GO TO 430

343 CONTINUE

C

132
C FOR COOLANT NODE:
C FOR THE SPECIAL CASE OF IS = NFWD+1, GO TO 350
C
FILMW = WFC(IS)
IF (IS.GT.NFWD) FILMW = FILMW + WFC(IS+1)
IF (IS.EQ.NFWD+1) GO TO 350
C
WXCP = WCROS(2,ISLICE,IS)*144.*CPC(IS)*3600.
IF (IS.EQ.ICOMF.OR.IS.EQ.ICOMS) WXCP = WJ(ISLICE,JS)*144.*
C CPC(IS)*3600./2.
C
DEFINE A COOLANT SIDE TRANSIENT TERM, TRTRMG
TRTRMG = 0.0
IF (DLTYME.GT.0.0.AND.TYME.GE.0.) TRTRMG = (1.+CPC(IS-2)/CPC(IS))
Z
Z *(A(LCUP)+A(LCOOL))*DLX(LCOOL)/(16.*DLTYME)*
Z WCROS(2,ISLICE,IS)*12.)
TCOF(LCOOL,J4) = HX*TRPS*HC(ISUP)*A1/WXCP
TCOF(LCOOL,J5) = 144.*3600.*TaEPS*WCROS(2,ISLICE,ISUP)*
Z
Z *(1.+AM2(ISUP)*(GAMC(ISUP)-1.)/2.)*CPC(ISUP)/WXCP
Z - HX*TRPS*HC(ISUP)*(YCONVU+YFINS)*(A1+A3)/WXCP - TRTRMG
TCOF(LCOOL,11) = HX*TRPS*HC(IS)*A2/WXCP
TCOF(LCOOL,12) = TRPS*((-FILMW*144.*3600.*CPC(IS)/WXCP - TRTRMG -
Z - ((GAMC(IS)-1.)/2.)*AM2(IS)
Z - HX*HC(IS)*(A2 + A4)*(YCONV+YFINS)/WXCP) - TRTRMG
TREDGE = 0.0
IF (IS.GT.NFWD) TREDGE = ((1.-TRPS)*HX/WXCP)*
C
Z HX*HC(ISUP)*A3 + T(1,ISLICE,LCUPP)*
Z *(1+AM2(ISUP)*(GAMC(ISUP)-1.)/2.)*CPC(ISUP)/
Z WXCP - HX*(1.-TRPS)*HC(ISUP)*YCONV+
Z
Z YFINSU)*(A1+A3)/WXCP - TRTRMG
Z T(1,ISLICE,LCUPS)*HX*(1.-TRPS)*HC(IS)*A2/WXCP
Z T(1,ISLICE,LCOOL)*((1.-TRPS)*WCROS(2,ISLICE,ISUP)*
Z *(1.+AM2(ISUP)*(GAMC(ISUP)-1.)/2.)*CPC(ISUP)/
Z WXCP - HX*(1.-TRPS)*HC(ISUP)*YCONV+YFINS)/WXCP - TRTRMG - TREDGE
IF (IS.GT.NFWD) TCOF(LCOOL,J6) = TRPS*HX*HC(ISUP)*A3/WXCP
IF (IS.GT.NFWD) TCOF(LCOOL,J8) = TRPS*HX*HC(IS)*A4/WXCP
C
C*** OF THE TERMS YIMP,YFINS,YCONV, ONLY ONE CAN BE NON-ZERO AT A TIME
C
YIMP = 1.0 MEANS THAT IMPINGEMENT HEAT TRANSFER IS BEING CONSIDERED
C YFINS = 1.0 MEANS THAT A PIN FINNED SURFACE IS BEING USED
C YCONV = 1.0 MEANS A FORCED CONVECTION CORRELATION IS BEING USED
C
IF (TYME.GT.0.) BCHRSD = (144./3600.)*AKW(ISLICE,IS)*DLTYME/
Z (RHOM*SPHTM*(((DX1+DX2)/2.)**2))
IF (TYME.GT.0.) BTRNV = (144./3600.)*AKW(ISLICE,IS)*DLTYME/
Z (RHOM*SPHTM*(TAU(L)**2))
IF (BCHRSD.GT.BCHRSDM) BCHRSD = BCHRSD
IF (BTRNV.GT.BTRNVM) BTRNV = BTRNV
GO TO 430
IF (IHC(IS),EQ.3) CALL HCPINS(IS,DELTAN,LCOOL,LCUP,LIN,LCOOLP,FINS)

CONTINUE

IF (IHC(IS),EQ.3) CALL EPAREA(IS)
AHTMN1 = EPAREA(IS)
AHTMN2 = EPAREA(IS)/2.
A3 = SPAN*DLY(LCOOLP)/2.
A4 = SPAN*DLY(LCOOLP)/2.
A5 = EPAREA(IS+1)/2.
IF (IHC(IS),EQ.3) A4 = EPAREA(IS+1)/2.
IF (IHC(IS),EQ.3) GO TO 360

C

IF (IHC(IS),EQ.2) HC(IS) = HCPFCD(IS,LCOOL,LIN)
AHTMN1 = (DX5+DX6)*S(ISLICE)/2.
CONTINUE

360 IF (IHC(IS-1),EQ.1) YIMPUU = 1.0
IF (IHC(IS-1),EQ.2) YCNVUU = Y1.0+RCVRY*AM2(IS-1)*(GAMC(IS-1)-1.)/2.
IF (IHC(IS-1),EQ.3) YFNSUU = 1.0

Z

WCROS = WCROS(2,ISLICE,IS)*CPC(IS)*TOG
RC = WCROS(2,ISLICE,IS)*DX5+DX6)*S(ISLICE)/2.

VOBAR = (A(LCUP)+A(LCUP+1)+A(LCOOL))*(DX5+DX6)*S(ISLICE)/2.

TCOF(LCOOL,1) = TREP*HX*HC(IS-2)*AM2(IS-2)
TCOF(LCOOL,1) = TREP*HX*HC(IS-1)*AM2(IS-1)
TCOF(LCOOL,2) = TREP*(WCROS(2,ISLICE,IS)/CPC(IS))
TCOF(LCOOL,3) = TREP*(WCROS(2,ISLICE,IS)*YCNVUU+YFNSUU)

CONTINUE

C

C FOR MIXING ZONE; STATION NO. NFHD+1:

IF (IHC(IS-1),EQ.3) HE(1) = HCPINS(IS,DELTAN,LCOOL,LCUP,LIN,LCOOLP,FINS)
CONTINUE

360 CONTINUE

CONTINUE

C

C FOR MIXING ZONE; STATION NO. NFHD+1:

IF (IHC(IS-1),EQ.3) CALL HCPINS(IS,DELTAN,LCOOL,LCUP,LIN,LCOOLP,FINS)
CONTINUE

360 CONTINUE

C

C FOR MIXING ZONE; STATION NO. NFHD+1:

IF (IHC(IS-1),EQ.3) CALL HCPINS(IS,DELTAN,LCOOL,LCUP,LIN,LCOOLP,FINS)
CONTINUE

360 CONTINUE

C
C

CONTINUE

ISUP = IS + 2
LCUP = LCOOL + 10
LCUPS = LCOOL + 9
LUP = LCOOL + 8
LDN = LCOOL - 12
IF (IS.EQ.2) LDN = 3
IF (IS.EQ.1) LDN = 8
IF (IHC(ISUP).EQ.1) YIMPU = 1.0
IF (IHC(ISUP).EQ.2) YCONV = 1.0 + RCVRY*AM2(ISUP)*(GAMC(ISUP)-1.)/2.

