COMPUTER PROGRAM FOR DESIGN OF TWO-DIMENSIONAL SUPERSONIC TURBINE ROTOR BLADES WITH BOUNDARY-LAYER CORRECTION

by Louis J. Goldman and Vincent J. Scullin

Lewis Research Center
Cleveland, Ohio 44135
A FORTRAN IV computer program for the design of two-dimensional supersonic rotor blade section corrected for boundary-layer displacement thickness is presented. The ideal rotor is designed by the method of characteristics to produce vortex flow within the blade passage. The boundary-layer parameters are calculated by Cohen and Reshotoko's method for laminar flow and Sasman and Cresci's method for turbulent flow. The program input consists essentially of the blade surface Mach number distribution and total flow conditions. The primary output is the corrected blade profile and the boundary-layer parameters.
A FORTRAN IV computer program for the design of two-dimensional supersonic rotor blade sections corrected for boundary-layer displacement thickness is presented. The ideal rotor is designed by the method of characteristics to produce vortex flow within the blade passage. The boundary-layer parameters are calculated by Cohen and Reshotko's method for laminar flow and Sasman and Cresci's method for turbulent flow. The program input consists essentially of the blade surface Mach number distribution and total flow conditions. The primary output is the corrected blade profile and the boundary-layer parameters.

INTRODUCTION

Methods for the design of supersonic turbines for possible use in turbopump and open-cycle auxiliary power systems, where high-energy fluids are used and high pressure ratios are available, have recently become of interest. The method of characteristics as applied to the two-dimensional isentropic flow of a perfect gas can be used for the design of the supersonic blading. Computer programs for the isentropic design of two-dimensional supersonic nozzles and rotor blade sections have been described in references 1 and 2, respectively. A computer program for the design of two-dimensional supersonic nozzles with boundary-layer correction is described in reference 3.

This report presents a computer program for the design of supersonic rotor blades where losses are accounted for by correcting the rotor profile for boundary-layer displacement thickness. The ideal rotor blade profile is obtained by using the computer program described in reference 2. Boundary-layer parameters are calculated by using the computer program described in reference 4. The final rotor blade profile is then
obtained by adding the displacement thicknesses to the ideal blade coordinates. The program described herein is essentially a modified combination of the two programs described in references 2 and 4.

The boundary-layer parameters (displacement and momentum thicknesses) are also used to calculate the conditions downstream of the rotor after the flow has mixed to a uniform state. The procedure described in reference 5 is used for this purpose.

This report presents a description of the input and output and a complete FORTRAN IV listing of the program. A brief description of the computer program and method of design is also given. An example of the program output is included to indicate the use of the program and the results obtainable.

**METHOD OF ANALYSIS**

The design of two-dimensional supersonic rotor blades that are corrected for boundary-layer displacement thickness is described herein. The ideal rotor passage is designed by the method of characteristics as applied to the isentropic flow of a perfect gas. Boundary-layer parameters (displacement and momentum thicknesses) are then calculated for the ideal blades. The final rotor blade profile is obtained by adding the displacement thicknesses to the ideal rotor coordinates.

**Rotor Blade Description and Design**

The design of the ideal blade passage is based on establishing vortex flow within the passage by a procedure analogous to that given in reference 2. A typical passage is shown in figure 1. The passage consists essentially of three major parts: (1) inlet transition arcs, (2) circular arcs, and (3) outlet transition arcs. The inlet transition arcs (upper and lower) are required to convert the assumed uniform parallel flow at the passage inlet into vortex flow. The concentric circular arcs turn and maintain the vortex flow condition. The outlet transition arcs reconvert the vortex flow into uniform parallel flow at the passage exit. Straight-line segments parallel to the inlet and outlet flow direction complete the ideal passage.

As seen from figure 1, the ideal passage is designed so that the outlet spacing is less than the inlet spacing. This is necessary if the corrected passage (i.e., the passage corrected by the boundary-layer displacement thicknesses) is to have equal outlet and inlet spacing (see fig. 2). For an ideal passage designed for impulse conditions (equal inlet and outlet Mach numbers), this is accomplished by having less circular turning for the outlet portion of the passage. That is, the outlet circular arcs JK and DE (fig. 1) are less than the corresponding inlet circular arcs IJ and CD. An iterative
Figure 1. - Design of ideal supersonic flow passage by method of characteristics.

Figure 2. - Design of supersonic rotor blade section.
procedure performed by the computer program determines the condition for which the corrected passage has equal inlet and outlet spacings.

**Boundary-Layer Calculations**

The boundary-layer parameters (displacement and momentum thicknesses) are calculated for the ideal rotor passage by using the computer program described in 4. The program uses Cohen and Reshotko's method (ref. 6) for the calculation of laminar boundary layers and Sasman and Cresci's method (ref. 7) for turbulent boundary layers. Curvature effects are not considered in these calculations.

In the laminar regime, a single ordinary differential equation (the momentum integral equation) is solved numerically. The results of this method (as explained in ref. 6) have to be extended for flows in highly favorable pressure gradients as might occur over some portions of the rotor blade. For turbulent flow, coupled first-order ordinary differential equations (the momentum and moment-of-momentum integral equations) are solved using Runge-Kutta techniques.

The displacement thicknesses obtained from the program are then added to the ideal rotor blade coordinates to obtain the corrected blade profile. As discussed previously, an iterative procedure is performed by the computer program to determine the conditions for which the corrected passage has equal inlet and outlet spacings. It should be noted that it is not always possible to obtain a blade design by this procedure because of boundary-layer separation or large boundary-layer growth.

A number of options related to the boundary-layer calculations are also available to the user. Because of the adverse pressure gradients that exist within the rotor passage, the computer program will generally predict laminar separation at these locations. The program allows for the reattachment of the flow and continuation of the calculations for turbulent flow, if this is desired. The user can also, if he wishes, force transition to turbulent flow at any point of the calculations, including the inlet.

**Aftermixing Conditions**

The displacement and momentum thickness at the rotor exit (station 1, fig. 2) can be used to calculate the aftermixing conditions downstream of the rotor assuming that the flow mixes to a uniform state. Application of the continuity, momentum, and energy equations between stations 1 and 2 (fig. 2) results in the determination of aftermixing Mach number, flow angle, pressure ratio, and kinetic energy loss. The calculation procedure has been described in reference 5.
Subsonic and supersonic aftermixing axial Mach number solutions are possible for this loss model when the free-stream axial Mach number at the blade exit (before mixing) is supersonic. The subsonic solution corresponds to mixing plus oblique shock losses, whereas the supersonic solution corresponds to shockless mixing. A more detailed discussion of the different solutions can be found in reference 8.

DESCRIPTION OF INPUT

A description of the input for the FORTRAN IV computer program is given in this section. The input consists primarily of rotor inlet and outlet Mach number, upper- and lower-surface Mach number, inlet flow angle, specific-heat ratio, and total flow conditions. Either U.S. customary units or the International System of units may be used for input. This option is controlled by the input variable KEM. The program gas properties are set up for air. For gases other than air the changes required to the program are described in appendix A.

The input format is shown in table I. The input variables are:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETAN</td>
<td>inlet flow angle, $\beta_i$, deg</td>
</tr>
<tr>
<td>VIN</td>
<td>inlet Prandtl-Meyer angle, $\nu_i$, deg</td>
</tr>
<tr>
<td>VLOW</td>
<td>lower-surface Prandtl-Meyer angle, $\nu_l$, deg</td>
</tr>
<tr>
<td>VUP</td>
<td>upper-surface Prandtl-Meyer angle, $\nu_u$, deg</td>
</tr>
<tr>
<td>VOUT</td>
<td>outlet Prandtl-Meyer angle, $\nu_o$, deg</td>
</tr>
<tr>
<td>BETAT</td>
<td>initial estimate of outlet flow angle, $\beta_o$, deg (in the absence of a better estimate use BETAN)</td>
</tr>
<tr>
<td>DELV</td>
<td>flow-turning increment (recommended value, 0.1), $\Delta \nu$, deg</td>
</tr>
<tr>
<td>GAM</td>
<td>specific-heat ratio, $\gamma$</td>
</tr>
<tr>
<td>NTURBU</td>
<td>integer number of station on upper surface, if any, at which user wishes turbulent boundary layer to begin (If NTURBU is set equal to zero, the program will begin laminar boundary-layer calculations at the rotor inlet. Value of CTHETU must be specified; see section Instructions for Preparing Input. NTURBU cannot be set equal to 1 because initial values of displacement and momentum thicknesses are not available to program.)</td>
</tr>
<tr>
<td>NTURBL</td>
<td>same as NTURBU except for lower surface</td>
</tr>
</tbody>
</table>
TABLE I. - INPUT FORM
[Numbers in corners are card column numbers.]

<table>
<thead>
<tr>
<th>1</th>
<th>6</th>
<th>9</th>
<th>14</th>
<th>17</th>
<th>22</th>
<th>25</th>
<th>30</th>
<th>33</th>
<th>38</th>
<th>41</th>
<th>46</th>
<th>49</th>
<th>54</th>
<th>57</th>
<th>62</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETAN</td>
<td>VIN</td>
<td>VLOW</td>
<td>VUP</td>
<td>VOUT</td>
<td>BETAT</td>
<td>DELV</td>
<td>GAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>34</td>
<td>67</td>
<td>9</td>
<td>11</td>
<td>19</td>
<td>21</td>
<td>29</td>
<td>31</td>
<td>39</td>
<td>41</td>
<td>45</td>
<td>49</td>
<td>51</td>
<td>59</td>
<td>61</td>
</tr>
<tr>
<td>NUP</td>
<td>NVP</td>
<td>R</td>
<td>PTZ</td>
<td>TTZ</td>
<td>XMAX</td>
<td>TE</td>
<td>CTHETU</td>
<td>CTHETL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>56</td>
<td>10111516212526303132</td>
<td>KPRE</td>
<td>KGRAD</td>
<td>KSDE</td>
<td>KLAM</td>
<td>KMAIN</td>
<td>KPROF</td>
<td>KEM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
NVP  integer number of points desired in velocity profile of the boundary layer at each station (must have at least 1)

R  gas constant, $J/(kg)(K)$; $(ft)(lbf)/(slug)(^{0^\circ}R)$

PTZ  inlet relative total pressure, $N/m^2$; $lbf/ft^2$

TTZ  inlet relative total temperature, $K$; $^{0^\circ}R$

XMAX  blade chord, m; ft

TE  thickness of trailing edge (used only for aftermixing calculations), m; ft

CTHETU  real variable indicating ratio of momentum thickness after reattachment to momentum thickness at laminar separation for upper surface

CTHETL  same as CTHETU except for lower surface

KPRE  integer (0 or 1) indicating whether printing of output from PRECAL is desired:
   Output suppressed ......................................................... 0
   Output printed ............................................................ 1

KGRAD  integer (0 or 1, see KPRE) indicating whether printing of surface gradients of velocity and Mach number is desired

KSDE  integer (0 or 1, see KPRE) indicating whether printing of solutions of laminar and turbulent differential equations is desired

KLAM  integer (0 or 1, see KPRE) indicating whether printing of laminar calculations for location of instability and transition is desired

KMAIN  integer (0 or 1, see KPRE) indicating whether printing of principal calculated boundary-layer parameters is desired

KPROF  integer (0 or 1, see KPRE) indicating whether printing of velocity profiles is desired

KEM  integer (0 or 1) indicating which of two allowable sets of units are used in input:
   U.S. customary (pounds force, slugs, feet, seconds, degrees Rankine, and foot-pounds) ........................................ 0
   International System (newtons, kilograms, meters, seconds, kelvin, and joules) .............................................. 1
Instructions for Preparing Input

Laminar separation and reattachment. - If NTURBU and NTURBL are set equal to zero, the program will begin laminar-boundary-layer calculations at the rotor inlet. Because of the adverse pressure gradients that exist on the upper and lower surfaces the program may predict separation on these surfaces. The program is set up so that reattachment of the flow is allowed to occur, and the calculations proceed for a turbulent boundary layer, if this is desired. Values for CTHETU and CTHETL (different from zero), which are the ratios of the momentum thickness after reattachment to the momentum thickness at laminar separation, must be specified as input. The values of CTHETU and CTHETL can be set equal to 1, which is essentially equivalent to assuming that transition to turbulent flow occurs at the point of imminent laminar separation. If CTHETU and CTHETL are set equal to zero, reattachment will not occur.

Output. - Usually KPRE, KGRAD, KSDE, KLAM, and KPROF are set equal to zero. They do not give the main output of the boundary-layer section. However, if this additional output is desired, these quantities are set equal to 1. A description of this output is given in appendix B.

DESCRIPTION OF MAIN OUTPUT

An example of the output from the program is shown in table II. The output is given in U.S. customary units, and each section has been numbered to correspond to the following description:

(1) Output 1 of the program is a listing of the input data used for the rotor design plus a listing of miscellaneous parameters including
   (a) Inlet, outlet, and surface critical velocity ratios M* and Mach numbers M.
   (b) Dimensionless blade spacing G*, chord C*, and solidity SIGMA.
For each surface (lower and upper) items (2) to (5) are given for each iteration (if more than one iteration is required).

(2) Output 2 gives the input needed by the boundary-layer program. The first two lines of this output are a listing of the input data except for NST (which is the total number of station points) and UPMACH (which is the inlet relative Mach number). The remaining output is the surface coordinates and velocity distribution:

XOM  dimensionless X-coordinate, X/C*
YOM  dimensionless Y-coordinate, Y/C*
VVCR critical velocity ratio
TWAL wall temperature, K; °R (wall temperature assumed equal to total temperature TTZ)
### Table II. Example of Program Output

#### DESIGN PARAMETERS

- $\text{BETA}(\text{IN}) = 70.0000 \text{ DEG}$
- $\text{V(IN)} = 39.1200 \text{ DEG}$
- $\text{V(OUT)} = 49.1200 \text{ DEG}$
- $\text{V(IN)} = 39.1200 \text{ DEG}$
- $\text{BETA}(\text{OUT}) = 49.1200 \text{ DEG}$

#### DELTA

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta f$</td>
<td>0.1000</td>
</tr>
<tr>
<td>$\text{JCG}_{\text{WIL3Y1}}$</td>
<td>1.1233</td>
</tr>
<tr>
<td>$\text{H}_{\text{IN11}}$</td>
<td>1.6257</td>
</tr>
<tr>
<td>$\text{H}_{\text{IN12}}$</td>
<td>2.4999</td>
</tr>
<tr>
<td>$\Delta \text{SUMA}$</td>
<td>1.4000</td>
</tr>
</tbody>
</table>

#### MISCELLANEOUS PARAMETERS

- $\text{M}^*(\text{IN}) = 1.0257$
- $\text{M}^*(\text{OUT}) = 2.4999$
- $\text{M}^*(\text{OUT}) = 1.0257$
- $\text{M}^*(\text{IN}) = 1.02568$
- $\text{M}^*(\text{OUT}) = 1.7448$
- $\text{M}^*(\text{UP}) = 2.4672$
- $\text{M}^*(\text{UP}) = 1.9562$

#### **UPPER SURFACE**

#### BOUNDARY LAYER - INPUT

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{VST}$</td>
<td>64</td>
</tr>
<tr>
<td>$\text{VVP}$</td>
<td>5</td>
</tr>
<tr>
<td>$\text{GAM}$</td>
<td>1.4003</td>
</tr>
<tr>
<td>$\text{H}$</td>
<td>1714.70</td>
</tr>
<tr>
<td>$\text{PI1}$</td>
<td>676.00</td>
</tr>
<tr>
<td>$\text{PI2}$</td>
<td>1313.33</td>
</tr>
<tr>
<td>$\text{UPMAX}$</td>
<td>2.4999</td>
</tr>
<tr>
<td>$\text{KMAX}$</td>
<td>0.0418</td>
</tr>
<tr>
<td>$\text{VTURM}$</td>
<td>3</td>
</tr>
<tr>
<td>$\text{SURFT}$</td>
<td>1.000</td>
</tr>
<tr>
<td>$\text{ENST}$</td>
<td>0.00045</td>
</tr>
</tbody>
</table>

#### VARIABLES

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{KPRE}$</td>
<td>0</td>
</tr>
<tr>
<td>$\text{KSAD}$</td>
<td>0</td>
</tr>
<tr>
<td>$\text{KSUE}$</td>
<td>0</td>
</tr>
<tr>
<td>$\text{KLAM}$</td>
<td>1</td>
</tr>
<tr>
<td>$\text{KMAIN}$</td>
<td>1</td>
</tr>
<tr>
<td>$\text{KPRDF}$</td>
<td>1</td>
</tr>
<tr>
<td>$\text{KEW}$</td>
<td>0</td>
</tr>
</tbody>
</table>

#### PARAMETERS

- $\text{KDH}$
- $\text{YDM}$
- $\text{VVC}$
- $\text{THAL}$

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta \text{KDH}$</td>
<td>0.0562</td>
</tr>
<tr>
<td>$\Delta \text{YDM}$</td>
<td>0.0562</td>
</tr>
<tr>
<td>$\Delta \text{VVC}$</td>
<td>0.0562</td>
</tr>
<tr>
<td>$\Delta \text{THAL}$</td>
<td>0.0562</td>
</tr>
</tbody>
</table>

#### *** UPPER SURFACE ***

#### BOUNDARY LAYER - INPUT

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{VST}$</td>
<td>64</td>
</tr>
<tr>
<td>$\text{VVP}$</td>
<td>5</td>
</tr>
<tr>
<td>$\text{GAM}$</td>
<td>1.4003</td>
</tr>
<tr>
<td>$\text{H}$</td>
<td>1714.70</td>
</tr>
<tr>
<td>$\text{PI1}$</td>
<td>676.00</td>
</tr>
<tr>
<td>$\text{PI2}$</td>
<td>1313.33</td>
</tr>
<tr>
<td>$\text{UPMAX}$</td>
<td>2.4999</td>
</tr>
<tr>
<td>$\text{KMAX}$</td>
<td>0.0418</td>
</tr>
<tr>
<td>$\text{VTURM}$</td>
<td>3</td>
</tr>
<tr>
<td>$\text{SURFT}$</td>
<td>1.000</td>
</tr>
<tr>
<td>$\text{ENST}$</td>
<td>0.00045</td>
</tr>
</tbody>
</table>

#### VARIABLES

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{KPRE}$</td>
<td>0</td>
</tr>
<tr>
<td>$\text{KSAD}$</td>
<td>0</td>
</tr>
<tr>
<td>$\text{KSUE}$</td>
<td>0</td>
</tr>
<tr>
<td>$\text{KLAM}$</td>
<td>1</td>
</tr>
<tr>
<td>$\text{KMAIN}$</td>
<td>1</td>
</tr>
<tr>
<td>$\text{KPRDF}$</td>
<td>1</td>
</tr>
<tr>
<td>$\text{KEW}$</td>
<td>0</td>
</tr>
</tbody>
</table>

#### PARAMETERS

- $\text{KDH}$
- $\text{YDM}$
- $\text{VVC}$
- $\text{THAL}$

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta \text{KDH}$</td>
<td>0.0562</td>
</tr>
<tr>
<td>$\Delta \text{YDM}$</td>
<td>0.0562</td>
</tr>
<tr>
<td>$\Delta \text{VVC}$</td>
<td>0.0562</td>
</tr>
<tr>
<td>$\Delta \text{THAL}$</td>
<td>0.0562</td>
</tr>
</tbody>
</table>
### Table II - Continued. Example of Program Output

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.28774</td>
<td>0.13117</td>
<td>1.85316</td>
<td>1.91330</td>
<td>0.29227</td>
<td>0.12362</td>
</tr>
</tbody>
</table>

### Preliminary Calculations

<table>
<thead>
<tr>
<th>PSZ</th>
<th>TSZ</th>
<th>US</th>
<th>ATZ</th>
<th>QHSZ</th>
<th>RH12</th>
<th>MUS2</th>
<th>MUTZ</th>
<th>RUSZ</th>
<th>YUTZ</th>
<th>CP</th>
<th>PR</th>
<th>TC</th>
<th>ARCL</th>
</tr>
</thead>
<tbody>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39.6063</td>
<td>3517.70855</td>
<td>2.499452</td>
<td>0.058541</td>
<td>1.325694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Principal Boundary Layer Information

- Instability does not occur.
- Transition occurs at Station 3.
- Separation does not occur.
- Laminar boundary layer - Stations 1 to 2.
- Turbulent boundary layer - Stations 3 to 6.