DX1 = DLX(LUP)
DX2 = DLX(L)
DX3 = DLX(LUP-2)
DX4 = DLX(LOUT)
DX5 = DLX(LCUPS)
DX6 = DLX(LIN)
DX7 = DLX(LCUP)
DX9 = DLX(LUP-1)
DX10 = DLX(LJ)

GO TO 332

DX2 = DLX(8)
DX4 = DLX(6)
DX6 = DLX(9)
DX10 = DLX(7)
J1 = 22
J2 = 17
J4 = 21
J5 = 22
J6 = 8
J8 = 15

GO TO 333

C
C** SET UP FOR TRAILING EDGE REGION:
C**
C******************************************************************************
380 CONTINUE
   ISENS = IS - 2*(IS/2)
C ISENS = 0 MEANS IS IS EVEN AND STATION IS ON SUCTION SIDE
C ISENS = 1 MEANS IS IS ODD AND STATION IS ON PRESSURE SIDE
C
   LCOOL = 5*IS
   LIN = LCOOL - 1
   L = LCOOL - 2
   LJ = LCOOL - 3
   LOUT = LCOOL - 4
   LUP = L - 10
   LDN = L + 10
   LCOUP = LCOOL - 10
   LCUPS = LCUP - 1
   LCUPP = LCOOL - 6
   LCOOLP = LCOOL + 4
C
   I3 = 12 - LIN + LCOOL
   J1 = 12 - L + LUP
   J2 = 12 - L + LDN
   J4 = 12 - LCOOL + LCUPS
   J5 = 12 - LCOOL + LCUP
   J6 = 12 - LCOOL + LCUPP
   J8 = 12 - LCOOL + LCOOLP
   J9 = 16
C
   A1 = SPAN*DLX(LIN)/2.
   A2 = A1
   A3 = SPAN*DLX(LCOOLP)/2.
   A4 = A3
   IF (IHC(IS-2).EQ.3) A1 = EAREA(IS-2)/2.
   IF (IHC(IS-2).EQ.3) A3 = EAREA(IS-1)/2.
   DX1 = DLX(L)
   DX2 = DLX(LDN)
   DX3 = DLX(LOUT)
   DX4 = DLX(LDN-2)
   DX5 = DLX(LIN)
   DX6 = DLX(LDN+1)
   DX7 = DLX(LCOOL)
   DX9 = DLX(LJ)
   DX10= DLX(LDN-1)
C
   IF (IHC(ISUP).EQ.1) YIMFU = 1.0
   IF (IHC(ISUP).EQ.2) YCONVU = 1.0*ECPRT*AM2(ISUP)*(GAMC(ISUP) - 1.)/2.
   IF (IHC(ISUP).EQ.3) YFINSU = 1.0
IF (I.LT.NSTA-1) GO TO 390
C FOR THE LAST STATIONS IN THE TRAILING EDGE:
C
DX2 = 0.0
DX4 = 0.0
DX6 = 0.0
DX10= 0.0
390 CONTINUE
IF (IHC(IS).EQ.3) GO TO 420
IF (IHC(IS).EQ.2) GO TO 410
GO TO 340
C
C*** HCFRCD Computes HC for Forced Convection
C
410 CONTINUE
TMP = (T(2,ISLICE,LCOOL) + T(2,ISLICE,LIN))/2.
CALL GASTBL(TMP,C,CP,GAM,PD,R,XMU)
RE(IS) = 12.*3600.*ABS(WCROS(2,ISLICE,IS))*DH(IS)/(A(LCOOL)*XMU)
HC(IS) = 0.023*12.*(C/DH(IS))* (RE(IS)**.5)*(PD**.333)
AHTRN1 = (DX5 + DX6)*SPAN/2.
GO TO 340
C
C***** SUBROUTINE HCPTNS Computes HC for a Pin Fin Surface or for
C TURBULENT FORCED CONVECTION CHANNEL FLOW
C
420 CONTINUE
IF(ISENS.EQ.0) GO TO 424
AHTRN1 = EFAREA(IS)
HC(IS) = HC(IS-1)
GO TO 340
C
424 CALL HCPTNS(IS,DELTAN,LCOOL,LCUP,LIN,LCOOLP,PINS,EFAREA)
AHTRN1 = EFAREA(IS)
IF (IS.GE.NSTA-1) AHTRN1 = AHTRN1/2.
A2 = EFAREA(IS)/2.
A4 = EFAREA(IS+1)/2.
GO TO 340
C
C
C
430 CONTINUE
440 CONTINUE
450 CONTINUE
C
RETURN
END
C
C---- Source.NTCOFTT
SUBROUTINE TCOEF(IWRITE,WS,NIT,IPLOT,ALPH2)
C
DIMENSION POLD(15,80), PSAV(5), X(80), ALPH(4), DELTAN(15),
Z(TOTC(80), JSO(15)
DIMENSION PEXOLD(15), PIMOLD(15)
REAL*8 TCOF
COMMON /MATRX/ TCOF(400,30)
C
COMMON /PRPS/ CPO, GAMO, DP(80), SP(80), RE(80),
END
C
C
C
137
BTA = 0. INDICATES THAT A HEAT TRANSFER COEFFICIENT BOUNDARY CONDITION IS SPECIFIED
C 1. INDICATES THAT A HEAT FLUX IS SPECIFIED ON THE GAS BOUNDARY
C UNITS EXPECTED ON THE FOLLOWING INPUT DATA ARE:
C LENGTHS ARE ALL IN INCHES-- DLX, TAU, SPAN, DH, DPX
C TEMPERATURES ARE ABSOLUTE (R)
C MASS FLOWS ARE ALL IN (LBM/SEC),-- WCROS, WJ, WPC, WDUMP
C HEAT TRANSFER COEFFICIENTS ARE IN BTU/(HR*FT**2*R),--HG, HC--GAS
C SIDE AND COOLANT SIDE.
C HEAT FLUX QG IS IN BTU/(AR*FT**2)
C HEAT SINK, QSNK, IS IN BTU/(HR)
C THERMAL PROPERTIES ARE: CONDUCTIVITY-AKU-BTU/ (HR*FT*R),
C HEAT CAPACITY-CP-BTU/(LBM*R)
C
-----SET UP FIRST GUESS AT TEMPERATURE DISTRIBUTION
C-----ASSUME COOLANT TEMPERATURE IS CONSTANT, = PLENUM STATIC TEMPERATURE
C-----ASSUME METAL TEMPERATURE IS CONSTANT, = 2200. R