### Dimensional Plane Coordinates

<table>
<thead>
<tr>
<th></th>
<th>UNCORRECTED</th>
<th>CORRECTED</th>
<th>TRANSLATED</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATION</td>
<td>X</td>
<td>Y</td>
<td>VM</td>
</tr>
<tr>
<td>1</td>
<td>-0.20735</td>
<td>-0.11835</td>
<td>-0.01845</td>
</tr>
<tr>
<td>2</td>
<td>-0.20945</td>
<td>-0.11945</td>
<td>-0.01955</td>
</tr>
<tr>
<td>3</td>
<td>-0.21155</td>
<td>-0.12055</td>
<td>-0.02065</td>
</tr>
<tr>
<td>4</td>
<td>-0.21365</td>
<td>-0.12165</td>
<td>-0.02175</td>
</tr>
<tr>
<td>5</td>
<td>-0.21575</td>
<td>-0.12275</td>
<td>-0.02285</td>
</tr>
<tr>
<td>6</td>
<td>-0.21785</td>
<td>-0.12385</td>
<td>-0.02395</td>
</tr>
<tr>
<td>7</td>
<td>-0.21995</td>
<td>-0.12495</td>
<td>-0.02505</td>
</tr>
<tr>
<td>8</td>
<td>-0.22205</td>
<td>-0.12605</td>
<td>-0.02615</td>
</tr>
<tr>
<td>9</td>
<td>-0.22415</td>
<td>-0.12715</td>
<td>-0.02725</td>
</tr>
<tr>
<td>10</td>
<td>-0.22625</td>
<td>-0.12825</td>
<td>-0.02835</td>
</tr>
<tr>
<td>11</td>
<td>-0.22835</td>
<td>-0.12935</td>
<td>-0.02945</td>
</tr>
<tr>
<td>12</td>
<td>-0.23045</td>
<td>-0.13045</td>
<td>-0.03055</td>
</tr>
</tbody>
</table>

### Angles of Rotation (Deg)

<table>
<thead>
<tr>
<th></th>
<th>YTRAY</th>
<th>70.00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-0.03351</td>
<td>70.00000</td>
</tr>
<tr>
<td>1</td>
<td>-0.02959</td>
<td>70.00000</td>
</tr>
<tr>
<td>2</td>
<td>-0.02560</td>
<td>70.00000</td>
</tr>
<tr>
<td>3</td>
<td>-0.02167</td>
<td>70.00000</td>
</tr>
<tr>
<td>4</td>
<td>-0.01779</td>
<td>70.00000</td>
</tr>
<tr>
<td>5</td>
<td>-0.01390</td>
<td>70.00000</td>
</tr>
<tr>
<td>6</td>
<td>-0.00982</td>
<td>70.00000</td>
</tr>
<tr>
<td>7</td>
<td>-0.00575</td>
<td>70.00000</td>
</tr>
<tr>
<td>8</td>
<td>-0.00168</td>
<td>70.00000</td>
</tr>
<tr>
<td>9</td>
<td>0.00000</td>
<td>70.00000</td>
</tr>
<tr>
<td>10</td>
<td>0.00327</td>
<td>70.00000</td>
</tr>
<tr>
<td>11</td>
<td>0.00650</td>
<td>70.00000</td>
</tr>
<tr>
<td>12</td>
<td>0.00975</td>
<td>70.00000</td>
</tr>
</tbody>
</table>
### TABLE II. - Continued. EXAMPLE OF PROGRAM OUTPUT

<table>
<thead>
<tr>
<th>STATION</th>
<th>X</th>
<th>S</th>
<th>DELSR</th>
<th>THET</th>
<th>DELTA</th>
<th>TAFN</th>
<th>TAFNL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0.02703</td>
<td>0.00258</td>
<td>0.00049</td>
<td>0.000064</td>
<td>0.000826</td>
<td>7.4563</td>
<td>7.4595</td>
</tr>
<tr>
<td>2</td>
<td>0.014958</td>
<td>0.00416</td>
<td>0.000386</td>
<td>0.000091</td>
<td>0.002334</td>
<td>4.2467</td>
<td>4.5000</td>
</tr>
<tr>
<td>3</td>
<td>0.015845</td>
<td>0.00674</td>
<td>0.000433</td>
<td>0.000099</td>
<td>0.002331</td>
<td>4.3709</td>
<td>4.4561</td>
</tr>
<tr>
<td>4</td>
<td>0.018113</td>
<td>0.00674</td>
<td>0.000433</td>
<td>0.000099</td>
<td>0.002331</td>
<td>4.3709</td>
<td>4.4561</td>
</tr>
<tr>
<td>5</td>
<td>0.017841</td>
<td>0.00633</td>
<td>0.000374</td>
<td>0.000106</td>
<td>0.002397</td>
<td>4.4594</td>
<td>4.5936</td>
</tr>
<tr>
<td>6</td>
<td>0.015845</td>
<td>0.01129</td>
<td>0.000013</td>
<td>0.002949</td>
<td>0.004081</td>
<td>1.5171</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.016096</td>
<td>0.01356</td>
<td>0.000055</td>
<td>0.000120</td>
<td>0.002660</td>
<td>4.5939</td>
<td>1.5309</td>
</tr>
<tr>
<td>8</td>
<td>0.015325</td>
<td>0.01507</td>
<td>0.000057</td>
<td>0.002719</td>
<td>0.005567</td>
<td>1.5367</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.015451</td>
<td>0.01866</td>
<td>0.000337</td>
<td>0.002844</td>
<td>0.005556</td>
<td>1.5426</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.013779</td>
<td>0.02033</td>
<td>0.000336</td>
<td>0.002733</td>
<td>0.005991</td>
<td>1.5442</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.013005</td>
<td>0.02278</td>
<td>0.000346</td>
<td>0.002402</td>
<td>0.003046</td>
<td>1.5266</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0.012707</td>
<td>0.02252</td>
<td>0.000364</td>
<td>0.001455</td>
<td>0.003332</td>
<td>4.5825</td>
<td>1.5041</td>
</tr>
</tbody>
</table>

### STATION CF TAUH RTH DITOY NUS$ HTRAN LCK

| J     | 0.00608 | 5.07681 | 3.0 | -261125.21 | 3.38 | -12776.7206 | 2.021 |
| 2     | 0.00708 | 1.32344 | 4.3 | 74653.36 | 2.69 | -584.0582 | 0.68 |
| 3     | 0.00808 | 1.19193 | 4.7 | -70279.32 | 3.41 | -343.8152 | 0.038 |
| 4     | 0.00908 | 1.10244 | 5.0 | 66494.51 | 4.46 | -318.0020 | 0.56 |
| 5     | 0.01008 | 1.04938 | 5.3 | -61622.65 | 5.28 | -301.4064 | 0.468 |
| 6     | 0.01108 | 1.00701 | 5.7 | -59366.07 | 6.11 | -290.4732 | 0.48 |
| 7     | 0.01208 | 0.98138 | 6.0 | -57893.39 | 6.44 | -282.9400 | 0.58 |
| 8     | 0.01308 | 0.96238 | 6.3 | -56737.07 | 7.78 | -277.1603 | 0.56 |
| 9     | 0.01408 | 0.94839 | 6.6 | -55909.82 | 9.52 | -273.5528 | 0.68 |
| 10    | 0.01508 | 0.93691 | 6.7 | -55081.50 | 9.76 | -274.2856 | 0.68 |
| 11    | 0.01608 | 0.92467 | 6.8 | -54262.25 | 7.77 | -278.8043 | 0.546 |

### BOUNDARY LAYER - INPUT

<table>
<thead>
<tr>
<th>MST</th>
<th>WVP</th>
<th>GAM</th>
<th>R</th>
<th>PTZ</th>
<th>FIT</th>
<th>UPMACH</th>
<th>KM</th>
<th>TURB</th>
<th>CHETA</th>
<th>FE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1.000</td>
<td>0.0043</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| J     | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 2     | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 3     | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 4     | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 5     | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 6     | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 7     | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 8     | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 9     | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 10    | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 11    | 0.04997 | 0.07642 | 1.8356 | 1913.30 |
| 12    | 0.04997 | 0.07642 | 1.8356 | 1913.30 |

### TABLE III. - CONTINUED

<table>
<thead>
<tr>
<th>KPRE</th>
<th>KJAD</th>
<th>KSDE</th>
<th>KLM</th>
<th>KMAM</th>
<th>KPMAP</th>
<th>KEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

### KEY

<table>
<thead>
<tr>
<th>XDM</th>
<th>YOM</th>
<th>VCR</th>
<th>TWAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
<td>1913.30</td>
</tr>
<tr>
<td>2</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>3</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>4</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>5</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>6</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>7</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>8</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>9</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>10</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>11</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
<tr>
<td>12</td>
<td>0.04997</td>
<td>0.07642</td>
<td>1.8356</td>
</tr>
</tbody>
</table>

** *** LOWER SURFACE *** **
<table>
<thead>
<tr>
<th>TABLE II. - Continued. EXAMPLE OF PROGRAM OUTPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.38025</td>
</tr>
<tr>
<td>0.40005</td>
</tr>
<tr>
<td>0.41272</td>
</tr>
<tr>
<td>0.42723</td>
</tr>
<tr>
<td>0.43983</td>
</tr>
<tr>
<td>0.44037</td>
</tr>
<tr>
<td>0.44282</td>
</tr>
<tr>
<td>0.44743</td>
</tr>
<tr>
<td>0.48793</td>
</tr>
<tr>
<td>0.50163</td>
</tr>
</tbody>
</table>

**PRELIMINARY CALCULATIONS**

<table>
<thead>
<tr>
<th>PSZ</th>
<th>TSZ</th>
<th>UZ</th>
<th>ASZ</th>
<th>ATZ</th>
<th>RSZ</th>
<th>RMS</th>
<th>USZ</th>
<th>UMT</th>
<th>NUS</th>
<th>NUT</th>
<th>CP</th>
<th>PR</th>
<th>TC</th>
<th>ACL</th>
</tr>
</thead>
<tbody>
<tr>
<td>34.62003</td>
<td>330.4114</td>
<td>3071.7965</td>
<td>1928.8039</td>
<td>2143.1355</td>
<td>0.271709E-04</td>
<td>0.2662752E-03</td>
<td>0.3372209E-06</td>
<td>0.9143356E-06</td>
<td>0.197111E-01</td>
<td>6001.63498</td>
<td>0.672322</td>
<td>0.482751E-02</td>
<td>0.05925</td>
<td></td>
</tr>
</tbody>
</table>

**STATION**

<table>
<thead>
<tr>
<th>STATION</th>
<th>PRESS</th>
<th>UE</th>
<th>WE</th>
<th>PORT</th>
<th>VOCH</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>19.62003</td>
<td>3371.79855</td>
<td>2.499852</td>
<td>0.058541</td>
<td>1.325694</td>
</tr>
<tr>
<td>2</td>
<td>43.33229</td>
<td>3544.40982</td>
<td>2.657325</td>
<td>0.067548</td>
<td>1.317194</td>
</tr>
<tr>
<td>3</td>
<td>45.18726</td>
<td>3516.68137</td>
<td>2.419483</td>
<td>0.065766</td>
<td>1.797551</td>
</tr>
<tr>
<td>4</td>
<td>48.19350</td>
<td>3468.58299</td>
<td>2.374237</td>
<td>0.071208</td>
<td>1.783359</td>
</tr>
<tr>
<td>5</td>
<td>51.35440</td>
<td>3460.12849</td>
<td>2.335927</td>
<td>0.075878</td>
<td>1.768815</td>
</tr>
<tr>
<td>6</td>
<td>54.17762</td>
<td>3431.29149</td>
<td>2.297311</td>
<td>0.080789</td>
<td>1.753875</td>
</tr>
<tr>
<td>7</td>
<td>58.15675</td>
<td>3402.09058</td>
<td>2.253987</td>
<td>0.085944</td>
<td>1.738949</td>
</tr>
<tr>
<td>8</td>
<td>51.48782</td>
<td>3372.50253</td>
<td>2.214900</td>
<td>0.091345</td>
<td>1.723829</td>
</tr>
<tr>
<td>9</td>
<td>65.60099</td>
<td>3342.52209</td>
<td>2.176470</td>
<td>0.097029</td>
<td>1.708501</td>
</tr>
<tr>
<td>10</td>
<td>69.49649</td>
<td>3312.16563</td>
<td>2.138441</td>
<td>0.102975</td>
<td>1.692975</td>
</tr>
<tr>
<td>11</td>
<td>73.92069</td>
<td>3281.39953</td>
<td>2.100862</td>
<td>0.109206</td>
<td>1.677239</td>
</tr>
<tr>
<td>12</td>
<td>78.32492</td>
<td>3250.15289</td>
<td>2.063711</td>
<td>0.115727</td>
<td>1.661287</td>
</tr>
</tbody>
</table>

**PRINCIPAL BOUNDARY LAYER INFORMATION**

**INSTABILITY DOES NOT OCCUR**

**TRANSITION OCCURS AT STATION 3**

**SEPARATION DOES NOT OCCUR**

**LAMINAR BOUNDARY LAYER - STATIONS 1 TO 2**

**TURBULENT BOUNDARY LAYER - STATIONS 3 TO 50**

**DIMENSIONED BLADE COORDINATES**

<table>
<thead>
<tr>
<th>STATION</th>
<th>UX</th>
<th>HY</th>
<th>VX</th>
<th>TY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.0263</td>
<td>-0.00320</td>
<td>-0.00320</td>
<td>-2.2344</td>
</tr>
<tr>
<td>2</td>
<td>-0.02033</td>
<td>-0.00121</td>
<td>-0.00121</td>
<td>-2.01835</td>
</tr>
<tr>
<td>3</td>
<td>-0.01999</td>
<td>-0.00111</td>
<td>-0.00111</td>
<td>-2.15544</td>
</tr>
<tr>
<td>4</td>
<td>-0.01953</td>
<td>-0.00015</td>
<td>-0.00015</td>
<td>-2.01459</td>
</tr>
<tr>
<td>5</td>
<td>-0.01943</td>
<td>-0.00076</td>
<td>-0.00076</td>
<td>-2.01456</td>
</tr>
<tr>
<td>6</td>
<td>-0.01847</td>
<td>-0.00163</td>
<td>-0.00163</td>
<td>-2.01728</td>
</tr>
<tr>
<td>7</td>
<td>-0.01847</td>
<td>-0.00215</td>
<td>-0.00215</td>
<td>-2.01728</td>
</tr>
<tr>
<td>8</td>
<td>-0.01795</td>
<td>-0.00334</td>
<td>-0.00334</td>
<td>-2.01116</td>
</tr>
<tr>
<td>9</td>
<td>-0.01756</td>
<td>-0.00479</td>
<td>-0.00479</td>
<td>-2.01040</td>
</tr>
<tr>
<td>10</td>
<td>-0.01317</td>
<td>-0.00740</td>
<td>-0.00740</td>
<td>-2.00569</td>
</tr>
<tr>
<td>11</td>
<td>-0.01610</td>
<td>-0.00953</td>
<td>-0.00953</td>
<td>-2.00400</td>
</tr>
<tr>
<td>12</td>
<td>-0.01400</td>
<td>-0.01064</td>
<td>-0.01064</td>
<td>-2.00835</td>
</tr>
</tbody>
</table>

**ANGLES OF ROTATION (DEG)**

<table>
<thead>
<tr>
<th>STATION</th>
<th>UX</th>
<th>HY</th>
<th>VX</th>
<th>TY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>2</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>3</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>4</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>5</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
</tbody>
</table>

12
<table>
<thead>
<tr>
<th>STATION</th>
<th>X</th>
<th>Y</th>
<th>DELX</th>
<th>THEX</th>
<th>DELTA</th>
<th>FORM</th>
<th>FORM1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.020370</td>
<td>0.00145</td>
<td>0.000329</td>
<td>0.000045</td>
<td>3.00573</td>
<td>7.7629</td>
<td>1.2613</td>
</tr>
<tr>
<td>2</td>
<td>-0.018427</td>
<td>0.002136</td>
<td>0.000254</td>
<td>0.000063</td>
<td>3.001355</td>
<td>7.0499</td>
<td>1.4600</td>
</tr>
<tr>
<td>3</td>
<td>-0.041928</td>
<td>0.003137</td>
<td>0.000064</td>
<td>0.000057</td>
<td>3.001435</td>
<td>4.1361</td>
<td>1.6000</td>
</tr>
<tr>
<td>4</td>
<td>-0.018132</td>
<td>0.004621</td>
<td>0.000098</td>
<td>0.000071</td>
<td>3.011384</td>
<td>4.1941</td>
<td>1.5909</td>
</tr>
<tr>
<td>5</td>
<td>-0.013735</td>
<td>0.005223</td>
<td>0.000069</td>
<td>0.000078</td>
<td>3.013197</td>
<td>4.2796</td>
<td>1.6121</td>
</tr>
<tr>
<td>6</td>
<td>-0.013842</td>
<td>0.006137</td>
<td>0.000085</td>
<td>0.000078</td>
<td>3.013329</td>
<td>4.2762</td>
<td>1.6646</td>
</tr>
<tr>
<td>7</td>
<td>-0.017450</td>
<td>0.007194</td>
<td>0.000094</td>
<td>0.000094</td>
<td>3.013119</td>
<td>4.2742</td>
<td>1.7104</td>
</tr>
<tr>
<td>8</td>
<td>-0.017159</td>
<td>0.007557</td>
<td>0.000094</td>
<td>0.000134</td>
<td>3.013113</td>
<td>4.2743</td>
<td>1.7512</td>
</tr>
<tr>
<td>9</td>
<td>-0.001711</td>
<td>0.008411</td>
<td>0.000088</td>
<td>0.000134</td>
<td>4.2263</td>
<td>1.7890</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>-0.005083</td>
<td>0.004956</td>
<td>0.000089</td>
<td>0.000112</td>
<td>4.2094</td>
<td>1.8454</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>-0.001639</td>
<td>0.010216</td>
<td>0.000099</td>
<td>0.000134</td>
<td>4.1921</td>
<td>1.8609</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>-0.004368</td>
<td>0.012006</td>
<td>0.000099</td>
<td>0.000134</td>
<td>4.1921</td>
<td>1.8609</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STATION</th>
<th>CF</th>
<th>TANW</th>
<th>R14</th>
<th>DPDW</th>
<th>NUSS</th>
<th>HTRAN</th>
<th>CRY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>2.021</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3.00369</td>
<td>6.71206</td>
<td>2.3</td>
<td>3.76773</td>
<td>3.33</td>
<td>-1.923.074</td>
<td>1.677</td>
</tr>
<tr>
<td>3</td>
<td>3.00599</td>
<td>1.34670</td>
<td>3.3</td>
<td>1.989.48</td>
<td>1.31</td>
<td>-55.0.057</td>
<td>2.611</td>
</tr>
<tr>
<td>4</td>
<td>3.00746</td>
<td>1.43355</td>
<td>3.2</td>
<td>-1.970.20</td>
<td>2.14</td>
<td>-4.01.059</td>
<td>2.643</td>
</tr>
<tr>
<td>5</td>
<td>3.00684</td>
<td>1.20903</td>
<td>4.2</td>
<td>-3.424.09</td>
<td>2.54</td>
<td>-5.36.160</td>
<td>3.659</td>
</tr>
<tr>
<td>6</td>
<td>3.00599</td>
<td>1.26582</td>
<td>4.7</td>
<td>-8.392.74</td>
<td>2.93</td>
<td>-3.34.194</td>
<td>2.697</td>
</tr>
<tr>
<td>7</td>
<td>3.00546</td>
<td>1.13436</td>
<td>5.1</td>
<td>-6.355.97</td>
<td>3.27</td>
<td>-31.1.667</td>
<td>2.618</td>
</tr>
<tr>
<td>8</td>
<td>3.00508</td>
<td>1.07856</td>
<td>5.7</td>
<td>-6.057.27</td>
<td>3.38</td>
<td>-2.93.747</td>
<td>2.640</td>
</tr>
<tr>
<td>9</td>
<td>3.00474</td>
<td>1.03590</td>
<td>6.2</td>
<td>-5.955.88</td>
<td>3.05</td>
<td>-2.78.877</td>
<td>2.702</td>
</tr>
<tr>
<td>10</td>
<td>3.00446</td>
<td>0.95946</td>
<td>6.6</td>
<td>-5.831.07</td>
<td>4.13</td>
<td>-2.65.740</td>
<td>2.714</td>
</tr>
<tr>
<td>11</td>
<td>3.00415</td>
<td>0.95733</td>
<td>7.4</td>
<td>-5.816.48</td>
<td>4.45</td>
<td>-2.53.690</td>
<td>2.716</td>
</tr>
<tr>
<td>12</td>
<td>3.00395</td>
<td>0.97221</td>
<td>8.0</td>
<td>-4.973.16</td>
<td>4.55</td>
<td>-2.42.059</td>
<td>2.716</td>
</tr>
</tbody>
</table>

**AFTERMIXING PROPERTIES**

**AFTER WITH NO BOUNDARY LAYER CORRECTION**

| XMFS1 | 2.499999 | 0.0219375 | 0.00364 | 1.5983 | 0.625 | 1.207 | 0.641 |
| ALPHAI | 61.133 | ALPHAI | 66.116 | PTD/PTO | 0.282 | PTO/PLO | 14.888 | T/P/TO | 0.666 | V/V/C/V | 2.844 | EBA/R | 0.378 | ETA-Y | 0.6.27 |

**AFTER WITH BOUNDARY LAYER CORRECTION**

| XMFS1 | 2.499999 | 0.0219375 | 0.00364 | 1.5983 | 0.625 | 1.207 | 0.641 |
| ALPHAI | 61.133 | ALPHAI | 67.221 | PTD/PTO | 0.454 | PTO/PLO | 13.513 | T/P/TO | 0.598 | V/V/C/V | 1.533 | EBA/R | 0.23089 | ETA-Y | 0.7091 |

**SUPERSATURATED SOLUTION**

| XMFS1 | 2.499999 | 0.0219375 | 0.00364 | 1.5983 | 0.625 | 1.207 | 0.641 |
| ALPHAI | 61.133 | ALPHAI | 52.603 | PTD/PTO | 0.4867 | PTO/PLO | 31.532 | T/P/TO | 0.4585 | V/V/C/V | 1.802 | EBA/R | 0.1364 | ETA-Y | 0.8.34 |

**ITERRATIVELY COMPLETED**

**DATA:**

T/P/TO | 0.59433 | 0.59435 | 0.60 (TOTAL) | 0.5943 |

**EFL:**

| T/P/TO | 0.59435 |
| SIGMA/ | 0.5943 |
| SIGMA/ | 0.5943 |

**END**

0.000002
(3) Output 3 lists the properties calculated for the upstream conditions:

- **PSZ**: upstream static pressure, N/m²; lbf/ft²
- **TSZ**: upstream static temperature, K; ⁰R
- **UZ**: upstream velocity, m/sec; ft/sec
- **ASZ (ATZ)**: speed of sound based on upstream static (total) temperature, m/sec; ft/sec
- **RHSZ (RHTZ)**: static (total) density based on upstream static (total) temperature, kg/m³; slug/ft³
- **MUSZ (MUTZ)**: dynamic viscosity based on upstream static (total) temperature, (N)(sec)/m²; (lbf)(sec)/ft²
- **NUSZ (NUTZ)**: kinematic viscosity based on upstream static (total) temperature, m²/sec; ft²/sec
- **CP**: specific heat at constant pressure, J/(kg)(K); (ft)(lbf)/(slug)(⁰R)
- **PR**: Prandtl number
- **TC**: thermal conductivity, J/(m)(sec)(K); (ft)(lbf)/(ft)(sec)(⁰R)
- **ARCL**: total distance along surface, m; ft

The next part of this output gives the variables describing the flow along the surface:

- **PRES**: static pressure, N/m²; lbf/ft²
- **UE**: free-stream velocity, m/sec; ft/sec
- **ME**: free-stream Mach number
- **POPTZ**: static-to-total pressure ratio
- **VOVCR**: critical velocity ratio

(4) Output 4 corresponds to KMAIN. It indicates the regions of laminar and turbulent boundary layers, and the stations at which instability, transition, and separation occur. It gives all the principal boundary-layer output parameters:

- **X**: X-coordinate, m; ft
- **S**: surface length, m; ft
- **DELSR**: displacement thickness, m; ft
- **THET**: momentum thickness, m; ft
DE’LTA boundary-layer thickness, m; ft
FORM compressible form factor
FORMI incompressible form factor

The next part of the output gives the skin-friction and heat-transfer parameters:

CF skin friction coefficient at wall
TAUW shear stress at wall, N/m^2; lbf/ft^2
RTH momentum-thickness Reynolds number
DTDY slope of temperature profile at wall, K/m; °R/ft
NUSS local Nusselt number
HTRAN heat transfer per unit area, J/(sec)(m^2); (ft)(lbf)/(sec)(ft^2)
CRN Reynolds analogy parameter

(5) Output 5 gives the ideal and corrected surface coordinates. Translated surface coordinates are also provided so that coordinates for a blade profile may be obtained.

(6) After the output for both surfaces is given, the final output, numbered 6, lists the aftermixing properties. Variable names ending in 0 refer to the upstream station, while those ending in 1 refer to the rotor exit (before mixing) and those ending in 2 refer to the mixed relative conditions downstream of the rotor. The output consists of rotor exit free-stream Mach number (XMFS1), rotor exit spacing (SPACING) in m (ft), trailing-edge thickness (TE) in m (ft), Mach number (XM2), axial Mach numbers (XM1 and XM2), critical velocity ratios (V/VCR1 and V/VCR2), flow angles measured from axial direction (ALPH1 and ALPH2) in degrees, total-to-total pressure ratio (PT2/PT0), total-to-static pressure ratio (PT0/P2), static-to-total temperature ratio (T2/TT0), rotor kinetic energy loss coefficient (EBAR2), and rotor efficiency (ETA-N). Also printed are some miscellaneous parameters for the corrected blade profile including the final outlet flow angle and solidity.

PROGRAM DESCRIPTION

The program SSROTR designs supersonic rotor blades corrected for boundary-layer displacement thickness. The program uses the input value of the outlet flow angle (which controls the outlet circular turning) for the initial design. If the corrected outlet spacing is not equal to the inlet spacing, the program estimates a new value of outlet flow angle and repeats the design. The iteration procedure is essentially a half-interval search. The program input consists essentially of the surface Mach number distribution
and total flow conditions. The primary output is the corrected blade profile and the boundary-layer parameters.

The program SSROTR is logically divided into two parts: (1) an ideal rotor design section, and (2) a section to calculate the boundary-layer parameters and aftermixing conditions. The ideal rotor design section consists principally of subroutines ROTORU and ROTORR. The boundary-layer section is controlled by subroutine BLAYR. The major subroutines called by BLAYR are PRECAL, LAMNAR, TURBLN, and AFMIX.

**SUBROUTINE ROTORU**

Subroutine ROTORU calculates the upper- and lower-surface transition arcs by the method of characteristics. The description of the method and the main dictionary of variables are given in reference 2. This routine is called only once for any given blade design. The program variables for ROTORU that are not in reference 2 are

- ETALOW: angle of tangent to lower surface, measured from axial direction
- ETAUP: angle of tangent to upper surface, measured from axial direction
- GAMMI: \( \gamma - 1 \)
- GAMPI: \( \gamma + 1 \)
- STML: lower-surface critical velocity ratio
- STMU: upper-surface critical velocity ratio

**SUBROUTINE ROTORR**

Subroutine ROTORR calculates the rotated ideal passage coordinates, as well as the inlet and outlet passage spacing based on the current value of outlet flow angle. This routine is called once for each iteration of outlet flow angle. The description of the method and the main dictionary of variables are given in reference 2. The variables for ROTORR that are not in reference 2 are

- ETACL: angle of flow along lower-surface circular arc
- ETACU: angle of flow along upper-surface circular arc
- ETALN: angle of flow along lower-surface inlet transition arc
- ETALO: angle of flow along lower-surface outlet transition arc
- ETASN: absolute value of inlet flow angle
ETASO  absolute value of outlet flow angle
ETAUN  angle of flow along upper-surface inlet transition arc
ETAOO  angle of flow along upper-surface outlet transition arc
SMLN   lower-surface critical velocity ratio for inlet
SMLO   lower-surface critical velocity ratio for outlet
SMUN   upper-surface critical velocity ratio for inlet
SMUO   upper-surface critical velocity ratio for outlet

SUBROUTINE BLAYR

Subroutine BLAYR provides calls for the calculation of the boundary-layer characteristics and the mixing losses. The major subroutines called by BLAYR are PRECAL, LAMNAR, TURBLN, and AFMIX. These, in turn, call several other subroutines. The methods used for calculating the boundary-layer characteristics and the program description are given in reference 4. The Lagrangian interpolation routine of reference 4 has been replaced by a linear interpolation.

Since a highly favorable pressure gradient exists for some portions of the blade surfaces, the range of the equations of reference 6 were extended by the method given in reference 6. These changes were made in LAMNAR and are for $SW$ (temperature function at the wall) = 0. These changes are noted in the program with comment cards. Two curve-fit ranges were also extended. $RCRIT$ and $DIFF$ were extended as follows: $RCRIT = 8.3163$ when $SHAPK$ is greater than 0.07; and $DIFF = 44000$ KBAR + 700 when $KBAR$ is greater than 0.03.

SUBROUTINE AFMIX

Subroutine AFMIX takes the boundary-layer parameters (displacement and momentum thicknesses) and the free-stream conditions at the rotor exit and calculates the aftermixing conditions by the method described in reference 5. In this loss model the flow sufficiently downstream of the blades is assumed to be mixed to uniform conditions.

The calling sequence for AFMIX is CALL AFMIX (ALPHI, DELS, DELP, THETS, THETP, TE, SP, XMFSI), where

ALPHI  rotor exit flow angle measured from axial direction
DELS   displacement thickness at rotor exit for suction (upper) surface
DELP   displacement thickness at rotor exit for pressure (lower) surface
THETS  momentum thickness at rotor exit for suction surface
THETP  momentum thickness at rotor exit for pressure surface
TE    trailing-edge thickness
SP    exit spacing
XMFSI free-stream Mach number at rotor exit

The subroutine description and the main dictionary of variables are given in reference 3.

PROGRAM LISTING

$IEFCTC SSROTR  DECK
C  COMBINED ROTOP-BOUNDARY LAYER PROGRAM
C
COMM/ CODE/KCDE, DELSRL, DELSRU
COMM/CTOBL/ALPH1, SPA, XMAX, TF, NTURB, NTURPL, CTHETU, CTHETL
COMM/LNK1/XOML(100), YOML(100), SMU(100), ETAML(100), NL, XOMU(100),
1 YOMU(100), SMU(100), ETAML(100), NU
COMM/LNK2/SMIN, SMOU, BETAT, CSTAP, SIGMAO, GSTAPO, GSTAPR, GAMMA,
* CONVER, RECONV
COMM/ROTR/V(4), BETAN, NKR(5), RR(4)
COMM/C1/DUM1(7), NTURB, DUM2(6), CTHET, DUM3(910)

DIMENSION X(100), Y(100)
1 BETAP = 0.
BETAM = 0.
C CALCULATE UNROTATED BLADE COORDINATES
CALL ROTORU
CALL INPUT1
CALB = 0.
C CALCULATE ROTATED BLADE COORDINATES
2 CALL ROTORR(CALB)
CALB = 1.
C NORMALIZE ROTATED BLADE COORDINATES TO CHORD LENGTH
DO 3 I=1,NL
X(I) = XOMU(I)/CSTAR
3 Y(I) = YCMU(I)/CSTAR
KODE = 0
NTURB = NTURBU
CTHTP = CTHETU
CALL INPUT2 (X, Y, SMU, ETAML, NU)
C CALCULATE UPPER SURFACE BOUNDARY LAYER
CALL PLAY?
C NORMALIZE ROTATED BLADE COORDINATES TO CHORD LENGTH
DO 4 I=1,NL
X(I) = XOML(I)/CSTAR

18
4 \( Y(I) = YCM(I)/CSTAP \)

\( \text{KODE} = 1 \)

\( \text{NTURBL} = \text{NTURBL} \)

\( \text{CTHET} = \text{CTHETL} \)

CALL LIPUT2 (X,Y,SML,ETAP,L)

C CALCULATE LOWER SURFACE BOUNDARY LAYER

CALL ELAYR

C CHECK FOR EQUAL SPACING AT INLET AND OUTLET

\( \text{DELTOT} = (\text{DELSRL} + \text{DELSRU}) \times \text{CSTAR} \)

\( \text{OUTSPA} = \text{GSTARO} + \text{DELTOT} \times \cos(\text{ETAP} \times \text{CONVR}) \)

\( \text{ELIMIT} = 0.001 \times \text{CSTAR} / \text{XMAX} \)

IF (ABS(\( \text{GSTARI} - \text{OUTSPA} \)) \( \leq \) ELIMIT) GO TO 5

IF (\( \text{OUTSPA} \geq \text{GSTARI} \)) \( \text{ETAP} = \text{ETAT} \)

IF (\( \text{OUTSPA} \lt \text{GSTARI} \)) \( \text{ETAM} = \text{ETAT} \)

\( \text{GSTARO} = \text{OUTSPA} \)

IF (\( \text{ETAP} \neq 0. \) AND \( \text{ETAM} \neq 0. \)) GO TO 6

C CALCULATE NEW OUTLET FLOW ANGLE

\( \text{BETAT} = -\text{ARCOS}(\cos(\text{ETAM}) \times \text{SMOUT} / \text{SMOUT} \times (1. + (\text{GAMMA} - 1.)/2.*\( \text{SMOUT} \times \text{SMOUT} )/(1. + (\text{GAMMA} - 1.)/2.*\text{SMOUT} \times \text{SMOUT} ))**((\text{GAMMA} + 1.)/(2.*\( \text{GAMMA} - 1.))) + \text{DELTOT} / \text{GSTARI} \times \text{P} \times \text{CONVR} \)

GO TO 2

6 \( \text{BETAT} = (\text{ETAP} + \text{ETAM}) / 2. \)

GO TO 2

5 WRITE (6, 100) \( \text{ETAT, GSTARI, OUTSPA} \)

100 FORMAT (15X, 19HITERATION COMPLETED/7X, 11PESTA (OUT) =, F10.4, 4H DEG
*, 10X, 8HG*(IN) =, F10.5, 10X, 22HG*(OUT) + DEL(TOTAL) =, F10.5)

CALB = 2.