100 CONTINUE
   NODST = 5*NSTA
   W = NSTA - 1
   NSAVE = NODST - 10
   TSAVE = T(2,ISLICE,NSAVE)
   NODST = NODE NUMBER OF LAST FLOW CHANNEL NODE, AT EXIT OF TRAILING EDG
   NODSF = 5*NFWD
   N
120 CONTINUE
C
C-- FOR TRANSIENT CASES, ADJUST INITIAL GUESS OF PRESSURE DISTRIBUTION
C-- BASED ON THE VARIATION OF SUPPLY AND EXIT PRESSURES.
C IF (TYME.GT.0.0) GO TO 234
C
C-- FOR STEADY STATE, ONLY INITIALIZE P'S ON FIRST OVERALL LOOP.
C IF (NIT.GT.1) GO TO 255
PEXOLD(ISLICE) = PEXIT(ISLICE)
PIMOLD(ISLICE) = PIM
WRITE(6,165) (I,T(1,ISLICE,I),I=1,NODST)
FORMAT(1H1,'ASSUMED INITIAL TEMPERATURE DISTRIBUTION, (NODE NO.',NTCOFTT 5172
Z 'T) '/7('I3',',F8.2',')")
NTCOFTT 5173
170 CONTINUE
C
C---- PRESSURE INITIALIZATION, GIVEN PIM (PLENUM STATIC PRESSURE) AND PEXIT (GAS SIDE STATIC PRESSURE AT TRAILING EDGE), FIT PRESSURE TO NTCOFTT 5177
A
C---- CUBIC EQN. OF THE FORM A+B*X**3 , ASSUMING 85% OF THE PRESSURE DROP OCCURS IN THE TRAILING EDGE CHANNEL
C
X(1) = 0.0
X(2) = DLX(10)
DO 180 I = 3,NFWD
X(I) = DLX(5*I-1) + X(I-2)
180 CONTINUE
P(1,ISLICE,1) = PIM - (PIM-PEXIT(ISLICE))*0.15
P(1,ISLICE,N) = PEXIT(ISLICE)
P(1,ISLICE,N+1) = P(1,ISLICE,N)
P(1,ISLICE,NFWD-1) = P(1,ISLICE,1) - 2*(P(1,ISLICE,1) - PEXIT(ISLICE))
190 P(1,ISLICE,NFWD) = P(1,ISLICE,NFWD-1)
DO 200 I = 3,NFWD,2
P(1,ISLICE,I) = P(1,ISLICE,1) - (P(1,ISLICE,1) - P(1,ISLICE,NFWD))*(X(I)/X(NFWD))**3
200 CONTINUE
DO 230 I = ISTRT,IFNL,2
P(1,ISLICE,I) = P(1,ISLICE,1) - (P(1,ISLICE,1) - P(1,ISLICE,NFWD-1))**3
ISTRT = NFWD+1
IFNL = N
230 CONTINUE
C---- FOR TRAILING EDGE CHANNEL, X VALUES ARE RELATIVE TO END OF IMPINGEMENT CHANNEL
C
X(NFWD+1) = (DLX(NODSF) + DLX(NODSF-5))/2.
210 ITEM = NFWD+3
DO 220 I = ITEM,N,2
LCOOL = 5*I
X(I) = X(I-2) + DLX(LCOOL)
220 CONTINUE
DO 230 I = ISTRT,IFNL,2
P(1,ISLICE,I) = P(1,ISLICE,1) - (P(1,ISLICE,1) - PEXIT(ISLICE))**3
Z 
230 CONTINUE
P(1,ISLICE,I+1) = P(1,ISLICE,I)
DO 232 I = 1,NSTA
232 POLD(ISLICE,I) = P(1,ISLICE,I)
DO 233 I = 1,NSTA
233 P(2,ISLICE,I) = P(1,ISLICE,I)
GO TO 240
C
C DLTAPC = .84*(PIM-PIMOLD(ISLICE))
PIMOLD(ISLICE) = PIM
139
DLTAP = PEXIT(ISLICE) - PEXOLD(ISLICE)
PEXOLD(ISLICE) = PEXIT(ISLICE)
DO 235 I = 1,NFWD
235 P(2,ISLICE,I) = P(2,ISLICE,I) + DLTAPC
ISTRT = NFWD+1
IFNL = NSTA-1
DO 236 I = ISTRT,IFNL,2
P(2,ISLICE,I) = P(2,ISLICE,I) + DLTAPC*(1.0-X(I)/X(IFNL)) +
DLTAP*Z
Z
236 P(2,ISLICE,I+1) = P(2,ISLICE,I)
DO 237 I = 1,NFWD
237 POLD(ISLICE,I) = P(2,ISLICE,I)
GO TO 255
C
240 WRITE(6,245)
245 FORMAT(/' INITIAL PRESSURE DIST. (STATION NO.,P) '/)
WRITE(6,250) (I,P(1,ISLICE,I),I=ISTRT)
WRITE(6,250) (I,P(1,ISLICE,I),I=ITEM,N,2)
250 FORMAT(7(' ('I13,'','P7.2,')'))
C
255 CONTINUE
DO 260 I=1,4
260 PSAV(I)=0.0
C
C----INITIALLY, THE FLOW SPLIT AT THE LEADING EDGE IS ASSUMED
C TO BE 50/50 (DELTA=.5)
C---- IDELT COUNTS THE NUMBER OF FLOW SPLIT ITERATIONS. IF NO
C CONVERGENCE, IDELT IS SET NEGATIVE.
C---- DELTAN IS THE FRACTION OF FLOW TO THE SUCTION SIDE (EVEN
C NUMBERED STATIONS)
C----IVERGE COUNTS THE NUMBER OF ITERATIONS AT A GIVEN FLOW SPLIT
C IFNL = THE NUMBER OF FLOW CHANNEL NODES, USED IN PRESSURE CALCULATIONS
C
IF (NIT.EQ.1.AND.TYME.LT.0.0) DELTAN(ISLICE) = .5
275 CONTINUE
IFNL = NSTA - 3
IVERGE = 0
IDELT = 1
JS = 1
IF (NIT.GT.1) JS = JSO(ISLICE)
290 CONTINUE
IVERGE = IVERGE + 1
300 CONTINUE
C
JSENS = JS - 2*(JS/2)
C
C----SUBROUTINE FLOWS COMPUTES JET FLOW RATES, CROSSFLOW RATES, AND
C THE SQUARE OF THE MACH NUMBER
C
CALL FLOWS(JS,DELTA,ICH0KE,AMC0KE)
320 CONTINUE
IF (WJ(ISLICE,JS).LE.0.0) GO TO 370
C
C SUBROUTINE HCOOL COMPUTES IMPINGEMENT REGION HEAT TRANSFER COEFF'NTS