CALL FOTORR (CALB)

GO TO 1

END

$IEFTC INPUT1 DECK

SUBROUTINE INPUT1

C BOUNDARY LAYER INPUT

COMMON/LTU/LDAT

COMMON/LINK2/UPMAC, SMOUT, ALP1K, CSTAP, SIGMAC, GSTARO, GSTARI, GAMMA,
* CONVR, RECON7

COMMON/CTOBL/ALPH1, SPA, XMAX, TE, NTURBU, NTURBL, CTHETU, CTHETL

COMMON/C1/GAM, R, PTZ, TTZ, UPMAC, NST, NVP, NTURB, KPVM, KEM, KSMTH,
1KSPNL, KLP, KATCH, CTHET, DLAM, TLAM, DTURPE, TTURB, KPPE, KGRAD, KSDF, KLM,
2KMAIN, KPROF, X(100), Y(100), PRPS(100), UP(100), ME(100), POPC2(100),
3VOVCR(100), TWAL(100), ETA(100)

READ (LDAT, 100) NTURBU, NTURBL, NVP, R, PTZ, TTZ, XMAX, TE, CTHETU, CTHETL

100 FORMAT (313, 1X, 7(F9.5, 1X))

READ (LDAT, 1010) KPPE, KGRAD, KSDF, KLM, KMAIN, KPROF, KEM

1010 FORMAT (7I5)

GAM = GAMMA

UPMACH = UPMAC

RETURN

END
$IBFTC ROTU  LIST,DECK
SUBROUTINE ROTUROU

C****CALCULATE THE UNROTATED PLANE COORDINATES

COMMON/LTU/LDAT
COMMON/FACTOR/PERM,GAMM1,GAMP1,GAM
COMMON/ROT1/VIN,VOUT,VLOW,VUP,BETAN,NFEP,KMAXN,KMAXO,JMAXN,JMAXO,
   1 RIN,ROUT,LOW,RUP
COMMON/ROT2/XLOW(400),YLOW(400),STML(400),ETALOW(400),KINDEX,
   1 XUP(400),YUP(400),STMU(400),ETAUP(400),INDEX
COMMON/LNK2/SMIN,SMOUT,ETAT,CSTAR,SIGMAO,GSTAR0,GSTAR1,GAMMA,
   * CONVER,RECONV

LOGICAL ANGLE
EXTERNAL FCFRS
DATA LDAT/5/
F(V,FN) = 2.*V - HALFPI*(PERM-1.) - 2.*(FN-1.)*DELV

1 READ (LDAT,11) BETAN,VIN,VLOW,VUP,VOUT,BETAT,DELV,GAM
11 FORMAT (8(F6.2,2X))

CC CONVERSION FACTORS AND CONSTANTS
GAMMA = GAM
CONVER = .174532925E-01
RECONV = 57.2957796
HALFPI = 3.14159265/2.

C ONE POINT WILL BE PRINTED FOR EVERY NPER POINTS CALCULATED
NPER = 10
IF (DELV .GE. 0.2) NPER = 1
GAMP1 = (GAM + 1.)/2.
GAMM1 = (GAM-1.)/2.
PERM = SQRT(GAMP1/GAMM1)
X0 = 1./PERM
X2 = 0.999999999
XINTL = (X0 + X2)/2.

ANGLE = .TFUE.
IF (VLOW .LE. AMIN1(VUP,VIN,VOUT)) GO TO 120
WRITE (6,119)
119 FORMAT (/31X,70HV(LOW) MUST BE LESS THAN OR EQUAL TO THE MINIMUM
   10F V(UP),V(IN),V(OUT))
ANGLE = .FALSE.

120 IF (VUP .GE. AMAX1(VIN,VOUT)) GO TO 118
WRITE (6,117)
117 FORMAT (/33X,66HV(UP) MUST BE GREATER THAN OR EQUAL TO THE MAXIMUM
1M OF V(IN),V(OUT))
ANGLE = .FALSE.

118 VUMAX = HALFPI*(PERM-1.)*RECONV
IF (VUP .LE. VUMAX) GC TO 116
WRITE (6,115) VUMAX
115 FORMAT (/41X,37HV(UP) MUST BE LESS THAN V(UP)(MAX) = .F9.4,0H DEG
1) ANGLE = .FALSE.

20
116 IF (.NOT. ANGLE) GO TO 1

WRITE (6,97)
97 FORMAT (1H1,57X,17HDESIGN PARAMETERS)

CC MISCELLANEOUS CALCULATIONS
DELV = DELV*CONVER
FN = 1.
V = VIN*CONVER
DO 4 I=1,2
FOPX = F(V,FN)
CALL FOOT (X0,X2,XINTL,FOPX,FOPRS,X1)
IF (I .EQ. 2) GO TO 4
XIN = X1
VOUT = VOUT*CONVER
4 CONTINUE

ROUT = X1

SSMIN = 1./RIN
SSMOUT = 1./ROUT

I = 1
16 SN = SORT(((1./GAM1)*SM*SMS)/(1.- (GAM1/GAM1)*SMS*SMS))
GO TO (17,18,19,20),I
17 SMIN = SM
SMS = SSMOUT
I = 2
GO TO 16
18 SMOUT = SM

DELV = DELV*CONV

CC PRINT ALL DESIGN PARAMETERS
WRITE (6,95) BETAN,VIN,VUP,VOUT,PRTAT
95 FORMAT (/2X,10HBETA (IN) =,F8.4,4H DEG,4X,7HV (IN) =,F8.4,4H DEG,6X,
1 7HV (UP) =,F9.4,4H DEG,7X,8HV (OUT) =,F8.4,4H DEG,4X,11HBETA (OUT) =
2,F9.4,4H DEG)
WRITE (6,94) DELV, VLOW, GAM
94 FORMAT (/20X,9HDELTA V =,F8.4,4H DEG,11X,8HV (LOW) =,F8.4,4H DEG,
1 11X,7HGAMMA =,F8.4)

CC CONVERT FROM DEGREES TO RADIANS
VIN = VIN*CONVER
VOUT = VOUT*CONVER
VUP = VUP*CONVER
VLOW = VLOW*CONVER
BETAN = BETAN*CONVER

CC CHOOSE LONGEST TRANSITION ARC OF LOWER SURFACE
VNL = VIN - VLOW
KMAXX = (VNL/DELV) + 0.5

VCL = VOUT - VLOW
KMAXO = (VOL/DELV) + 0.5
KMN = MAX0(KMAXN,KMAXO)
V = AMAX1(VIN,VOUT)
CC  CALCULATE  *(LOW)=BLOW,  M*(LOW)=SSMLGW,  M(LOW)=SYLOW  
   IF (VLOW .EQ. 0.0) GO TO 2  
   FN = KMN + 1  
   F0FX = F(V, FN)  
   CALL ROOT (X0, X2, XINTL, F0FX, FOFRS, PLOW)  
   GO TO 3  
2  BLOW = 1.0  
3  SSMLGW = 1./BIOW  
   SMS = SSMLGW  
   I = 3  
   GO TO 16  
19  SSMLGW = SM  

CC  SET INITIAL POINTS FOR LOWER ARC CALCULATIONS  
   KNDEx = KMN/NPER  
   JNDEx = KNDEx  
   STM(NDEx+1) = SSMLGW  
   XLOW(JNDEx+1) = 0.0  
   YLOW(JNDEx+1) = BLOW  
   ETALOW(JNDEx+1) = 0.  
   PHIKP1 = -(V-VLOW) + FLOAT(KMN)*DELV  
   UMKP1 = ARS IN(SORT(GAMP1*PLOW*BLOW - GAMM1))  
   TXLO = XLOW(JNDEx+1)  
   TYLO = YLOW(JNDEx+1)  

CC  CHOOSE LONGEST TRANSITION ARC OF UPPER SURFACE  
   VUT = VUP - VCUT  
   JMAXO = (VUT/DElV)*0.5  
   VUI = VUP - VIN  
   JMAXN = (VUI/DElV)*0.5  
   JMN = MAXO(JMAXO, JMAXN)  
   V = AM IN1(VCUT, VIN)  

CC  CALCULATE  *(UP)=PUP,  M*(UP)=SSMUP,  M(UP)=MUP  
   FN = -(JMN+1) + 2  
   F0FX = F(V, FN)  
   CALL ROOT (X0, X2, XINTL, F0FX, FOFRS, RUP)  
   SSMUP = 1./RU P  
   SMS = SSMUP  
   I = 4  
   GO TO 16  
20  SSMUP = SM  

CC  SET INITIAL POINTS FOR UPPER ARC CALCULATIONS  
   JNDEx = JMN/NPER  
   JNDEx = JNDEx  
   STM(JNDEx+1) = SSMUP  
   XUP(JNDEx+1) = 0.0  
   YUP(JNDEx+1) = RUP  
   ETAUP(JNDEx+1) = 0.  
   PHIJP1 = -(VUP-V) + FLOAT(JMN)*DELV  
   UMJP1 = ARS IN(SORT(GAMP1*RUP*RUP - GAMM1))  
   TXUP = XUP(JNDEx+1)  
   TYUP = YUP(JNDEx+1)  
   
   IF (VIN .EQ. VLOW .AND. VLOW .EQ. VOUT) GO TO 170  

C****CALCULATE COORDINATES FOR LOWER TRANSITION ARC - UNROTATED
KDEX = KNDEX + 1
NUM = 0
V = AMAX1(VIN,VOUT)
KEEP = KMN + 1
DO 30 KK=1,KMN
K = KEEP - KK
NUM = NUM + 1
PHIK = PHIKP1 - DELV
FN = K
FOFX = F(V,FN)
CALL ROOT (X0,X2,XINTL,FOFX,FOPRS,TR)
TX = TR*SIN(PHIK)
TY = TR*COS(PHIK)
EMWJ = TAN(-PHIKP1)
UMJ = ARSIN(SCRT(GAMPI*TR*TR - GAMM1))
EMJ = -TAN((PHIK+UMJ+PHIKP1+UMKP1)/2.)
TEMP = TYLO - EMWJ*TXLO
TEMPP = TY - EMJ*TX
TXLO = (TEMP - TEMPP)/TEMPP
TYLC = ((EMJ*TEMP) - (EMWJ*TEMPP))/TEMPP
PHIKP1 = PHIK
UMKP1 = UMJ

CC      SAVE EVERY "NPER-TH" POINT
N = NUM - (NUM/NPER)*NPEP
IF (N .GT. 0) GO TO 30
KDEX = KDEX - 1
STML(KDEX) = 1./TR
XLOW(KDEX) = TXLO
YLOW(KDEX) = TYLO
ETALOW(KDEX) = -PHIK

30 CONTINUE

100 IF (VIN .EQ. VUP .AND. VUP .EQ. VOUT) GO TO 200

C****CALCULATE COORDINATES FOR UPPER TRANSITION APC - UNROTATED
JDEX = JNDEX + 1
NUM = 0
V = AMIN1(VOUT,VIN)
JEEP = JMN + 1
DO 41 JJ=1,JMN
J = JEEP - JJ
NUM = NUM + 1
PHIJ = PHIJF1 - DELV
FN = -J + 2
FOFX = F(V,FN)
CALL ROOT (X0,X2,XINTL,FCFX,FOPRS,TR)
TX = TR*SIN(PHIJ)
TY = TR*COS(PHIJ)
EMWJ = TAN(-PHIJP1)
UMJ = ARSIN(SCRT(GAMPI*TR*TR - GAMM1))
EMJ = TAN((PHIJ+UMJ-PHIJP1+UMKP1)/2.)
TEMPP = TYUP - EMWJ*TXUP
TFMPP = TY - EMJ*TX
TEXUP = ((EMJ*TEMP) - (EMWJ*TEMPP))/TFMPP
TYUP = ((EMJ*TEMP) - (EMWJ*TEMPP))/TFMPP

23
PHIJPl = PHIJ
UMJP1 = UMJ

CC  SAVE EVERY "NPER-TH" POINT
N = NUM - (NUM/NPER)*NPER
IF (N .GT. 0) GO TO 41
JDEX = JDEX - 1
STMU(JDEX) = 1./TR
XUP(JDEX) = TXUP
YUP(JDEX) = TYUP
ETAUP(JDEX) = -PHIJ

41 CONTINUE

CC  MISCELLANEOUS OUTPUT
200 WRITE (6,622)
   622 FORMAT (/54X,24HMISCELLANEOUS PARAMETERS//)
WRITE (6,1000) SSMTN,SMTN,SMOUT,SSMOUT
1000 FORMAT (/25X,8HM*(IN) =,F9.4,3X,7HM*(IN) =,F9.4,10X,8HM*(OUT) =,F9.4
   1,5X,9HM*(OUT) =,F9.4)
WRITE (6,1001) SMLOW,SMLOW,SMUP,SSMUF
1001 FORMAT (/25X,9HM*(LOW) =,F9.4,2X,8HM*(LOW) =,F9.4,11X,7HM*(UP) =,*
   * F9.4,6X,8HM*(UP) =,F9.4)

KINDEX = KINDEX + 1
JINDEX = JINDEX + 1

RETURN
END

$IPFTC ROO  DECK
SUBROUTINE ROOT (X0,X2,XINTL,FOFX,FUNC,X1)

DOUBLE PRECISION X,XX0,XX2

C  WE ARE SEEKING AN X SUCH THAT FUNC(X) = FOFX WHERE FOFX IS A KNOWN
C  FUNCTIONAL VALUE
C  1 LOCATE FOFX IN (F0,FX) OR (FX,F2) WHERE FX IS THE PREVIOUS
C  APPROXIMATION TO FOFX
C  2 LET X = 1/2(XX0+X) OR X = 1/2(X+XX2)
C  3 IS FUNC(X) = FOFX ? IF NOT, REPEAT PROCEDURE

XX0 = X0
XX2 = X2
F0 = FUNC(XX0)
F2 = FUNC(XX2)
IF ( FOFX .LT. F0 .AND. FOFX .LT. F2 .OR. FOFX .GT. F0 .AND.
   FOFX .GT. F2 ) GO TO 1005

IF (ABS(FOFX-F0) .LT. .00001) GO TO 1007
IF (ABS(FOFX-F2) .LT. .00001) GC TO 1008

X = XINTL
KCOUNT = 0
1000 X1 = X
   KCOUNT = KCOUNT + 1
   A = FOFX - F2
   FX = FUNC(X)
   IF (KCOUNT .GE. 60) WRITE (6,1004) KCOUNT,X,FX,FOFX
1004 FORMAT (14H,9H KOUNT, G16.9,9H X, G16.9,9H FX, G16.9,19H FOFX, G16.9,19H)
   IF (ABS (FX-FOFX) .LE. .0001) RETURN
   IF (KOUNT .EQ. 75) GO TO 1002
   IF (A*(FX-FCFX) .LT. 0.) GO TO 1001
   XX0 = X
   X = (X+XX2)/2.
   GO TO 1000

1001 XX2 = X
   X = (XX0+X)/2.
   F2 = FX
   GO TO 1000

1002 WRITE (6,1003)
1003 FORMAT (/30X,62H ITERATIONS HAVE BEEN PERFORMED WITHOUT CONVERGING TO A ROOT)
   RETURN

1005 WRITE (6,1006) FOFX
1006 FORMAT (/10X,7HF(X) = ,G16.9,31H IS OUTSIDE OF SPECIFIED LIMITS)
   RETURN

1007 X1 = X0
   RETURN

1008 X1 = X2
   RETURN

END

$IBFTC FELI  DECK
FUNCTION FOFRS(X)
DOUBLE PRECISION X
COMMON/FACTOR/PERM,GAMM1,GAMM1,GAM

ARG1 = 2.*GAMM1/(X*X) - GAM
ARG2 = 2.*GAMM1*X*X - GAM
   IF (ABS(ARG1) .GT. 1.0 .OR. ABS(ARG2) .GT. 1.0) WRITE (6,1) ARG1
     ARG2
     FORMAT (/14X,61HARGUMENT OF ARCSIN IS OUTSIDE DOMAIN OF DEFINITION IN ARG1 = ,G16.9,11H ARG2 = ,G16.9)
   FOFRS = PERM*ARSIN(ARG1) + APSIN(ARG2)
   RETURN
END

$IPFTC ROTR  LIST, DECK
SUBROUTINE ROTORS (CALP)

C       CALCULATE (1) ROTATED BLADE COORDINATES
C       (2) INLET AND OUTLET ELAFT SPACING

DIMENSION XLOWN(80), YLOWN(80), SMLN(80), ETAUN(80), XCLOW(35),
   1 YCLOW(35), ETAUCL(35), XLOWO(80), YLOWO(80), SMLO(80), ETAULO(80)
DIMENSION XSN(11), YSN(11), ETAASN(11), XUPN(80), YUPN(80), SMUN(80),
   1 ETAUN(80), XCP(35), YCP(35), ETAUC(35), XUPC(80), YUPC(80), SMUC(80),
   2 ETAUC(80), XSO(11), YSO(11), ETAUSO(11)

COMMON/LNK1/XOML(100), YGML(100), SML(100), ETAAML(100), XL,XOMU(100),
CETAT = BETAT * CCNVE4

CALCULATE COORDINATES FCF LOWER TRANSITION ARC - ROTATED

1 YOMU (100), SMU (100), ETAHY (100), NU
COMMON/VIN, VOUT, VLOW, VUP, BETAN, NPEP, KMAXN, FMAXO, JMAXN, JMAXC,
1 RIN, FOUT, BLOW, RUP
COMMON/ROT2/VLOW (400), YLOW (400), STM (400), ETALOW (400), KINDEX,
1 XUP (400), YUP (400), STMU (400), ETAUP (400), INDEX
COMMON/IWB2/STM, SMOUT, FETAT, CSTA, SIGMAO, CSTAO, GSTAPI, GAMMA,
* CONVF, FCQCNV