C CALL HCOOL(JS)
335 CONTINUE
C CALL THCON
C SUBROUTINE THCON EXTRACTS THERMAL CONDUCTIVITIES FROM INPUT
C TABLES AKCTBL AND AKWTBL.
C
335 CONTINUE
C
C--- CHECK TO MAKE SURE STAGNATION HC IS LESS THAN MAXIMUM PHYSICALLY
C POSSIBLE VALUE.
C
IF (JS.EQ.1) ASTG = (DLX(9)+DLX(14))*SPAN/2.0
IF (JS.GT.1) ASTG = (DLX(5*JS-1)+DLX(5*JS+9))*SPAN/2.0
HSTGMX = WJ(ISLICE,JS)*CFO*144.*3600./ASTG
IF (HC(JS).LT.HSTGMX) HC(JS) = HSTGMX
C
337 CONTINUE
C CALL TAPF4Y (JS,JSENS,DELTA)
C
340 CONTINUE
CALL GAUSS(NODST,23)
DO 350 I = 1,NODST
T(2,ISLICE,I) = TCOF(I,24)
IF (T(2,ISLICE,I).LT.0.0) T(2,ISLICE,I) = T0G
350 CONTINUE
C**************
360 IF (ABS((T(2,ISLICE,NSAVE)-TSAVE)/TSAVE).GT.05)
Z CALL FLOWS (JS,DELTA,ICHOK,AMCHOK)
TSAVE = T(2,ISLICE,NSAVE)
C**************
370 IF (ICHOK.EQ.0) GO TO 370
WRITE(8,'(8,355) ISLICE,IVERGE,IDENT,NI3,ICHOK,AMCHOK
Z I3',' I3',' NIT =',I3',' ICHKE =',I4',' M**2 =',F6.3)
370 CONTINUE
C COMPUTE NEW PRESSURES
CALL PBARAY(JS,JSENS,ICHOK)
C SOLVE THE TCFC ARRAY AND COMPUTE NEW PRESSURES
C
433 CONTINUE
CALL GAUSS(IFNL,19)
440 CONTINUE
DO 460 I = 1,IFNL
450 P(2,ISLICE,I) = TCOF(I,20)
460 CONTINUE
P(2,ISLICE,IFNL+1) = P(2,ISLICE,IFNL)
P(2,ISLICE,NSTA-1) = PEXIT(ISLICE)
P(2,ISLICE,NSTA) = PEXIT(ISLICE)
470 CONTINUE
IF (IWRITE.EQ.2) CALL WRUT(IDELT,JS,DELTA,IVERGE)
IF (IPLOT.EQ.2) CALL PLOTHP(ALPH2)
C
460 CONTINUE
C CHECK OVERALL CONVERGENCE
C CALCULATIONS ARE REPEATED UNTIL THE PRESSURE AT STATION 1 (NODE 5)
C HAS STABILIZED FOR FOUR ITERATIONS. THEN WE GO TO CHECK THE FLOW SPLIT.
C
DO 490 I=1,3
   K=5-I
PSAV(K)=PSAV(K-1)
PSAV(1)=P(2,ISLICE,JS)
C
DIFO=0.0
DO 500 I=1,3
   JJ=I+1
   DO 500 K=JJ,4
   DIFN=BEP(ISLICE,JS)-PSAV(K)
   IF (DIFO.LT.DIFN) DIFO=DIFN
CONTINUE
DIFO=DIFO/P(2,ISLICE,JS)
PCNVRG=0.01
IF (NIT.EQ.1.AND.NSLICE.GT.1) PCNVRG=.05
EPSN=(P(2,ISLICE,NFWD-1)-P(2,ISLICE,NFWD))/P(2,ISLICE,NFWD-1)
IF (IDELT.EQ.1.AND.IVERGE.LT.3.AND.TYFIE.LE.0) GO TO 516
IF (DIFO.LT.PCNVRG.AND.IVERGE.GT.4) GO TO 560
CONTINUE
IF (IVERGE.LT.1) GO TO 520
IF (ABS(P(2,ISLICE,1)-P(2,ISLICE,2)).GT.ABS(P(2,ISLICE,3)-P(2,ISLICE,4))) V=1.-V/2.
CONTINUE
DO 520 I=1,NSTAGE
   P(2,ISLICE,I)=P(2,ISLICE,I)+V*(POLD(ISLICE,I)-P(2,ISLICE,I))
P(2,ISLICE,I)=.999*PIM
   T(2,ISLICE,I)=T(2,ISLICE,I)*GAM(ISLICE,1)/AM2(ISLICE,1)
CONTINUE
IF (IVERGE.GT.30.OR.V.GT.0.95) WRITE(8,550) IVERGE,V,
   (PSAV(I),I=1,4); Z=IVERGE'=13';
550 FORMAT( ' ************* CONVERGENCE PROBLEMS **************'/
   ' Z = IVERGE=',$I13,'; V = ',F6.4,'; P(2,ISLICE,1)',F10.2服用PSAV=','4(F10.2))
   IF (IVERGE.GT.50) GO TO 590
GO TO 290
C
C********** ONCE PRESSURE-TEMPERATURE ITERATION HAS CONVERGED, CHECK
C THE FLOW SPLIT AND ADJUST IF NECESSARY.
C
CONTINUE
IF (IVERGE.GT.30.OR.V.GT.0.95) WRITE(8,550) IVERGE,V,
   (PSAV(I),I=1,4);
Z=IVERGE='13'; V='F6.4'; P(2,ISLICE,1)='4(F10.2)';
550 FORMAT( ' ************* CONVERGENCE PROBLEMS **************'/
   ' Z = IVERGE=',$I13,'; V = ',F6.4,'; P(2,ISLICE,1)',F10.2服用PSAV=','4(F10.2))
   IF (IVERGE.GT.50) GO TO 590
GO TO 290
C
CONTINUE
IF (WRITE.EQ.1) CALL WRWRT(IDELT,JS,DELTAN,IVERGE)
IF (ICHOK.EQ.0) WRITE(6,565) ICHOK,AMCHOK
565 FORMAT('/10X,'MACH NO. > 1 AT. STATION ',I4,',' M**2 = ',F6.3)/
CONTINUE
EPSN=(P(2,ISLICE,NFWD-1)-P(2,ISLICE,NFWD))/P(2,ISLICE,NFWD-1)
CONTINUE
IF (TYFIE.GT.0.0) GO TO 590
CALL FLSPUT(AJET,EPSN,ISLICE,NODSF,IDELT,JS,DELTAN,ICONV)
CONTINUE
IF (ICONV.EQ.1) CALL WRWRT(IDELT,JS,DELTAN,IVERGE)
IVERGE=0
IF (ICONV.EQ.0) GO TO 290

142
CONTINUE
IF (TIME.GT.0.0) CALL WROUT(IDELT,JS,DELTAN,IVERGE)
JSO(ISLICE) = JS
RETURN
END

C---- SOURCE.NTHRCNT
SUBROUTINE THRCON
C
C SOURCE.NTHRCNT----

COMMON /BOUND/ BCXS(400), BCXP(400), BCHGS(1000), BCHGP(1000), NTHRCNT 5401
Z BCTGS(1000), BCTGP(1000), BCQGS(1000), BCQGP(1000), NTHRCNT 5402
Z JHUBIN(400), JHUB(80), TTIPIN(400), TTIP(80), NTHRCNT 5403
Z QTIPIN(400), QTIP(80), RHOGV(400), PEX(400), NTHRCNT 5404
Z BCTIME(50), TTIO(50), WPLEN, NTHRCNT 5405
Z WSVST(50), AKCTBL(20), AKWTLB(20), NBCS, NBCP NTHRCNT 5406

COMMON /TCO/ ADUMP, BTA, CD, CP, NTHRCNT 5407
Z JAM, EIM, R, SPIN, TOG, NTHRCNT 5408
Z JAM, WIM, AKC(15,80), AKW(15,80), NTHRCNT 5409
Z A(400), AJET(80), AM2(80), CNUM(80), NTHRCNT 5410
Z DH(80), DHF(80), DHJ(80), NTHRCNT 5411
Z DLX(400), FF(80), HG(80), NTHRCNT 5412
Z P(2,15,80), PEXIT(15), S(15), T(2,15,400), NTHRCNT 5413
Z JSXK(80), FE(80), T(2,15,400), NTHRCNT 5414
Z IG(6C), TAU(6C), WFC(6C), NTHRCNT 5415
Z WJ(15,80), WCRS(2,15,80), XN(60), NTHRCNT 5416
Z ICOR, IFILM, IHUB, ITIP, NTHRCNT 5417
Z ISSBLOK, NBLKSZ, NSLICE, NTHRCNT 5418
Z NFWD, NSTA, IH(80), NTHRCNT 5419

DO 100 I = 1,NSTA
L = 5*I - 2
LJ = L - 1
LOUT = L - 2
TC = (T(2,ISLICE,LJ) + T(2,ISLICE,LOUT))/2.C - 46C. NTHRCNT 5420
T* = T(2,ISLICE,L) - 46C. NTHRCNT 5421

LOOK UP COATING THERMAL CONDUCTIVITY IN TABLE AKCTBL.

IF (TC.GT.AKCTBL(1)) GO TO 150

FOR A TEMPERATURE LOWER THAN THE BOTTOM OF THE TABLE, EXTRAPOLATE
BELOW TABLE.
RATIO = (TC - AKCTBL(1))/(AKCTBL(3) - AKCTBL(1))
AKC(ISLICE,L) = AKCTBL(1) + (AKCTBL(4) - AKCTBL(2))*RATIO
GO TO 500

CONTINUE

C FIND SIZE OF TABLE
DO 152 J = 3,10,2
JLST = J-1
IF (AKCTBL(J).LE.0.1) GO TO 156
152 CONTINUE
156 JLSTM = JLST-1 ' LOCATE WHERE TEMPERATURE FALLS IN THE TABLE AKCTBL.
C DO 170 J = 3, JLIST, 2
   IF (TC.GT.AKCTBL(J)) GO TO 160
C FOUND LOCATION, NOW INTERPOLATE.
C RATIO = (TC - AKCTBL(J-2))/(AKCTBL(J) - AKCTBL(J-2))
   AKW(ISLICE,I) = AKCTBL(J-1) + (AKCTBL(J+1) - AKCTBL(J-1)) * RATIO
   GO TO 500
160 IF (J.LT.JLIST) GO TO 170
C TEMPERATURE IS ABOVE THE RANGE OF THE TABLE, SO EXTRAPOLATE UP.
C RATIO = (TC - AKCTBL(J-2))/(AKCTBL(J) - AKCTBL(J-2))
   AKW(ISLICE,I) = AKCTBL(J-1) + (AKCTBL(J+1) - AKCTBL(J-1)) * RATIO
   GO TO 500
170 CONTINUE
500 CONTINUE
C NOW LOOK UP METAL CONDUCTIVITY IN TABLE AKWTBL.
C IF (TW.GT.AKWTBL(1)) GO TO 550
C FOR A TEMPERATURE LOWER THAN THE BOTTOM OF THE TABLE, EXTRAPOLATE
C BELOW TABLE
C RATIO = (TW - AKWTBL(1))/(AKWTBL(3) - AKWTBL(1))
   AKW(ISLICE,I) = AKWTBL(2) + (AKWTBL(4) - AKWTBL(2)) * RATIO
   GO TO 1000
550 CONTINUE
C FIND SIZE OF TABLE
C DO 552 J = 3, 19, 2
   JLIST = J-1
   IF (AKWTBL(J).LE.0.1) GO TO 554
552 CONTINUE
554 JLIST = JLIST - 1
C LOCATE WHERE TEMPERATURE FALLS IN THE TABLE AKWTBL.
C DO 570 J = 3, JLIST, 2
   IF (TW.GT.AKWTBL(J)) GO TO 560
C FOUND LOCATION, NOW INTERPOLATE.
C RATIO = (TW - AKWTBL(J-2))/(AKWTBL(J) - AKWTBL(J-2))
   AKW(ISLICE,I) = AKWTBL(J-1) + (AKWTBL(J+1) - AKWTBL(J-1)) * RATIO
   GO TO 1000
560 IF (J.LT.JLIST) GO TO 570
C TEMPERATURE IS ABOVE THE RANGE OF THE TABLE, SO EXTRAPOLATE UP.
C RATIO = (TW - AKWTBL(J-2))/(AKWTBL(J) - AKWTBL(J-2))
   AKW(ISLICE,I) = AKWTBL(J-1) + (AKWTBL(J+1) - AKWTBL(J-1)) * RATIO
   GO TO 1000
570 CONTINUE
1000 CONTINUE
RETURN
END
SUBROUTINE WROUT (IDELT, JS, DELTAN, IVERGE)

C DUM2 CARRIES THE IMPINGEMENT JET REYNOLDS NO. IN FROM HCCOOL.

COMMON /CHKHOL/ WCHK(80), WCHKDM

COMMON /FLMCOL/ RHOGA(60), FG(80), XFC(80), FLMEFF(80),
Z XMUC(80), EMES(80), REFC(80), NFCSUP(80)

COMMON /PPPS/ CPO, GAMO, DP(80), SP(80), RE(80),
Z CPC(80), CMC(80), DUMR1(80), DUMR2(80)

COMMON /RADL/ APLN(15), DPLN(15), RIN(15), ROUT(15),
Z PIN(15), TIN(15), W(15), WS

COMMON /TCO/ ADUMP, BT, CD, SP, CP,
Z GAM, FIM, R, SPAN, TOG,
Z WDUMP, WIM, AKC(15,80), AKW(15,80),
Z A(400), B(JET(80), AM2(80), CNUM(80),
Z DH(80), DHP(80), DHJ(80),
Z DLX(400), PP(80), HC(80), HG(80),
Z P(2,15,80), PEH(15), PUMP(80), QG(80),
Z QSNK(80), RR(80), S(15), T(2,15,400),
Z TG(80), TAU(400), WFC(80),
Z TJ(15,80), WCOS(2,15,80), XN(80),
Z ICOI, IFILM, IHub, ITIP,
Z ISBLOK, ISLICE, NBLKSN, NSLICE,
Z NFWD, NSTA, IH(80)