IF (CALP .NE. 0.) GO TO 2
WRITE (6, 100) BETAN
100 FORMAT (/10X,15HINPUT BETA(0) =F10.4,4H DIG//)
GO TO 3

2 IF (CALB .NE. 1.) GO TO 3
WRITE (6, 101) BETAN, GSTAPO
101 FORMAT (/10X,20HCalculated BETA(0) =F10.4,4F DIG,5X,14WHEN G*(0
1UT) =F10.5)

3 BETAT = BETAT * CONVER

ALPHLN = (VIN - VLOW) - PETAN
ALPHLO = -(VOUT - VLOW) - BFTAT
IF (ALPHLN .LE. 0. .AND. ALPHLO .GE. 0.) GO TO 4
WRITE (6, 102)
102 FORMAT (/27X,79H(VLOW) MUST BE GREATER THAN OR EQUAL TO V(IN) - B
1ETA(IN) AND V(OUT) + ETA(OUT))
STOP

C****CALCULATE COORDINATES FCF LOWER TRANSITION ARC - ROTATED

4 SINAL0 = SIN(ALPHLN)
COSAL0 = COS(ALPHLN)
ABSAL0 = ABS(ALPHLN)
SINALO = SIN(ALPHLO)
COSALO = COS(ALPHLO)
ABSALC = ABS(ALPHLO)
KDEX = KINDEX + 1
KN = (KMAXN/NPEP) + 2
KO = (KMAXO/NPEP) + 2

DO 5 KK = 1, KNDFX
K = KDEX - KK
KN = KN - 1
KO = KO - 1
IF (KN .LE. 0) GO TO 6
SMLN(KN) = STM(K)
XLOWN(KN) = YLOW(K) * SINALN + XICW(K) * COSALN
YLOWN(KN) = YLOW(K) * COSALN - XICW(K) * SINALN
ETALOW(KN) = ETALOW(K) + ABSALN

6 IF (KO .IE. 0) GO TO 5
SMLO(KO) = STM(K)
XLOWC(KO) = YLOW(K) * SINALO - XICW(K) * COSALO
YLOWC(KO) = YLOW(K) * COSALO + XICW(K) * SINALO
ETALOW(KO) = ETALOW(K) + ABSALO
CONTINUE

ALPHUN = (VUP - VIN) - BFTAN
ALPHUC = -(VUP - VOUT) - BETAT
IF (ALPHUN .LE. 0. .AND. ALPHUC .GE. 0.) GO TO 7
WRITE (6, 103)

STOP
C****CALCULATE COORDINATES FOR UPPER TRANSITION ARC - ROTATED

7 SINAUN = SIN(ALPHUN)
COSAUN = COS(ALPHUN)
ABSAUN = ABS(ALPHUN)
SINAUO = SIN(ALPHUO)
COSAOU = COS(ALPHUO)
ABSAOU = ABS(ALPHUO)
JDEX = JNDEX + 1
JN = (JMAXN/NPER) + 2
JO = (JMAXO/NPER) + 2

DO 8 JJ=1,JNDEX
J = JDEX - JJ
JO = JO - 1
JN = JN - 1
IF (JO .LE. 0) GO TO 9
SMUN(JO) = STMU(J)
XUPO(JO) = YUP(J) * SINAUN - XUP(J) * COSAUN
YUPO(JO) = YUP(J) * COSAUN + XUP(J) * SINAUN
ETAUO(JO) = ETAUP(J) + ABSAOU

9 IF (JN .LE. 0) GO TO 8
SMUJ(JN) = STMU(J)
XUPN(JN) = YUP(J) * SINAUN + XUP(J) * COSAUN
YUPN(JN) = YUP(J) * COSAUN - XUP(J) * SINAUN
ETAN(JN) = ETAUP(J) + ABSAUN
8 CONTINUE

C****CALCULATE THE INLET AND OUTLET (DIMENSIONLESS) BLADE SPACING
YLASTI = YUPN(1) + TAN(BETAN)*(XLOWN(1) - XUPN(1))
GSTARI = YLOWN(1) - YLASTI
YLASTO = YUPO(1) + TAN(BETANH)*(XLOWO(1) - XUPO(1))
GSTARO = YLOWO(1) - YLASTO

DALPH = 5.*CONVER

C****CIRCULAR ARC (LOWER)
ALPH = ALPHLO + DALPH
ALPLOW = ALPHLN
KOUNT = 0
10 KOUNT = KOUNT + 1
XCLOW(KOUNT) = FLOW*SIN(ALPLOW)
YCLOW(KOUNT) = FLOW*COS(ALPLOW)
ETACL(KOUNT) = ABS(ALPLOW)
ALPLOW = ALPLOW + DALPH
IF (ABS(ALPH-ALPLOW) .LE. .001) GO TO 11
IF (ALPHLO .LT. ALPLOW .AND. ALPLOW .LT. ALPH) ALPLOW = ALPHLO
GO TO 10
11 NCL = KOUNT

C****CIRCULAR ARC (UPPER)
ALPH = ALPHUC + DALPH
ALPHUP = ALPHUN
KOUNT = 0
12 KCOUNT = KCOUNT + 1
    XCUP(KCOUNT) = RUP*SIN(ALPHUP)
    YCUP(KCOUNT) = RUP*COS(ALPHUP)
    ETACU(KCOUNT) = ABS(ALPHUP)
    ALPHUP = ALPHUP + DALPH
    IF (ABS(ALPH-ALPHUP) .LE. .001) GO TO 13
    IF (ALPHUP .LT. ALPH .AND. ALPHUP .LT. ALPH) ALPHUP = ALPHUP.
    GO TO 12
13 NCU = KCOUNT

C****CALCULATE COORDINATES FOR STRAIGHT LINE POPTION OF UPPER ARC

    KCOUNT = 1
    IF (XLOWN(1) .LE. XUPN(1)) GO TO 15
    WRITE (6,106)
106 FORMAT (8H0, ROTRR), 4X, 71HUPPER SURFACE INLET LONGER THAN LOWER SUR
    *FACE INLET - CASE TERMINATED)
    STOP
15 DELX = (XUPN(1) - XLOWN(1))/10.
    IF (XLOWO(1) .GE. XUPO(1)) GO TO 16
    WRITE (6,105)
105 FORMAT (8H0, ROTRR), 4X, 73YSURFACE OUTLET LONGER THAN LOWER SU
    *RFACE OUTLET - CASE TERMINATED)
    STOP
16 DELXO = (XLOWO(1) - XUPO(1))/10.
    XSIN = XUPN(1)
    YSIN = YUPN(1)
    YSOUT = YUPC(1)
    YSOOUT = YUPO(1)
    ETASN(KCOUNT) = XSIN
    YSN(KCOUNT) = YSIN
    ETASO(KCOUNT) = ETAUN(1)
    XSO(KCOUNT) = XSOUT
    YSO(KCOUNT) = YSOOUT
    ETASO(KCOUNT) = ETAUO(1)
    TANBN = TAN(ETAN)
    ABSN = ABS(ETAN)
    TANSO = TAN(ETAT)
    ABST = ABS(ETAT)
    DO 14 I=2,11
    XSIN = XSIN - DELXI
    XSOUT = XSOUT + DELXO
    YSIN = YUPN(1) + TANBN*(XSIN - XUPN(1))
    YSOUT = YUPO(1) + TANSO*(XSOUT - XUPO(1))
    ETASN(I) = ABSBN
    XSN(I) = XSIN
    YSN(I) = YSIN
    ETASO(I) = ABST
    XSO(I) = XSOUT
    YSO(I) = YSOOUT

C****PREPARE DATA FOR BOUNDARY LAYER PROGRAM

    CSTAR = SORT((YLOWO(1) - XLOWN(1))**2 + (YLOWO(1) - YLOWN(1))**2)
    SIGMAI = CSTAR/GSTAPI
    SIGMAC = CSTAR/GSTAF0
    WRITE (6,104) GSTAPI, SIGMAI, CSTAP, GSTAF0, SIGMAC

28
C STORE ROTATED BLADE COORDINATES FOR LOWER SURFACE
   KN = (KMAXN/NPFR) + 1
   DO 310 I=1,KN
       XOML(I) = XLOWN(I)
       YOML(I) = YLOWN(I)
       SML(I) = SMIN(I)
       ETAML(I) = ETALN(I)
   310 CONTINUE
   NL = KN
   NCL = NCL-1
   DO 311 I=2,NCL
       NL = NL + 1
       XOML(NL) = XCLOW(I)
       YOML(NL) = YCLOW(I)
       SML(NL) = SSMLOW
       ETAML(NL) = ETACL(I)
   311 CONTINUE
   KO = (KMAXO/NPFR) + 1
   DO 312 I=1,KO
       J = KO + 1 - I
       NL = NL + 1
       XOML(NL) = XLOWO(J)
       YOML(NL) = YLOWO(J)
       SML(NL) = SMLO(J)
       ETAML(NL) = ETAO(J)
   312 CONTINUE
C STORE ROTATED BLADE COORDINATES FOR UPPER SURFACE
   DO 313 I=1,10
       J = 12 - I
       YOMU(I) = YSN(J)
       SMU(I) = SSMIN
       ETAMU(I) = ETAUJ(J)
   313 CONTINUE
   NU = 10
   JN = (JMAXN/NPFR) + 1
   DO 314 I=1,JN
       NU = NU + 1
       XOMU(NU) = XUPN(I)
       YOMU(NU) = YUPN(I)
       SMU(NU) = SMUN(I)
       ETAMU(NU) = ETAOU(I)
   314 CONTINUE
   NCU = NCU-1
   DO 315 I=2,NCU
       NU = NU + 1
       XOMU(NU) = XCU(I)
       YOMU(NU) = YCU(I)
       SMU(NU) = SMU(I)
       ETAMU(NU) = ETACU(I)
CONTINUE
  J = J + 1 - I
  NU = NU + 1
  XOMU(NU) = XUPO(J)
  YOMU(NU) = YUPO(J)
  SMU(NU) = SMUO(J)
  ETAMU(NU) = ETAUO(J)
CONTINUE
  DO I = 2, NST
  SW = NU + 1
  XOMU(SW) = XSO(I)
  YOMU(SW) = YSO(I)
  SMU(SW) = SSMOUT
  ETAMU(SW) = ETASO(I)
CONTINUE
RETURN
END

SUSROUTINE INPUT2 (XM, YM, VVCR, ETAM, NN)

LOGICAL ERROR, TRANS, SEFPN

COMMON/CODE/KCDE, DELSRL, DELSPU
COMMON/LNK2/UPMAC, SMOUT, ALP1M, CSTAR, SIGMAO, GSTARO, GSTARI, GAMMA,
  * CONVER, RECONV
COMMON/CT03L/ALPH1, SPA, XMAX, TF, NTUPBU, NTUPBL, CTHETU, CTHETL
COMMON/C1/GAM, R, PTZ, TTZ, UPMACH, NST, NVP, NTUPB, KDUM, KEM, KSTMH,
  1KSPNL, KLE, KATCH, CTHET, DLAM, TLAM, DTURB, TTURB, KERR, KGRAD, KSDE, KLM,
  2KMAIN, KPROF, X(100), Y(100), PPOS (100), UF (100), MF (100), POPTZ (100),
  3VVCPC (100), IWL (100), ETA (100)
COMMON/CK/XCM (100), YOM (100), S (100), SOL (100), AE (100), TSE (100),
  1TAWL (100), TAWT (100), TBAR (100), RN (100), SW (100), SUTHL (100),
  2RHSR (100), RHSE (100), HEAW (100), HEAD (100), NUK (100), MUBAR (100),
  3AA (100), BB (100), FF (100), DMDS (100), DMCL (100)
COMMON/C9/ERROR, TRANS, SEFPN

DIMENSION XM(NN), YM(NN), VVCP(NN), ETAM(NN)

C THE VVCR DISTRIBUTION IS RECEIVED FROM THE ROTCP PROGRAM
C IT WILL BE USED TO CALCULATE THE PPRESSUPF DISTRIBUTION

ERROR = .FALSE.
TRANS = .FALSE.
SEFPN = .FALSE.

C BOUNDARY LAYER SETUP
KATCH = 0
KSPNL = 1
KLE = 1
TLAM = 0.
DLAM = 0.
DTURB = 0.

NST = NN
DO 2 I = 1, NST
  XOM(I) = XM(I)
2 CONTINUE
YOM(I) = YM(I)
X(I) = XOM(I)*XMAX
Y(I) = YOM(I)*XMAX
TWA(I) = TTZ
ETA(I) = ETAN(I)
2 VOCVR(I) = VVCR(I)
ALPH1 = -ALF1M
SPA = XMAX/SIGMAO

IF (KCODE .NE. 0) GO TO 3
WRITE (6,100)
100 FORMAT (1H1,53X,23H*** UPPER SURFACE ***)
GO TO 4
3 WRITE (6,101)
101 FORMAT (1H1,53X,23H*** LOWER SURFACE ***)
4 WRITE (6,1020) NST,NVP,GAM,P,PTZ,TTZ,UMACH,XMAX,NTURB,CTETU,TE
1020 FORMAT (1H0/5X,22HBOUNDARY LAYER - INPUT///5X,3Hnst,5X,3HNVP,7X,
* 3Hgam,11X,1HR,12X,3HPTZ,11X,3HTTZ,9X,6HUMACH,8X,4HMAX,7X,
* 5HNTURB,6X,6HCETETA,7X,2HTE/4X,2(I3,5X),F7.3,3(5X,F9.2),5X,F9.4,
* 5X,8F5.6X,13,7X,F7.3,5X,F7.5///)
WRITE (6,1090) KPRE,KGRAD,KDF,KLAM,KMAIN,KPROF,KE
1090 FORMAT (/6X,4HKPRE,7X,5HKGRAD,7X,4HKSDE,8X,4HKLAM,7X,5HKMAIN,7X,5HK
1PROF,7X,5HKKM,7X,12,9X,12,10X,12,9X,12,10X,12,10X,12///)
WRITE (6,1032) (XOM(I),YOM(I),VOCVR(I),TWA(I),I=1,NST)
1032 FORMAT (9X,3HOM,7X,3HYOM,9X,4HVCR,9X,4HTWA/5X,F8.5,5X,F8.5,5X,
1F11.5,F9.2))
IF(NST.GT.100.OR.NTURB.GT.NST.05.KFMBLT.0.OP.KE.1.0P.KSPN.0T.
10,0P.KSPLN.0T.0P.KLFT.0P.KFMB.0P.KATCH.LT.0P.KATCH.GT.
21) GO TO 70
RETURN
70 ERROR = .TRUE.
WRITE(6,1170)
1170 FORMAT (///10X,48HERROR IN INPUT DATA. PECHECK INPUT INSTRUCTIONS
1)
RETURN
END $IEFTC ELAYR CECK
SUBROUTINE ELAYR

COMMON/CODF/KCDE,EELSRl,DELSRU
COMMON/CTOL/H/AL,ST,PAM,TE,NTURBU,NTURPL,CHTETU,CHETL
COMMON/C1/GAM,P,PTZ,TTZ,UMACH,NSN,NVPE,NTURB,KPVM,KEM,KSMTH,
1KSPLN,KLF,KATCH,CHET,DLAM,TLAM,DTURB,TTURB,KFRR,KGRAD,KSDRE,KLAM,
2KMAIN,KPROF,X(100),Y(100),PRFS(100),UE(100),ME(100),POTZ(100),
3VOCR(100),TWA(100),ETA(100)
COMMON/C2/PS1,TSZ,UZ,AS7,AT7,HSZH,HTZ,H1,MSZH,MTS,ZFlush,CP,
1PR,TC,ARCL
COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TE(100),
1TAWL(100),TAWT(100),TEAP(100),REW(100),SW(100),STWEL(100),
2RHSH(100),RHSF(100),HEADW(100),HEADR(100),XUM(100),MURAP(100),
3AA(100),BB(100),FF(100),DDUS(100),DMDS(100),DMDL(100)
COMMON/C4/THET(100),DELSR(100),DELTA(100),FCRM(100),
CALL PRECAL
IF (ERROR) RETURN
CALL LAMNAP
IF (ERROR) RETURN
IF (SEPBN) GO TO 20
IF (.NOT. TRANS) GO TO 20
CALL TURBLN
IF (FRRCN) RETURN
20 CALL PROFIL
IF (KODE .EQ. 1) GO TO 9
DELS = DELSR(NST)
THETS = THET(NST)
KODE = 1
RETURN
9 DELF = DELSR(NST)
THETP = THET(NST)
ALPH1 = AL
SPA = SP
CALL APMIX (ALPH1, DELS, DELF, THETS, THETP, TF, SPA, MF(NST))
KODE = 0
RETURN

END
$IEPTC  PRECL  LIST, DECK
SUBROUTINE PRECAL

COMMON/GAMPM/GMP1, GMM1
COMMON/CTOBL/DUM (2), XMAX, DUM2 (5)
COMMON/C1/GAM, P, PTZ, TTZ, UPRACH, NST, NVE, NTURB, KPVM, KEM, KSNTH,
1KSPIN, KLE, KATCH, CTHET, DLAM, TLAM, DTURB, TTURB, KPRE, KGRAD, FSDE, KLAM,
2KMAIN, KPROP, X(100), Y(100), PRES(100), ME(100), ME(100), P0PTZ(100),
3VOVCR(100), TAWL(100), ETA(100)
COMMON/C2/PSZ, TSZ, UZ, ASZ, ATZ, RHSZ, PHTZ, NUSZ, MUTZ, NUSZ, MUTZ, CP,
1PR, TC, ARCL
COMMON/C3/XCM(100), XOM(100), S(100), SOL(100), AE(100), TSE(100),
1TAWL(100), TAWT(100), TBAP(100), PW(100), SW(100), SUTHL(100),
2RHSW(100), RHSE(100), HEALW(100), HEADE(100), NUW(100), MUBAR(100),
3AA(100), BB(100), FF(100), DUDS(100), DMTS(100), DMDL(100)
COMMON/C9/ERROR, TRANS, SEPPN
DIMENSION SDER(100), CM!!(20), CPR(20), CTC(20)
REAL NUSZ, MUTZ, NUSZ, MUTZ, MUSL, ME, NUW, MUBAR
LOGICAL ERROR, TRANS, SEPPN

C READ DATA FOR MU, PB, AND TC CURVE FITS
C
DATA (CM!!(1), I = 1, 5) = /-.01945170, 1.3019531, -.34511323,
1.068277826, -.00566593/
DATA (CPR(I), I = 1, 5) = /9557, -.234136, 1078624,
1.0236214, -.00202863/
DATA (CTC(I), I = 1, 5) = /-.03839323, 1.2697427, -.3091252,
1.08743781, -.009674725/

C
GMP1 = GAM + 1.
GMM1 = GAM - 1.

32
C INITIALIZE STATIC AND TOTAL PARAMETERS

TSLE = 518.6E8
TSLM = 298.160
MUSLE = 3.711702E-7
MUSLM = 1.777029E-5
TCSLE = 3.202206E-3
TCSLM = 2.561796E-2
TSZ = TTZ/(1. + GM**1/2. * UP**2)
PSZ = PTZ*(TSLE/TTZ)**(GM**1)
RHSZ = PSZ/R/TSZ
RHTZ = PTZ/R/TSZ
ASZ = SQRT(GM*R*TSZ)
ATZ = SQRT(GM*R*TSZ)
UZ = UP**ASZ
CP = R*GM
IF (KEM.PC.1) GO TO 10
TCON = 198.60
TR1 = TSZ/TSLM
TR2 = TSZ/TSLM
GO TO 20
10 TCON = 110.33
TR1 = TSZ/TSLM
TR2 = TSZ/TSLM
20 CALL CURVFT(CPR,PR,TR1,0,4,0)
CALL CURVFT(CTC,TC,TR1,0,4,0)
CALL CURVFT(CMU,MUSZ,TR1,0,4,0)
CALL CURVFT(CMU,MUTZ,TR2,0,4,0)
IF (KEM.PC.1) GO TO 30
TC = TC*TCSLF
MUSZ = MUSZ*MUSLE
MUTZ = MUTZ*MUSLE
GO TO 40
30 TC = TC*TCSLM
MUSZ = MUSZ*MUSLM
MUTZ = MUTZ*MUSLM
40 NUSZ = MUSZ/PHSZ
NUTZ = MUTZ/RHTZ
C CALCULATE GEOMETRY RATIOS AND ARC LENGTHS
C
S(1) = 0.
DO 50 I=2,NST
50 S(I) = S(I-1) + SQRT((X(I) - X(I-1))**2 + (Y(I) - Y(I-1))**2)
ARCL = S(NST)
DO 60 I=1,NST
60 SOL(I) = S(I)/ARCL
C CALCULATE PFS, UE, ME, POPTZ, AND VOVCR AT EACH STATION
C
VELOCITY OVER CRITICAL VELOCITY GIVEN AS INPUT
150 DO 160 I=1,NST
POPTZ(I) = (1. - GM1/GM**1 * VOVCR(I)**2)**(GM/GM1)
UE(I) = SQRT(2. * GM1/GM**1 * PTZ**2 / RHTZ**2 * (1. - POPTZ(I)**2) * GM1/GM)
TSE(I) = TTZ - UE(I)**2 / (2. * CT)
AE(I) = SQRT(GM**R*TSE(I))
ME(I) = UE(I)/AE(I)
160 PRES(I) = PCPTZ(I)*PTZ
C PRINT INITIAL CALCULATED PARAMETERS

170 WRITE(6,1000)
    WRITE(6,1010) PSZ,TSZ,HZ,ASZ,ATZ,RHSZ,PHTZ,MUSZ,MUTZ,NUSZ,NUTZ,CP,
    1PR,TC,ARCL
    WRITE(6,1020) (I,PRES(I),UE(I),ME(I),POPTZ(I),VOVCR(I),I=1,NST)

C PRINT GEOMETRY PARAMETERS

200 IF (KPRE,NE,1) GO TO 210
    WRITE(6,1030) (I,X(I),Y(I),S(I),XCM(I),YCM(I),SCL(I),I=1,NST)

C CALCULATE OTHER NECESSARY PARAMETERS AT EACH STATION

210 DO 220 I=1,NST
    TEM1 = 1.*GMM1/2.*ME(I)**2
    RHSW(I) = PRES(I)/P/TWAL(I)
    RHSP(I) = PRES(I)/P/TSP(I)
    HEADW(I) = 5.*RHSW(I)*UE(I)**2
    HEADE(I) = 5.*PHSE(I)*UP(I)**2
    SW(I) = TWAL(I)/TTZ-1.
    SUTHL(I) = SQRT(TWAL(I)/TTZ)*(TTZ+TCON)/(TWAL(I)+TCON)
    NWU(I) = SUTHL(I)*NUTZ*(1.+SW(I))**2*TEM1**2*(GAM/GMM1)
    RW(I) = UE(I)*S(I)/NWU(I)
    TAWL(I) = TSE(I)*(1.+PR**(1./2.))*(TEM1-1.)
    TAWT(I) = TSE(I)*(1.+PR**(1./3.))*(TEM1-1.)
    TBAP(I) = 5.*(TWAL(I)+TSE(I))+.22*PR**(1./3.)*(TTZ-TSF(I))
    MUBAR(I) = MUTZ*SUTHL(I)*TBAP(I)/TTZ
    BR(I) = ME(I)*ATZ/NUTZ*(TSE(I)/TTZ)**2*(GMP1/2.*GMM1)
    AA(I) = BB(I)*TSE(I)/TBAR(I)*MUBAR(I)/MUTZ)**2.68
    FF(I) = 1.+1.599*ME(I)**2+.66*SW(I)+.2101*SW(I)*ME(I)**2+.0114*ME(I)
           1**4+.0180*SW(I)*ME(I)**4+.1825*SW(I)**2+.0735*SW(I)**2*ME(I)**2
    2+.0073*SW(I)**2*ME(I)**4

220 CONTINUE

C FIDITT DIFFERENCE METHOD USED TO CALCULATE VELOCITY AND MACH NUMBER

C GRADIENTS ALONG THE SURFACE

C DUDS(I) = (UE(2) - UE(I))/(S(2) - S(I))
C DMDS(I) = (ME(2) - ME(I))/(S(2) - S(I))

DO 230 I=2,NST
    IM = I - 1
    IF (I.LT.EQ.,NST) GO TO 230
    IN = I + 1
    DUDS(I) = (UE(IP)-UE(IM))/(S(IP)-S(IM))
    DMDS(I) = (ME(IP)-ME(IM))/(S(IP)-S(IM))

230 CONTINUE

C DDS(NST) = (UE(NST) - UE(1))/(S(NST) - S(1))
C DMDS(NST) = (ME(NST) - ME(1))/(S(NST) - S(1))

240 DO 250 I=1,NST
250 DMOL(I) = ARCL*DMDS(I)

C PRINT OTHER CALCULATED PARAMETERS

C IF(KPRE,NE,1) GO TO 260
    WRITE(6,1050) (I,AE(I),TSE(I),TWAL(I),TAVL(I),TAWT(I),TEAP(I),
           I=1,NST)
CITAUW (100), NUSS (100), DTDY (100), HTRAN (100), CPN (100)
COMMON/C5/SHAPL (100), SHAPK (100), ENS
COMMON/C6/FTRAN, FORMS
COMMON/C7/INST, ITRAN, ISEP
COMMON/C9/ERROR, TRANS, SEPRN

DIMENSION CCFLN (100), COEML (100), SHFAR (100), DTH (100)
DIMENSION CCN (20), CRCR (20), CDIF (20), CSHR (20), CCRN (20), CDTH (20)
DIMENSION STAB (505), CTAP1 (505), CTAP2 (505)
REAL MUSZ, NUSZ, MUTZ, NUTZ, ME, NUW, NUEAF, NUSS, NUPK, KBAR, INT1, INT2
LOGICAL FFRC, TRANS, SEPRN
EXTERNAL FFNC, INT1, TTURR

READ DATA PCR CCIRLN (1), RCFIT, DIFF, SHEAP, CPN, AND DTH CURVF PITS

DATA (CCN(I), I=1,6)/-0.8178, -0.6670, -0.3143,
  1.00873, -0.1657, -0.01052/
DATA (CRCR(I), I=1,6)/5.47073, 43.6053, 227.198,
  1.2067, 0.4, -27172.7, 13691.2/
DATA (CDIF(I), I=1,6)/903.785, 26365.0, 3.85695E+5,
  1.11042E+6, -4.53853E+7, -7.70276E+7/
DATA (CSHR(I), I=1,16)/-224488, -1.91539, -9.894, -68.13498,
  1.001512, -1.4768, -10.52925, -152.2781, -0.02406, -0.15629,
  1.45743, -126.23395, 0.00752, 0.05585, 9.17838, -39.40644/
DATA (CCRN(I), I=1,16)/2.20256, -19.7211, -24.0495, -1400.002,
  -0.50979, -10.88012, -62.4419, -5081.76, -0.14343, 2.279845,
  1129.7008, -6257.848, -0.0270567, -1.677051, 5.7.4397, -2552.266/
DATA (CDTH(I), I=1,16)/-8.02829, -4.30979, 8.8244, 36.4336,
  -7.42259, 242.293, -16.293, -16394, -7.61942, 285.9795,
  164.1118, -16758, -3.70289, 130.8107, 111.3276/

INITIALIZE PARAMETERS

INST = 0
ITRAN = 0
ISEP = 0
CP(1) = 0.
TAUW (1) = 0.
NUSS (1) = 0.
DTDY (1) = 0.
HTRAN (1) = 0.
CRN (1) = 0.
RTRAN= 0.
IF (CTHET .GT. 0.) KATCH = 1

CHECK CONSISTENCY OF INITIAL VALUES

IF (DIAM.GE.0. AND. TLAM.GE.0. AND. TTURB.GE.0. AND. TTURB.GE.0.)
10 GO TO 10
ERROR = .TRUE.
WRITE(6,1000)
RETURN
10 IF (NTURB .NE. 1) GO TO 30
ITRAN = 1
IF (DTURB.GT.0. AND. TTURB.GT.0.) GO TO 20
ERROR = .TRUE.
WRITE(6,1010)
RETURN
C

20 IF (UE(1) .GT. 0.) GO TO 240
   ERROR = .TRUE.
   WRITE(6,1020)
   RETURN
C
C BEGIN CALCULATION IN LAMINAR REGION - CHFCK FOR INITIAL VALUES
C CALCULATE INITIAL CORRELATION NUMBER
C
30 IF (DLAM .EQ. 0. .AND. TLAM .EQ. 0.) GO TO 70
   IF (UE(1) .GT. 0.) GO TO 40
   ERROR = .TRUE.
   WRITE(6,1030)
   RETURN
40 IF (TLAM .EQ. 0.) GO TO 50
C INITIAL MOMENTUM THICKNESS WAS GIVEN
   TEM1 = 1. *GMM1/2. *ME(1) **2
   CORL(1) = -ATZ *TLAM **2 /NUTZ /SUTHL(1) /ACHCL /TEM1 ** ((3. - GAM) /1(2. *GMM1))
   CORLN(1) = CORML(1) *DMDL(1)
   GO TO 90
C INITIAL DISPLACEMENT THICKNESS WAS GIVEN
50 IF (ABS(DMDL(1)) .GE. 0.0001) GO TO 60
   CORLN(1) = 0.
   TEM1 = 1. *GMM1/2. *ME(1) **2
   FORM(1) = 2.38411 * (1. + (2.79 - 1.78*PR**.5) * ((1. + SW(1)) *TEM1 - 1.)) * (4.6
   15*PR***(1./3.) - 3.65*PR*5) *PR**.5 *TEM1 - 1.1
   THET(1) = DLAM /FORM(1)
   CORL(1) = -ATZ *THET(1) **2 /NUTZ /SUTHL(1) /ACHCL /TEM1 ** ((3. - GAM) /2. *GMM1))
   GO TO 90
60 IF (DMDL(1) .GT. 0.) CALL ROOTB(-1., 0., DLAM, FUNCT, .5E-5, CORLN(1), SL)
   IF (DMDL(1) .LT. 0.) CALL ROOBT(0., -2., DLAM, FUNCT, .5E-5, CORLN(1), SL)
   CORML(1) = CORLN(1) /DMDL(1)
   GO TO 90
C
C NO INITIAL LAMINAR VALUES GIVEN
C CALCULATE INITIAL CORRELATION NUMBER
C
C SHARP LEADING EDGE
70 IF (KLE .NE. 1. .AND. ABS(DMDL(1)) .GE. 0.0001) GO TO 80
   CORLN(1) = 0.
   CORML(1) = 0.
   GO TO 90
C STAGNATION POINT
80 CALL CURVFT (CCN, CORLN(1), SW(1), 0, 5, 0)
   CORL(1) = CORLN(1) /DMDL(1)
   IF (CORL(1) .LT. 0.) GO TO 90
   ERROR = .TRUE.
   WRITE(6,1040)
   RETURN
C
C SOLVE LAMINAR DIFFERENTIAL EQUATION
C CALCULATE CORRELATION NUMBERS ALONG THE SURFACE
C
90 TEM1 = 1. *GMM1/2. *ME(1) **2
   TEM2 = (3. *GAM - 1.) / (2. *GMM1)
   DEL = 0.002 *APCL
   SS = -DEL
   NTAB = 1
CTAB1 (1) = CORML (1)
CTAB2 (1) = CORML (1)
STAB (1) = 0.

100 SS = SS + DEL
SSDEL = SS + DEL
CALL LGRNGE (S, SP, NST, SS, ANS1)
CALL LGRNGE (S, ME, NST, SS, ANS2)
CALL LGRNGE (S, ME, NST, SSDEL, ANS3)
CALL LGRNGE (S, DN, NST, SSDEL, ANS4)
A1 = 0.43631 - 0.00367*ANSL + 0.0481*ANSL**2 + 0.00651*ANSL**3
A2 = 5.43220 + 2.25490*ANSL - 0.06672*ANSL**2 - 0.20637*ANSL**3
A3 = 4.51093 - 10.49775*ANSL - 12.7732*ANSL**2 - 29.5270*ANSL**3
A4 = 0.1831 + 62.76597*ANSL + 115.00996*ANSL**2 + 62.53113*ANSL**3
A = A1*A3*CTAB1 (NTAB)**2 - 2*A4*CTAB1 (NTAB)**3
B = A2**2*A3*CTAB1 (NTAB) + 3*A1*CTAB1 (NTAB)**2

C
FOR SW = 0.0
IF ((CTAB1 (NTAB) .GE. -1)) Go to 101
A = .3953
B = 4.739

101 K1 = 0
SOL1 = SS/ARCL
SOL2 = SSDEL/ARCL
TEM3 = STMP5 (SOL1, SOL2, INT1, K1)
IF (TEM3 .EQ. 0. OR. K1 .EQ. 0) Go to 110
ERROR = .TRUE.
WRITE (6, 1050)
RETURN

110 IF (NTAB .GT. 1) TEM4 = ANS2**(-B)*TEM1**TEM2
TEM1 = 1. + GM1/2.*ANSL**2
TEM5 = ANS3**(-B)*TEM1**TEM2
TEM6 = -A*TEM5*TEM3
IF (NTAB .LE. 1) TEM7 = 0.
IF (NTAB .GT. 1) TEM7 = TEM5/TEM4*CTAB2 (NTAB)
NTAB = NTAB + 1
CTAB2 (NTAB) = TEM6*TEM7
CTAB1 (NTAB) = CTAB2 (NTAB) * ANS4
STAB (NTAB) = SSDEL

C
WHEN SW IS NOT EQUAL TO 0.0 , CURVE FIT RANGE ON CORNL
C IS FROM -.32 TO .16

C
IF (CTAB1 (NTAB) .GT. .50) Go to 120
IF (SS_LT_ARCL) Go to 100
120 IF (KSDE_NE.1) Go to 130
WRITE (6, 1060)
WRITE (6, 1070) (STAB (I), CTAB1 (I), I=1, NTAB)

C
CALCULATE LAMINAR BOUNDARY LAYER PARAMETERS AT EACH STATION

C
130 IF (KLAM_NE.1) Go to 140
WRITE (6, 1080)
140 I = 0
150 I = I + 1
IF (I .EQ. NTURE) ITRAN = -1
IF (S(I).LT.STAB(NTAB)) Go to 160
WRITE (6, 1090)
160 IF (KLE .EQ. 1 .AND. I .EQ. 1) Go to 151
CALL LGRNGF (STAB, CTAB1, NTAB, S(I), CORNL(I))
CALL LGRNGE (STAB, CTAB2, NTAB, S(I), CORML(I))
C OBTAIN SHEAP, CFN, AND DTH FROM CURVE FITS VS CORLN AND SW

151 CALL CURVFT(CSSH, SHEAR(I), CORL(I), SW(I), 3, 3)
   CALL CURVFT(CCRN, CRN(I), CORL(I), SW(I), 3, 3)
   CALL CURVFT(CDH, DTH(I), CORL(I), SW(I), 3, 3)

C FOR SW = 0.0
   IF(CORL(I).GE.-1.0) GO TO 161
   SHEAR(I) = -1.2222*CORL(I) + .26
   CRN(I) = -58.924 * CORL(I) - 67.65
   DTH(I) = -22.222*CORL(I) + 7.1112

C CALCULATE GTHP LAMINAR BCUNDARY LAYER PARAMETERS

161 TEM1 = 1.0 + GMM1/2.0 + ME(I)*2
   THET(I) = SOFT(-CORML(I)*NUTZ*SUTHL(I)*APCL/ATZ*TEM1**((3.0 - GAM)/
     1.0 + GMM1))
   FORM(I) = (-1.1138*CORLN(I) + 2.38411)*((1.0 + SW(I))*TEM1**(-1.0))
   DELSR(I) = THET(I)*FORM(I)
   RTH(I) = UE(I)*THET(I)/NWW(I)
   FORMR(I) = (FORM(I) - SOFT(PRO)*TEM1**(-1.0))/(1.0 + SW(I))
   FORMR(I) = (FORMR(I) - SOFT(WH*TEM1**(-1.0)))/(1.0 + SW(I))
   DELT(I) = THET(I)*DTH(I)*TEM1**(-1.0)/(FCPMT(I) + 1.0)
   SHAPE(I) = DELT(I)**2/NWW(I)*SURE(I)
   IF (I.EQ.1) GO TO 190
   CFRW = 2.0*SHEAR(I)*SOFT(-SOL(I)/MIF(I))/CORML(I)
   CF(I) = CFRW/SORT(RW(I))
   TAULW(I) = CF(I)*HEADW(I)
   NFRI = CFRW*EM**.3/CRN(I)
   DUWW(I) = NFRI*SORT(RW(I))
   DTBY(I) = DUWW(I)*SOUT(RW(I))
   THRTAN(I) = TC*DTBY(I)
   IF (TAWU(I).LT.0.0) GO TO 190
   IF (KATH, NF.0) GO TO 170
   ISEP = I
   SEPPN = .TRUE.
   RETURN