COMMON /TRNSST/ RAOC, RHOIT, SFHTC, SPHT3,
Z DLTYME, TYME, TEPS, TMMAX

COMMON /UNITS/ CINCH(2), CHTC(2), CHFLX(2), CPRSR(2), CMSFL(2),
Z CTMPF(2), CTCGN(2), CDEN(2), CSHRT(2), CGASC(2),
Z CVISC(2), CHOOGV(2), IUNITS

DIMENSION DUM1(10),DUM2(10),DELTAN(15)

10 CONTINUE
IF (ISLICE.EQ.1) TBULK = 0.0
IF (ISLICE.EQ.1) TOTSFN = 0.0
TTYME = TYME
IF (TTYME.LT.0.) TTYME = 0.0
WRITE(6,90) TTYME,DLTYME,WS

90 FORMAT(1H1,10X,'TIME = ',F6.2,' SEC., STEP SIZE = ',F6.3,
Z ' SEC., WHEEL SPEED =',F8.1,' RPM')

9805 WRITE(6,8806) ISLICE, IDELT, JS, DELTAN(ISLICE), IVERGE
ITPBG = NFWD + 2
WRITE(6,1600) ITRBG

C

IF (IUNITS.EQ.2) WRITE(6,270)
IF (IUNITS.EQ.1) WRITE(6,271)
DO 210 I = 1,NSTA,2
II = I
LCool = 5*II
NOS = LCool - 4
DO 205 J = 1,4
JM = NOS+J-1
DUM1(J) = T(2,ISLICE,JM) - UTG.

IF (IUNITS.EQ.1) DUM1(J) = T(2,ISLICE,JM)/CTMPF(1)
CONTINUE

DUM1(5) = T(2,ISLICE,LCOOL) - 460.

IF (IUNITS.EQ.1) DUM1(5) = T(2,ISLICE,LCOOL)/CTMPF(1)

DUM1(6) = P(2,ISLICE,II)/CPRSFL(IUNITS)

DUM1(7) = DUM1(6)*((1.+(GAMC(II)-1.)*AM2(II)/2.)*(GAMC(II)/

Z)

DUM1(8) = HC(II)/CHTC(IUNITS)

DUM1(9) = HG(II)/CHTC(IUNITS)

DUM1(10) = TG(II)/1.8

IF (BTA.GT.0.001) DUM1(10) = 5.E20

IF (BTA.GT.0.001) DUM1(9) = 4.E20

WRITE(6,274) (II,LCOOL, (DUM1(J),J=1,10))

IF (I.EQ.NFWD) WRITE(6,276)

DO 220 I = 1,NSTB,2

II = I

LCOOL = 5*II

NOS = LCOOL - 4

DUM2(1) = WJ(ISLICE,II)/CMSFL(IUNITS)

DUM2(2) = DUM22(II)

DUM2(3) = WCROS(2,ISLICE,II)/CMSFL(IUNITS)

DUM2(4) = RE(II)

DUM2(5) = SQRT(AM2(II))

DUM2(6) = FF(II)

DUM2(7) = WPC(II)/CMSFL(IUNITS)

DUM2(8) = FMLMFL(II)

WRITE(6,280) (II,LCOOL,DUM2(1),WCHK(II),(DUM2(J),J=2,8))

IF (I.EQ.NFWD) WRITE(6,276)

CONTINUE

DUM2(9) = WDUMP/CMSFL(IUNITS)

IF (ADUMP.GT.0.0.AND.IUNITS.EQ.2) WRITE(6,290) DUM2(9),WCHKDM

IF (ADUMP.GT.0.0.AND.IUNITS.EQ.1) WRITE(6,291) DUM2(9),WCHKDM

ITRBG = NFWD + 1

WRITE(6,124) ISLICE,ITRBG

IF (IUNITS.EQ.2) WRITE(6,284)

IF (IUNITS.EQ.1) WRITE(6,279)

DO 230 I = 1,NSTA,2

II = I

IF (I.GT.1) II = I-1

LCOOL = 5*II

NOS = LCOOL - 4

DO 225 J = 1,5

JM = NOS+J-1

DUM1(J) = T(2,ISLICE,JM) - 460.

IF (IUNITS.EQ.1) DUM1(J) = T(2,ISLICE,JM)/CTMPF(1)

CONTINUE

DUM1(6) = P(2,ISLICE,II)/CPRSFL(IUNITS)

DUM1(7) = DUM1(6)*((1.+(GAMC(II)-1.)*AM2(II)/2.)*(GAMC(II)/

Z)

DUM1(8) = HC(II)/CHTC(IUNITS)

DUM1(9) = HG(III)/CHTC(IUNITS)

DUM1(10) = TG(II) - 460.

IF (IUNITS.EQ.1) DUM1(10) = TG(II)/1.8

146
IF (BTA.GT..001) DUM1(10) = 9.E20
IF (BTA.GT..001) DUM1(9) = 9.E20
WRITE(6,274) (II,LCOOL, (DUM1(J),J=1,10))
IF (I.EQ.NFYD) WRITE(6,276)
CONTINUE
IF (IUNITS.EQ.2) WRITE(6,278)
IF (IUNITS.EQ.1) WRITE(6,279)
DO 240 I = 1,NSTA2
II = I
IF (I.GT.1) II = 11
LCOOL = 5*II
NOS = LCOOL - 4
DUM2(1) = WJ(ISLICE,II)/CMSFL(IUNITS)
DUM2(2) = DUM2(II)
DUM2(3) = WCROS(2,ISLICE,II)/CMSFL(IUNITS)
DUM2(4) = RE(II)
DUM2(5) = SQR(T AM2(II))
DUM2(6) = FF(II)
DUM2(7) = WFC(II)/CMSFL(IUNITS)
DUM2(8) = EFMEFF(II)
WRITE(6,280) (II,LCOOL,DUM2(1),WCHK(II), (DUM2(J),J=2,8))
IF (I.EQ.NFYD) WRITE(6,276)
CONTINUE
DUM2(9) = WDMPLCMSFL(IUNITS)
IF (ADUMP.GT.0.0.AND.IUNITS.EQ.2) WRITE(6,290) DUM2(9),WCHKM
IF (ADUMP.GT.0.0.AND.IUNITS.EQ.1) WRITE(6,251) DUM2(9),WCHKM
1000 CONTINUE
C TO DETERMINE THE HEAN OUTSIDE SURFACE TEMPERATURE FOR EACH
C SIDE OF THE BLADE, AND
C TO LOCATE THE EXTREME TEMPERATURE POINTS, BOTH HIGH AND LOW.
C
XTOT = 0.
XTOTMD = 0.0
HBAR = HC(I)*(DLX(2)+DLX(3))/2.
TBAR = T(2,ISLICE,1)*(DLX(5)+DLX(11))/2.
TBARMD = T(2,ISLICE,3)*(DLX(8)+DLX(13))/2.
ISTAT = 1
DO 1004 I = 2,NSTA2
NODN = 5*I-2
ISTATD = ISTAT + 5
ISTATD = ISTAT + 10
TBARMD = TBARMD + T(2,ISLICE,NODN)*(DLX(NODM)+DLX(NODM*10))/2.
XTOTMD = XTOTMD + DLX(NODM)
TBAR = TBAR + T(2,ISLICE,ISTAT)*(DLX(ISTAT)+DLX(ISTATD))/2.
XTOT = XTOT + DLX(ISTAT)
HBAR = HBAR + HC(I)*(DLX(ISTAT)+DLX(ISTATD))/2.
C
1004 CONTINUE
IF (IUNITS.EQ.1) TBAR = TBAR/(1.8*XTOT)
IF (IUNITS.EQ.2) TBAR = TBAR/XTOT - 460.
IF (IUNITS.EQ.1) TBARMD = TBARMD/(1.8*XTOTMD)
IF (IUNITS.EQ.2) TBARMD = TBARMD/XTOTMD -460.
TBULK = TBULK + TBARMD*S(ISLICE)
TOTS PN = TOTS PN + S(ISLICE)
HBAR = HBAR/(XTOT*CHT(IUNITS))
C
1008 IF (IUNITS.EQ.2) WRITE(6,1115) TBAR,TBARMD,HBAR
IF (IUNITS.EQ.1) WRITE(6,1116) TBAR,TBARMD,HBAR
C
TSMAX = T(2,ISLICE,1) - 460.
TSMIN = T(2,ISLICE,1) - 460.
TPMAX = T(2,ISLICE,1) - 460.
TPMIN = T(2,ISLICE,1) - 460.
ISUCMX = 1
ISUCMN = 1
IPRSMX = 1
IPRSMN = 1
IPRES = 1
ISUCT = -4

DO 1080 I = 3, NSTA, 2
IPRES = IPRES + 10
ISUCT = ISUCT + 10

C IF (T(2,ISLICE,IPRES)-460..GT.TPMAX) GO TO 1030
IF (T(2,ISLICE,IPRES)-460..LT.TPMIN) GO TO 1040
GO TO 1050

1030 TPMAX = T(2,ISLICE,IPRES)-460.
IPRSMX = I
GO TO 1050

1040 TPMIN = T(2,ISLICE,IPRES)-460.
IPRSMN = I
GO TO 1050

1050 IF (T(2,ISLICE,ISUCT)-460..GT.TSMAX) GO TO 1060
IF (T(2,ISLICE,ISUCT)-460..LT.TSMIN) GO TO 1070
GO TO 1080

1060 TSMAX = T(2,ISLICE,ISUCT)-460.
ISUCMX = I - 1
GO TO 1080

1070 TSMIN = T(2,ISLICE,ISUCT)-460.
ISUCMN = I - 1
GO TO 1080

1080 CONTINUE

C IF (ISLICE.LT.NSLICL) GO TO 1095
TBULK = TBULK/TOTSPN
IF (IUNITS.EQ.1) WRITE(6,1091) TYLE,TBULK
IF (IUNITS.EQ.2) WRITE(6,1090) TYLE,TBULK

1090 FORMAT(1H2,30X,'TIME =',F6.3,' SEC., OVERALL BULK TEMPERATURE =',F7.1,' F')
1091 FORMAT(1H2,30X,'TIME =',F6.3,' SEC., OVERALL BULK TEMPERATURE =',F7.1,' K')
1095 CONTINUE

1115 FORMAT(/12X,'EXTRMRS CF OUTEE SPACE TEMPERATUReS (F)'/6X,'(F7.1')
1116 FORMAT(/12X,'EXTREMES OF OUTER SURFACE TEMPERATURES (F)'/6X,'F7.1',' AT STATION ',I2,'.',F7.1,' AT STATION ',I2/6X,'',F7.1,' AT STATION ',I2/6X,'F7.1')
1125 FORMAT (/12X,'EXTREMES OF OUTER SURFACE TEMPERATURES (K)'/6X, 
'I,' PRESSURE SIDE: ', F7.1,' AT STATION ',I2,', ',F7.1,' AT STATION ',I2) NWROTTT 5766