170 ITRAN = -2
   GO TO 270

180 IF (I.EQ.1 AND UF(I).EQ.0.0) GO TO 190
   SHAPK(I) = NUTZ*RTH(I)**2*SUTH(I)**2*(1.0 + SW(I))**4*/ATZ/MET(I)**2/
     1PP(I)/APCL*EMEL(I)**TEM1**((3.0 - GAM))
   GO TO 200

190 SHAPK(I) = 0.07
   200 RTHI(I) = RTH(I)*SUTHL(I)*(1.0 + SW(I))**2*/FF(I)/SORT(TEM1)

C CALCULATE RCRIT TO CHECK FOR INSTABILITY AND TRANSITION

C CALL CURVFT(CFCR, RCRIT, SHAPK(I), 0, 5, 0)
   IF (SHAPK(I) .GT. 0.07) RCRIT = 9.3163
   RCRIT = XFR(RCRIT)
   IF(INST. NE. 0) GO TO 210

C CHECK FOR INSTABILITY

C IF (RTHI(I) .LT. RCRIT) GO TO 270
   RINS = RTHI(I)
   INST = I
   GO TO 270

C CHECK FOR TRANSITION
C
210 K1= 0
     NS= I
     TFM= SIMPS1(SOL(INST),SOL(I),INT2,K1)
     IF (TFM.EQ.0.IOR.K1.EQ.0) GO TO 220
     FPOR = .TRUE.
     WRITE(6,1100)
     RETURN
220 KBAR= TFM/(SOL(I)-SOL(INST))
     CALL CURVFT(CCIF,DIFF,KBAR,0.5,0)
     IF (KBAR .GT. .03) DIFF = 2400.*KBAR+700.0
     PTRAN= RINS+DIFF
     IF (PVT(I).LT.RTPAN) GO TO 270
     IF (I.LT.NTURB) GO TO 270
     ITRAN= -1
     GO TO 270
230 ITRAN= I
C
C COMPUTE INITIAL VALUES FOR TURBULENT SOLUTION
C
240 TRANS= .TRUE.
     IF (DTURB.EQ.0.IOR.TTURP.EQ.0.) GO TO 260
     IF (DTURB.GT.0.IOR.TTURB.GT.0.) GO TO 250
     ERROR = .TRUE.
     WRITE(6,1110)
     RETURN
250 THET(ITRAN) = TTURB
     FORM(ITRAN) = TTURP/TTURB
     TEM1 = 1.+GMM1/2.*ME(ITRAN)**2
     FORMI(ITRAN) = (FORM(ITRAN)-ME*(1./3.)*(TEM1-1.))/((1.+SW(ITRAN))
*TEM1)
260 IF (C100*E*10.0+D0*E*10.0) THET(ITRAN) =
     C1*THET*THET(ITRAN)
     THETTP = THET(ITRAN)*((ISP(ITRAN)/TTZ)**(GMM1/(2.*GMM1)))
     PTRAN= (ME(ITRAN)*ATZ*THETTP/HTZ)**2.686
     IF (PTRAN .LE. 0.) GO TO 265
     FORMS= FORMI(ITRAN)-0.59390+0.6591*ALOG(PTRAN)+0.001272*(ALOG(PTR
AN)**2)
     IF (DTURB.GT.0.IOR.TTURB.GT.0.) FORMS=FORMI(ITRAN)
     RETURN
265 FORMS = 1.4
     RETURN
C
C PRINT OUTPUT
C
270 IF (Klam.NE.1) GO TO 290
     IF (INST.EQ.0.IOR.INST.EQ.1) WRITE(6,1120) I,COPLN(I),SHFAR(I),
*TH1(1),FORMT(I),SHAPL(I),PTH1(I),SHAR(1),PCRT
     IF (INST.NE.0.IOR.INST.NE.1) WRITE(6,1130) I,COPLN(I),SHFAP(I),
*TH1(1),FORMT(I),SHAP(I),PTH1(I),SHAR(1),PCRT
     IF (ITRAN.EQ.-1.OR.ITRAN.EQ.-2) WRITE(6,1140)
280 IF (TTRP.EQ.-1.OR.TTRP.EQ.-2) GO TO 270
     IF (I.EQ.NST) RETURN
     GO TO 150
C
C FORMAT STATEMENTS
C
1000 FORMAT(10X,G00A) NEGATIVE INITIAL VALUE HAS BEEN GIVEN. THIS
    I(S NOT ALLOWED)
1010 FORMAT(///,10X,75X,INITIAL VALUES WERE NOT GIVEN FOR THE TURBULENT
1T BOUNDARY LAYER AT STATION 1)
1020 FORMAT(///,10X,80X,INITIAL VALUES WERE GIVEN FOR THE TURBULENT BO
1UNDARY LAYER AT A STAGNATION POINT)
1030 FORMAT(///,10X,94X,INITIAL VALUES OTHER THAN ZERO WERE GIVEN FOR
1THE LAMINAR BOUNDARY LAYER AT A STAGNATION POINT)
1040 FORMAT(///,10X,106X,FOR THIS INPUT DATA STATION 1 IS ASSUMED TO BE
1 A STAGNATION POINT, STNCRC NO INITIAL THICKNESSES ARE GIVEN.
210X,118X,IN THIS CASE PRESSURE SHOULD DECREASE INITIALLY. EITHER GIVE AN
2 INITIAL VALUE FOR DISPLACEMENT OR MOMENTUM THICKNESS.
410X,60X,OR BEGIN WITH A SHOULDER REGION OF FAVORABLE PRESSURE GRADIENT.
5.)
1050 FORMAT(///,10X,37X,ERROR IN COMPUTING INTEGRAL FOR CORLN)
1060 FORMAT(1H1///,7X,50X,LAMINAR DIFFERENTIAL EQUATION - SOLUTION FOR CO
1RLN///,5(24H
S
CCRLN )///)
1070 FORMAT(///,1X,59X,LAMINAR CALCULATION OF INSTABILITY AND TRANSITION
10X LOCATIONS///,1X,7HSTATION,2X,5HCCRLN,5X,5HSHEAR,5X,3HDTH,6X,5HPO
2MTTH,4X,5HSHAPL,9X,4HHTHI,6X,5HSHEAR,9X,5HBCRT,6X,4HKBAP,10X,4HDT
3FP,9X,5HTRAN)
1090 FORMAT(///,10X,65X,LAMINAR SOLUTION HAS PROCEEDED BEYOND THE RANGE
10X WHERE IT IS VALID)
1100 FORMAT(///,10X,36X,ERROR IN COMPUTING INTEGRAL FOR KBR)
1110 FORMAT(///,10X,54X,IF INITIAL TURBULENT VALUES ARE GIVEN, THEY BE
1TH MUST BE NCNZERO)
1120 FORMAT(I4,1X,5F10.4,1X,F12.1,1X,F10.5,1X,F12.1)
1130 FORMAT(I4,1X,5F10.4,1X,F12.1,24X,F12.5,1X,F12.1,1X,F12.1)
1140 FORMAT(///,10X,85X,LAMINAR SEPARATION HAS OCCURRED. ASSUMED TO BE
1 TRANSITION TO TURBULENT BOUNDARY LAYER)
END
$IEFTC TURBL LIST,DECK
SUBROUTINP TURBLN
COMMON/GAMPY/GMP1,GMPM
COMMON/C1/GAM,F,PTZ,TTZ,UPMACH,USt,NVE,NTHRR,KPVC,KEM,KSMTH,
1KSPML,KLF,KATC,CTHET,CTLAM,TLAM,DTUPE,TTUPE,KEFF,KGRAT,KSDE,KLAM,
2KMAIN,KPROP,X(100),Y(100),PRES(100),UF(100),ME(100),DOTZ(100),
3VOCR(100),TW(100),ETA(100)
COMMON/C2/PSZ,Tsz,UZ,ASZ,ATZ,PHSZ,PHTTZ,PHSZ,MTZ,NUSZ,NUTZ,CP,
1PP,TC,ARCL
COMMON/C3/XCM(100),YOM(100),S(100),SOL(100),AE(100),TSS(100),
1TAW(100),TAWT(100),TBAV(100),PK(100),SK(100),SUTHI(100),
2PHSW(100),PHSE(100),HFEAW(100),HFAE(100),HNSK(100),MURAP(100),
3AA(100),EE(100),FF(100),SEDS(100),MMOS(100),THDL(100)
COMMON/C4/THET(100),DELSR(100),CSRTE(100),CSFTR(100),
1FORM(100),EERMT(100),EETO(100),FH(100),G(100),F(100),
1TAW(100),NUSS(100),DTDY(100),HTHR(100),CF(100)
COMMON/C5/FTAN,FORMS
COMMON/C7/INST,ITTRAN,ISEP
COMMON/CP/XTAB(505),YTAB1(505),YTAB2(505),XTAB
COMMON/C9/ERRCR,TRANS,SFPAR
REAL MUSZ,MSZ,NUTZ,NUHZ,NUHZ,UMAP,NSZ,NUHZ,ME,NUHZ,UMAP,NSZ
LOGICAL FRCR,TRANS,SEPAN
C
C SOLVE TURBULENT BOUNDARY LAYER DIFFERENTIAL EQUATIONS
41
C USING RUNGA-KUTTA
CALL RUNKUT
IF (KSEDE.NE.1) GO TO 10
WRITE(6,1000)
WRITE(6,1010) (XTAB(I),YTAB1(I),YTAB2(I),I=1,NTAB)
10 DO 5 I=1,NTAB
IF (YTAB2(I) .LE. 2.8) GO TO 5
WRITE(6,100)
100 FORMAT (9HO(TURBLN),5X,63HINCOMPLETE FORM FACTOR GREATER THAN
*2.8 - CASE TERMINATED)
STOP
5 CONTINUE
C
C CALCULATE TURBULENT BOUNDARY LAYER PARAMETERS AT EACH STATION
C
DO 30 I=ITRAN,NST
IF (S(I) .LE.XTAB(NTAB)) GO TO 20
ISEP = I-1
SEPRL = .TRUE.
RETURN
20 TEM1 = 1.0*GMM1/2.*ME(I)**2
CALL LGHNSGE(XTAB,YTAB1,NTAB,S(I),F)
THETTR = NUTZ*F**.7886/ME(I)/ATZ
THET(I) = THETTR*(ITZ/TS(I))**(GMF1/(2.*GMM1))
RTH(I) = OR(I)*THET(I)/NUW(I)
CALL LGHNSGE(XTAB,YTAB2,NTAB,S(I),FOMT(I))
FORMF(I) = FORMI(I)*(1.*SW(I))
FORM(I) = FORMF(I)*TEM1*PR**((1./3.)*(TFM1-1.))
DELSR(I) = THET(I)*FORM(I)
PWRM = 2.0/(FORMI(I)**1.0)
IF (FORMI(I).LT.1.02) POWER=100.
DELTA(I) = (1.+POWER)*DELSR(I)
CF(I) = 0.246*EXP(-1.561*FORMI(I))*(UE(I)*THET(I)/NUTZ/(TFM1**(1./(1.0-GAM-1.))))**((-2.68)*TSF(I)/TBAR(I)*(MUBAR(I)/NUTZ)**(.269)
TAUW(I) = CF(I)*HEADE(I)
IF (I.EQ.1) GO TO 30
HTRAN(I) = CF(I)/2.*P**((2./3.)*PHSF(I)*UE(I)*CP*(TAWT(I)-TWAI(I))
DTDY(I) = HTRAN(I)/7C
NUSS(I) = S(I)*DTDY(T)/TWAI(I)
CRN(I) = CF(I)*RW(I)/NUSS(I)
30 CONTINUE
RETURN
1000 FORMAT (1H1//5X,62HTURBULENT DIFFERENTIAL EQUATIONS - SOLUTION FOR
F AND FORM ///4(31H S F FORMI )///)
1010 FORMAT((4(F10.5,2X,F8.1,2X,F7.4,2X)))
END
$IPFTC PROFI LIST, DECK
SUBROUTINE PROFI

COMMON/CCDE/KCDE,DELSRL,DELSRU
COMMON/CTOBL/DUM2,DUM2(5)
COMMON/LINK2/DUM3,CSLL,CSTAB,DUM35,CSGAPI,DUM4(3)
COMMON/C1/GAM,PS,PTZ,TTZ,UPMACH,NCST,NAV,NTURN,KPVN,KPV,KPM,KSMTH,
1KSPLN,KLE,KATCH,CHET,DLAM,TLM,DTUDE,DTUPP,KPF,KGAD,KGDE,KIAM,
2KMAIN,KPROF,X(100),Y(100),PPES(100),UE(100),ME(100),PDTZ(100),
3VOCR(100),TWL(100),ETA(100)
COMMON/C/XOM(100),YOJ(100),S(100),SOL(100),AF(100),TSE(100),
1TAWL(100),TAWT(100),TPAP(100),RW(100),SW(100),SUTHL(100),
2RHSW(100),RHSF(100),HDADW(100),HEADE(100),NUW(100),MUBAP(100),
PRINT LOCATIONS OF INSTABILITY, TRANSITION, AND SEPARATION

IF (KMAIN.NE.1) GO TO 60
WRITE(6,1000)
IF(INST.EQ.0) GO TO 10
WRITE(6,1010) INST
GO TO 20
10 WRITE(6,1020)
20 IF (ITRAN.LE.1) GO TO 30
WRITE(6,1030) ITRAN
GO TO 40
30 WRITE(6,1040)
40 IF (ISEP.EQ.0) GO TO 50
WRITE(6,1050) ISEP
GO TO 60
50 WRITE(6,1060)

PRINT LOCATIONS OF LAMINAR AND TURBULENT BOUNDARY LAYERS

60 IEND = ITRAN-1
   IF (IEND.EQ.-1.OR.IEND.EQ.0) IEND=ISEP
   IF (IEND.EQ.0) IEND=NST
   IF (KMAIN.NE.1) GO TO 70
   IF (ITRAN.EQ.1) WRITE(6,1070)
   IF (ITRAN.NE.1) WRITE(6,1080) IEND
   IF (ITRAN.EQ.0) WRITE(6,1090)
   IF (ITRAN.EQ.1) WRITE(6,1100) ITRAN,IEND
70 IF (ITRAN.LE.1) GO TO 80
   IEND = ISEP
   IF(IEND.EQ.0) IEND=NST
   IF (KMAIN.NE.1) GO TO 90
   WRITE(6,1100) ITRAN,IEND

80 IF (KODE.EQ.0) DELSRU = DELSR(IEND)/XMAX
   IF (KODE.EQ.1) DELSRL = DELSR(IEND)/XMAX
   WRITE(6,2000)
   2000 FORMAT (1I1,35X,29HDIMENSIONED BLADE COORDINATES,36X,24HANGLES OF
     *ROTATION (DEG)/37X,11HUNCORRECTED,14X,9HCORRECTED,15X,10HTRANSLAT
     $ED/1X,7STATION,12X,1HX,23X,1HY,21X,5HXCORR,18X,5HYTRAN)
   SIGN = +1.
   IF (KODE.EQ.0) SIGN = -1.
   GSTAR = GSTARI*XMAX/CSTAR
   DO 170 I=1,IEND
     YCORR = Y(I) + SIGN*AES(DELSR(I)/COS(ETA(I)))
   YTRAN = YCORR - GSTAR
   ETAD = 57.2957796*ETA(I)
   WRITE (6,2001) I,X(I),Y(I),YCORR,YTRAN,ETAD
   2001 FORMAT (2X,I3,9X,4(F9.5,15X),F9.4)
   170 CONTINUE
C PRINT CALCULATED BOUNDARY LAYER PARAMETERS

IF (KMAIN .NE. 1) GO TO 90
WRITE (6, 1110)
WRITE (6, 1120) (I, X(I), S(I), DFLSR(I), THET(I), DELTA(I), FORM(I),
  IFORM(I), I=1, IEND)
WRITE (6, 1130)
WRITE (6, 1140) (I, CF(I), TAUW(I), RTH(I), DTDY(I), NUSS(I), HTRAN(I),
  ICRN(I), I=1, IEND)

C COMPUTE BOUNDS ON VELOCITY PROFILES

90 IF (KRCF .NE. 1) RETURN
WRITE (6, 1150)
IF (ITFAN .NE. 0) GO TO 100
IL1 = 2
IT1 = 0
IT2 = 0
GO TO 110
100 IL1 = 2
IT1 = ITRAN
IT2 = IEND
IF (IT1 .EQ. 1) IT1 = 2

C CALCULATE AND PRINT LAMINAR BOUNDARY LAYER VELOCITY PROFILES

110 NVP1 = NVP+1
IF (IL2 .LT. IL1) GO TO 140
DO 130 I=IL1, IL2
WRITE (6, 1160) I
AAA = 2.*SHAPL(I)/6.
BBB = -.5*SHAPL(I)
CCC = -2.+.5*SHAPL(I)
DDD = 1.-SHAPL(I)/6.
DEL = DELTA(I)/FLOAT(NVP)
YP = -DEL
DO 120 J=1, NVP1
YP = YP+DEL
ETAA = YP/DELTA(I)
YXMAX = YP/X(NST)
UUE = ((DDD*ETAA+CCC)*ETAA+BBB)*ETAA+AAA)*ETAA
U = UUE*UE(I)
120 WRITE (6, 1180) ETAA, YP, YXMAX, U, UUE
130 CONTINUE

C CALCULATE AND PRINT TURBULENT BOUNDARY LAYER VELOCITY PROFILES

140 IF (IT1 .EQ. 0) RETURN
DO 160 I=IT1, IT2
POWER = DELTA(I)/DELSR(I)-1.
WRITE (6, 1170) I, POWER
DEL = DELTA(I)/FLOAT(NVP)
YP = -DEL
DO 150 J=1, NVP1
YP = YP+DEL
ETAA = YP/DELTA(I)
150 CONTINUE
\text{YXMAX} = \text{YP/XMAX}