WRITE(6,1125) TPHAX,IPRSMX,TPMIN,IPRSNN,TSMAX,ISUCHX,TSUIN,ISUCHN NWROTTT 5768

1125 FORRAT(/12X, 'EXTREMES OF OUTER SURFACE TEMPERATURES (K)'/6X, NWROTTT 5769

1130 CONTINUE NWROTTT 5770

1GO FORMAT(30X,'PRESSURE SIDE
2130ATED EDGE REGION BEGINS AT ',NWROTTT 5771

5f z

124 FORMAT(H1,' SLICE NO.',I2,', FLOW SPLIT NO.',I3,', SPLIT AT ',NWROTTT 5774

Z

8806 FOPHAT(' SLICE NO.',I2,', FLOW SPLIT NO.',I3), NWROTTT 5775

8806 NUMRAT(/ NWBOTTT 5776

Z

270 FORMAT(/ NWROTTT 5777

Z

271 FORMAT(/ NWROTTT 5778

Z

274 FORMAT(I6,2X,I6,1X,7F10.1,3F11.1) NWROTTT 5779

276 FORMAT(47X,'BEGIN TRAILING EDGE REGION') NWROTTT 5780

278 FORMAT(IH2,' STATION * COOLANT * OUTSIDE * INTERFACE* MID-WALL* INSIDE * ',NWROTTT 5781

Z

Z

279 FORMAT(IH2,' STATION * COOLANT * IMP. FLOW * RE-NO. * CROSSFLOW',NWROTTT 5782

Z

Z

280 FORMAT(IH ,I5,5X,5X,'F9.6,5X,5X,F9.6,5X,F9.6,4X,F9.6)' NWROTTT 5783

280 NUMRAT(/ NWBOTTT 5784

Z

290 FORMAT(' FLOW DUMPED DIRECTLY INTO TRAILING EDGE REGION IS ',NWROTTT 5785

Z

291 FORMAT(' FLOW DUMPED DIRECTLY INTO TRAILING EDGE REGION IS ',NWROTTT 5786

Z

RETURN NWROTTT 5787

END NWROTTT 5788
REFERENCES


<table>
<thead>
<tr>
<th>Subroutine name</th>
<th>Source module</th>
<th>COMMON blocks</th>
<th>Called subroutines</th>
<th>Calling subroutines</th>
<th>Subroutine name</th>
<th>Source module</th>
<th>COMMON blocks</th>
<th>Called subroutines</th>
<th>Calling subroutines</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCK DATA</td>
<td>NGASDAT</td>
<td>/GAAS/</td>
<td>NONE</td>
<td>NONE</td>
<td>PARRAY</td>
<td>NPARAYT</td>
<td>/MATRIX/</td>
<td>NONE</td>
<td>TCOEF</td>
</tr>
<tr>
<td>FLOWS</td>
<td>NFLOEST</td>
<td>/CHKHOL/</td>
<td>GASTBL</td>
<td>TCOEF</td>
<td>FLNUM</td>
<td>NPLENMP</td>
<td>/RADL/</td>
<td>GASTBL</td>
<td>MAIN PROGRAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/FLMCOL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/PRPS/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FLSPLT</td>
<td>NFLSPLP</td>
<td>NONE</td>
<td>NONE</td>
<td>TCOEF</td>
<td>PLOTMF</td>
<td>NPLOTIT</td>
<td>/PRPS/</td>
<td>NONE</td>
<td>TCOEF</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/SPECL/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GASTBL</td>
<td>NGASTB</td>
<td>/GAAS/</td>
<td>NONE</td>
<td>TCOEF</td>
<td>PREP</td>
<td>NPREPAT</td>
<td>/BOUND/</td>
<td>NONE</td>
<td>INPRT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/FLMCOL/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/PRPS/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/SPECL/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GAUSS</td>
<td>NGAUS</td>
<td>/MATRX/</td>
<td>NONE</td>
<td>TCOEF</td>
<td>TARRAY</td>
<td>NTARRAY</td>
<td>/BOUND/</td>
<td>GASTBL</td>
<td>HCOOL</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/FLMCOL/</td>
<td>HCFRCD</td>
<td>HCPINS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/PRPS/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/SPECL/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GETIN</td>
<td>NGETINT</td>
<td>/BOUND/</td>
<td>INPRT</td>
<td>MAIN PROGRAM</td>
<td>HCFRCD</td>
<td>NHCFRCT</td>
<td>/TCO/</td>
<td>GASTBL</td>
<td>HCOOL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/FLMCOL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td>TARRAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/IMPCOR/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/RADL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/SPECL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HCOOL</td>
<td>NHCOOLT</td>
<td>/IMPCOR/</td>
<td>GASTBL</td>
<td>TCOEF</td>
<td>TCOEF</td>
<td>NTCOPTT</td>
<td>/BOUND/</td>
<td>FLOWS</td>
<td>MAIN PROGRAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/PRPS/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/PRPS/</td>
<td>HCOOL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/RADL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/PRPS/</td>
<td>HCFRCD</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/SPECL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/SPECL/</td>
<td>HCPINS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HCPINS</td>
<td>NHCPINT</td>
<td>/PRPS/</td>
<td>GASTBL</td>
<td>TARRAY</td>
<td>THRCNT</td>
<td>NTHRCNT</td>
<td>/BOUND/</td>
<td>NONE</td>
<td>TCOEF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/RADL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/SPECL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INPRT</td>
<td>NINPRTT</td>
<td>/BOUND/</td>
<td>PREP</td>
<td>GETIN</td>
<td>WROUT</td>
<td>NWROTTT</td>
<td>/CHKHOL/</td>
<td>NONE</td>
<td>TCOEF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/GAAS/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/FLMCOL/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/RADL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/PRPS/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/SPECL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/RADL/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAIN PROGRAM</td>
<td>NTACT</td>
<td>/BOUND/</td>
<td>GETIN</td>
<td>NONE</td>
<td>WROUT</td>
<td>NWROTTT</td>
<td>/CHKHOL/</td>
<td>NONE</td>
<td>TCOEF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/FLMCOL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/FLMCOL/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/GAAS/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/PRPS/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/RADL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/RADL/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/SPECL/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/SPECL/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TCO/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/TRANSNT/</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

151
<table>
<thead>
<tr>
<th>Subroutine</th>
<th>COMMON block</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BOUND</td>
</tr>
<tr>
<td>BLOCK DATA</td>
<td></td>
</tr>
<tr>
<td>FLOWS</td>
<td></td>
</tr>
<tr>
<td>FLSPLT</td>
<td></td>
</tr>
<tr>
<td>GASTBL</td>
<td></td>
</tr>
<tr>
<td>GAUSS</td>
<td></td>
</tr>
<tr>
<td>GETIN</td>
<td>x</td>
</tr>
<tr>
<td>HCFRCD</td>
<td></td>
</tr>
<tr>
<td>HCOOL</td>
<td></td>
</tr>
<tr>
<td>HCPINS</td>
<td></td>
</tr>
<tr>
<td>INPRT</td>
<td>x</td>
</tr>
<tr>
<td>NTTACT</td>
<td>x</td>
</tr>
<tr>
<td>PARRAY</td>
<td></td>
</tr>
<tr>
<td>PLNUM</td>
<td></td>
</tr>
<tr>
<td>PLOTMF</td>
<td></td>
</tr>
<tr>
<td>PREP</td>
<td>x</td>
</tr>
<tr>
<td>TARRAY</td>
<td>x</td>
</tr>
<tr>
<td>TCOEF</td>
<td></td>
</tr>
<tr>
<td>THRCON</td>
<td>x</td>
</tr>
<tr>
<td>WROUT</td>
<td></td>
</tr>
</tbody>
</table>
Figure 1. - Overall division of blade into slices.

Figure 2. - Blade geometric model.
Figure 3. - Overall program procedure.
Figure 4. - Subroutine calling relations.
Interpolate in transient-boundary-condition tables: PTIN, TTIN, WS, PEXIT, RHOVGA, QHUB, THUB, OTIP, TTIP

Begin marching up blade do 1000. i=1, NSLICE

Call PLNUM to calculate coolant conditions

Call PREP to prepare input data for current slice

Call TCOEF to calculate flows, temperatures, and pressures

Calculate amount of coolant remaining

1000

Check overall coolant balance

1100

Adjust coolant-inlet conditions

Job complete

Figure 5. - Flow chart of main program.

Figure 6. - Coolant-channel mass balance.
Figure 7. - Flow chart for subroutine FLSPLT.
Figure 7. - Continued.
Figure 7. - Continued.
QY,

Print error message

Terminate program

DELTAN<0.5

EPLAST=EPSN

JSNCH=0

Figure 7. - Concluded.

(a) Detail of station numbering at end of impingement insert.

(b) Detail of flow-split point.

Figure 8. - Details of flow-split parameters.
A computer program to calculate transient and steady-state temperatures, pressures, and coolant flows in a cooled axial-flow turbine blade or vane with an impingement insert is described. Coolant-side heat-transfer coefficients are calculated internally in the program, with the user specifying either impingement or convection heat transfer at each internal flow station. Spent impingement air flows in a chordwise direction and is discharged through the trailing edge and through film-cooling holes. The ability of the program to handle film cooling is limited by the internal flow model. Input to the program includes a description of the blade geometry, coolant-supply conditions, outside thermal boundary conditions, and wheel speed. The blade wall can have two layers of different materials, such as a ceramic thermal-barrier coating over a metallic substrate. Program output includes the temperature at each node, the coolant pressures and flow rates, and the coolant-side heat-transfer coefficients.