\text{UUE} = \text{ETA}^{**} (1./\text{POWER})

\text{U} = \text{UUE*UE}(I)

\text{150 WRITE} (6, 1180) \text{ETA,YP,YXMAX,U,UUE}

\text{160 CONTINUE}

\text{RETURN}

\text{C FORMAT STATEMENTS}

\text{1000 FORMAT}(1\text{H1}//1\text{X}, 36\text{HPRINCIPAL BOUNDARY LAYER INFORMATION}///)

\text{1010 FORMAT}(//1\text{X}, 31\text{HINSTABILITY OCCURS AT STATION },1\text{I3})

\text{1020 FORMAT}(//1\text{X}, 26\text{HINSTABILITY DOES NOT OCCUR})

\text{1030 FORMAT}(//1\text{X}, 30\text{HTRANSITION OCCURS AT STATION },1\text{I3})

\text{1040 FORMAT}(//1\text{X}, 25\text{HTRANSITION DOES NOT OCCUR})

\text{1050 FORMAT}(//1\text{X}, 30\text{HSEPARATION OCCURS AT STATION },1\text{I3})

\text{1060 FORMAT}(//1\text{X}, 25\text{HSEPARATION DOES NOT OCCUR})

\text{1070 FORMAT}(//1\text{X}, 37\text{HLAMINAR BOUNDARY LAYER DOES NOT OCCUR})

\text{1080 FORMAT}(//1\text{X}, 42\text{HLAMINAR BOUNDARY LAYER - STATIONS 1 TO },1\text{I3})

\text{1090 FORMAT}(//1\text{X}, 39\text{HTURBULENT BOUNDARY LAYER DOES NOT OCCUR///})

\text{1100 FORMAT}(//1\text{X}, 35\text{HTURBULENT BOUNDARY LAYER - STATIONS,2X,}\n\text{113.6H TO },1\text{I3///})

\text{1110 FORMAT}(//1\text{X}, 7\text{HSTATION,AX,1HX,12X,1HS,12X,5HDDELSP,10X,4HTHET,11X,}15\text{HDELTA,11X,4HFORM,10X,5HFORM})

\text{1120 FORMAT}(2\text{X,I3,3X,2F13.6,F14.6,1X,F14.6,1X,F14.6,1X,F14.4})

\text{1130 FORMAT}(//1\text{X}, 7\text{HSTATION,6X,2HCP,13X,4HTAUW,11X,3HRTH,14X,4HDTDY,}113X,4HNUSS,10X,5HHTRN,12X,3HCEN)


\text{1150 FORMAT}(1\text{H1}//1\text{X}, 17\text{HVELLOCITY PROFILES///})

\text{1160 FORMAT}(//1\text{X}, 7\text{HSTATION,1X,15,2X,7HPROFILE/3X,7HY/Delta,9X,}11HY,12X,6HY/XMAX,10X,1HU,12X,4HY/UE)

\text{1170 FORMAT}(//1\text{X}, 7\text{HSTATION,1X,15,2X,7HPROFILE,28X,2HNN=,1X,F6.2/3X,7HY/D1ELTA,9X,1HY,12X,6HY/XMAX,10X,1HU,12X,4HY/UE})

\text{1180 FORMAT}(1\text{X}, F8.4, 2X, 2G15.6, 2X, F9.2, 6X, F8.4)

\text{END}

\$IEPTC RUNKUT LIST, DECK

\text{SUBROUTINE RUNKUT}

\text{C RUNKUT SOLVES SIMULTANEOUS FIRST ORDINARY INITIAL VALUE}

\text{ORDINARY DIFFERENTIAL EQUATIONS}

\text{COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTUPR,KPVK,KEM,KSMT,1KSPAN,KLE,KATCH,CHTET,DLAM,TLMAT,DTUPR,TPURB,KPRE,KRAD,KSDE,KLAM,2KMAIN,KPROF,}X(100),Y(100),PPRS(100),ME(100),POPTZ(100),3VOVC(100),TWL(100),ETA(100)

\text{COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AP(100),TSE(100),}1TAHL(100),TAW(100),TBAR(100),MW(100),SW(100),SUTHL(100),2RHSW(100),RSEE(100),HEA(100),HEADE(100),NWW(100),MUBAR(100),3AA(100),BB(100),FF(100),DUDS(100),DMS(100),DMDL(100)

\text{COMMON/C6/FTTRAN,FORMS}

\text{COMMON/C7/INST,ITRAN,ISEP}

\text{COMMON/C8/YTAB(505),YTAB1(505),YTAB2(505),N1N}

\text{DIMENSION YY(2),YY(2),YINC(2),DCT(2),DIYK(2,4)}

\text{DOUBLE PRECISION XX,RY,RY,RUK,DEL,DOT,ITEM1,ITEM2,ITEM3,ITEM4,ITEM5,ITEM6}

\text{REAL ME,NWW,MUBAR}

\text{C SET DEL SPACING AND STORE INITIAL VALUES}

45
DEL = 0.002*S(NST)

10 YY (1) = FTRAN
YY (2) = FORMS
XX = S(ITRAN)
NV = 2
NTAB = 1
YTAB1 (1) = YY (1)
YTAB2 (1) = YY (2)
XTAB (1) = XX

SOLVE FOR YY(1) AND YY(2) AT NEXT XX INCREMENT

SAVE PREVIOUS YY(1) AND YY(2)

DO 30 J = 1, NV
30 RY (J) = YY (J)
RX = XX

CALCULATE NEW YY(1) AND YY(2)

DO 90 L = 1, 4
CALL LGRNGE (S, RE, NST, XX, ANSl)
CALL LGRNGE (S, SW, NST, XX, ANS2)
CALL LGRNGE (S, AA, NST, XX, ANS3)
CALL LGRNGE (S, BB, NST, XX, ANS4)
CALL LGRNGE (S, DMDS, NST, XX, ANS5)
CALL LGRNGE (S, TBAR, NST, XX, ANS6)
TEMP1 = 1. + (1. + ANS2) * YY (2)
TEMP2 = .123*EXP (-1.561*YY (2)) * ANS3
DOT (1) = 1.268 * (-YY (1) / ANS1 * ANS5 * TEM1 + TEM2)
TEMP3 = YY (2) * (YY (2) + 1.) ** 2 * (YY (2) - 1.)
TEMP4 = 1. + ANS2 * (YY (2) * YY (2) + 1.) * YY (2) - 1.) / ((YY (2) + 1.) * (YY (2) + 3.))
TEMP5 = (YY (2) * YY (2) - 1.) / YY (1) ** (2.7886) * (.011 * (YY (2) + 1.) * (YY (2) - 1.)
1**2 / YY (2) ** 2 * TTZ / ANS6) * ANS4
DOT (2) = -ANS5 * .5 / ANS1 * TEM3 * TEM4 + TEM5 - TEM6

APPLY THE RUNGA-KUTTA SCHEM

DO 40 J = 1, NV
40 RUK (J, L) = DEL * DOT (J)
GO TO (50, 50, 70, 90), L
50 DO 60 J = 1, NV
60 YY (J) = RY (J) + RUK (J, L) / 2.
XX = RX + DEL / 2.
GO TO 90
70 DO 80 J = 1, NV
80 YY (J) = RY (J) + RUK (J, L)
XX = RX + DEL
GO TO 90
CONTINUE

INCREMENT THE DEPENDENT VARIABLES TO OBTAIN NEW YY(1) AND YY(2)

DO 100 J = 1, NV
YINC (J) = (RUK (J, 1) + 2. * RUK (J, 2) + 2. * RUK (J, 3) + RUK (J, 4)) / 6.
100 YY (J) = RY (J) + YINC (J)

STORE NEW COMPUTED VALUES IN A TABLE
C
NTAB = NTAB + 1
YTAE1(NTAE) = YY(1)
YTAE2(NTAE) = YY(2)
XTAE(NTAE) = XX
IF (YTAE2(NTAB) .GT. 2.8) RETURN
IF (XX .LT. 10) GO TO 20
RETURN
END
$IBPTC SPLIN DECK
SUBROUTINE SPLINE(X,Y,N,DYDX,D2YDX2)
C
C SPLINE FITS A SPLINE CURVE TO X AND Y
C AND CALCULATES FIRST AND SECOND DERIVATIVES AT THE SPLINE POINTS
C END POINT SECOND DERIVATIVES EQUAL THOSE AT ADJACENT POINTS
C
DIMENSION X(N),Y(N),DYDX(N),D2YDX2(N)
DIMENSION G(100),H(100)
G(1) = -1.
H(1) = 0.
N1 = N - 1
IF (N1 .LT. 2) GO TO 20
DO 10 I = 2, N1
A = (X(I) - X(I-1)) / 6.
B = (X(I+1) - X(I)) / 6.
C = 2. * (A + B) - A * G(I-1)
D = (Y(I+1) - Y(I)) / (X(I+1) - X(I)) - (Y(I) - Y(I-1)) / (X(I) - X(I-1))
G(I) = B / C
10 H(I) = (D - A * H(I-1)) / C
20 D2YDX2(N) = H(N1) / (1. + G(N1))
DO 30 I = 2, N
K = N + 1 - I
30 D2YDX2(K) = H(K) - G(K) * D2YDX2(K + 1)
DYDX(1) = (X(1) - X(2)) / 6. * (2. * D2YDX2(1) + D2YDX2(2)) + (Y(2) - Y(1)) / (X(2) - X(1))
DO 40 I = 2, N
40 DYDX(I) = (X(I) - X(I-1)) / 6. * (2. * D2YDX2(I) + D2YDX2(I-1)) + (Y(I) - Y(I-1)) / (X(I) - X(I-1))
RETURN
END
$IBPTC LGRNGV DECK
SUBROUTINE LGRNGE(X,Y,N,ARG,ANS)
C
LINEAR INTERPOLATION (REPLACES 4-POINT LAGRANGE)
C
DIMENSION X(N), Y(N)

IF (ARG .GE. 0.80 * X(1)) GO TO 2
WRITE (6, 100) ARG, X(1), X(N)
100 FORMAT (9HO (LGRNGE), 5X, 10HABSCISSA =.E13.5, 27H IS OUT OF RANGE -
* X(1) =.E13.5, 5X, 6HX(N) =.E13.5/15X, 27H EXTRAPOLATED VALUE RETURNED 
*)
2 NM = N - 1
DO 3 I = 2, NM
IF (ARG .GT. X(I)) GO TO 3
M = I
GO TO 4
3 CONTINUE
IF (ARG .LE. 1.20*X(N)) GO TO 5
WRITE (6,100) ARG,X(I),X(N)
5 M = N
4 ANS = Y(M) + (Y(M) - Y(M-1))/(X(M) - X(M-1))*(ARG - X(M))
RETURN

END

$IEPTC SIMP DECK
FUNCTION SIMPS1(X1,X2,FUNC,KSIG)
DIMENSION V(200),H(200),A(200),B(200),C(200),P(200),E(200)
LOGICAL SPILL
DOUBLE PRECISION ANS,Q
DATA TW2,THREE,FOUR,THIRTY/2.0,3.0,4.0,30.0/
DATA T,NMAX,NSIG/3.0E-5,200,1/
C INITIALIZE FIRST ELEMENTS OF ARRAYS.
V=X1
H=(X2-V)/TW2
A=FUNC(V)
B=FUNC(V+H)
C=FUNC(X2)
P=H*(A+FOUR*B+C)
E=P
ANS=P
N=1
FRAC=T
SPILL=.FALSE.
10 TEST=ABS(FRAC*ANS)
K=N
DO 30 I=1,K
C TEST MAGNITUDE OF 4TH ORDER ERROR IN THIS INTERVAL.
IF (ABS(E(I)).LT.TEST) GO TO 30
IF (N.LT.NMAX) GO TO 20
C GO TO FINISH IF STORAGE IS FILLED UP.
SPILL=.TRUE.
KSIG=KSIG+NSIG
GO TO 40
C SUBDIVIDE INTERVAL AGAIN TO REDUCE 4TH ORDER ERROR.
20 N=N+1
V(N)=V(I)+H(I)
H(N)=H(I)/TW2
A(N)=E(I)
B(N)=FUNC(V(N)+H(N))
C(N)=C(I)
P(N)=H(N)*(A(N)+FOUR*B(N)+C(N))
H(I)=H(N)
B(I)=FUNC(V(I)+H(I))
C(I)=A(N)
Q=P(I)
P(I)=H(I)*(A(I)+FOUR*B(I)+C(I))
Q=P(I)+P(N)-Q
ANS=ANS+Q
E(I)=0
E(N)=0
30 CONTINUE
C TEST ALL INTERVALS AGAIN IF ANY WERE SUBDIVIDED THE LAST TIME.
IF (N.GT.K) GO TO 10
40 Q=0.0
DO 50 I=1,N
50 Q=Q+E(I)
C TIGHTEN ERROR LIMIT IF TOTAL ACCUMULATED ERROR TOO LARGE.
    IF (ABS(Q/T).LE.ABS(ANS).OR.SPLIL) GO TO 60
    FRAC=FRAC/TWO
    GO TO 10
C FINISH CALCULATION.
60 SIMPS1=(ANS+Q/THIRTY)/THREE
    RETURN
END
$IEPTC CURVF   DECK
    SUBROUTINE CURVF(COEF,ANS,X,Y,NX,NY)
C C EVALUATE THE POLYNOMIAL FUNCTION, ANS=F(X,Y), USING COEFFICIENTS, COEF
C
    DIMENSION COEF(20)
    NX1 = NX+1
    NY1 = NY+1
    ANS = COEF(1)
    IF (X.EQ..O.AND.Y.EQ..O) RETURN
    IF (Y.EQ..O) GO TO 10
    IF (X.EQ..O) GO TO 30
    GO TO 50
10 DO 20 I=2,NX1
20 ANS = ANS+COEF(I)*X**(I-1)
    RETURN
30 DO 40 I=2,NY1
40 K = ((I-1)*NX1+1
40 ANS = ANS+COEF(K)*Y**(I-1)
    RETURN
50 ANS = .0
    DO 60 I=1,NY1
    DO 60 J=1,NX1
    K = ((I-1)*NX1+J
    60 ANS = ANS+COEF(K)*Y**(I-1)*X**(J-1)
    RETURN
END
$IEPTC AFTMIX LIST,DECK
    SUBROUTINE AFTMIX (ALPH1,DELS,DELP,THETS,THFTP,TE,SP,XMFS1)
C
    DIMENSION XXX(100),YYY(100),SSS(100)

    COMMON/GAMP/GMP1,GMM1
    COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVE,NTURB,KPVM,KEM,KSMT,TH,
       KSPLN,KLE,KATCH,CTHET,DLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAN,
       2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
       3VOVCR(100),TWAL(100),ETA(100)
    COMMON/C2/XOM(100),YOM(100),S(100),SOLE(100),AE(100),TSE(100),
       1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
       2RHSW(100),RISE(100),HEADW(100),HEADF(100),NWW(100),MUWBAR(100),
       3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
    COMMON/C3/THET(100),DELSR(100),DELTA(100),FSPM(100),
       1FOMI(100),FORMP(100),RTH(100),RTHI(100),C(100),
       1TAWM(100),NUSS(100),DTDY(100),HTRAN(100),CRN(100),
       EQUIVALENCE (X,XXX), (Y,YYY), (S,SSS), (NST,N)
    REAL ME

    KODE = 0
    WRITE(6,5)
5 FORMAT(1H1,20X,22HAFTERMIXING PROPERTIES)
    XALPH1 = ALPH1
    ALPH1 = ALPH1*0.017453

2 \( VVCR_1 = \text{SQRT}( \frac{GMPl/2 \cdot XMFS1**2}{1 + GM1/2 \cdot XMFS1**2}) \)

\[
XX = SP \cdot \cos(\alpha_1)
\]

\[
DELSRT = \frac{(DELS + DELP)/XX}{1 + SQF(T \cdot (GMP1/2, XMFS1**2) / (XX)}
\]

\[
\theta = \frac{\theta_1 - \theta_2}{XX}
\]

\[
A = 1.0 - DELSRT - DTE - \theta
\]

\[
A1 = 1.0 - DELSRT - DTE
\]

IF \((A \leq 0.0)\) GO TO 16

IF \((\text{KODE} = 0)\) WRITE \((6,1)\)

1 FORMAT \((1H0, 10X, 36H\text{ROTOR WITH NO BOUNDARY LAYER CORRECTION})\)

\[
C = \left( (1 - 1.0) \cdot GM1 \cdot VVCR1 \cdot (\cos(\alpha_1) \cdot A1) \right) \cdot VVCR1**2
\]

\[
D = VVCR1 \cdot \sin(\alpha_1) / \left( \frac{A1}{VCR1} \right)
\]

\[
XX = SP \cdot \cos(\alpha_1)
\]

\[
DELSRT = \frac{(DELS + DELP)}{XX}
\]

\[
\theta = \frac{\theta_1 - \theta_2}{XX}
\]

\[
A = 1.0 - DELSRT - DTE - \theta
\]

\[
A1 = 1.0 - DELSRT - DTE
\]

IF \((A \leq 0.0)\) GO TO 16

IF \((\text{KODE} = 0)\) WRITE \((6,1)\)

1 FORMAT \((1H0, 10X, 36H\text{ROTOR WITH NO BOUNDARY LAYER CORRECTION})\)

\[
C = \left( (1 - 1.0) \cdot GM1 \cdot VVCR1 \cdot (\cos(\alpha_1) \cdot A1) \right) \cdot VVCR1**2
\]

\[
D = VVCR1 \cdot \sin(\alpha_1) / \left( \frac{A1}{VCR1} \right)
\]

\[
XX = SP \cdot \cos(\alpha_1)
\]

\[
DELSRT = \frac{(DELS + DELP)}{XX}
\]

\[
\theta = \frac{\theta_1 - \theta_2}{XX}
\]

\[
A = 1.0 - DELSRT - DTE - \theta
\]

\[
A1 = 1.0 - DELSRT - DTE
\]

IF \((A \leq 0.0)\) GO TO 16

IF \((\text{KODE} = 2)\) RETURN

IF \((\text{KODE} = 1)\) GO TO 3

END

50
$\text{IEFTC FUNC DECK}$

**SUBROUTINE FUNCT (XX, FX, DFX, INF)**

```
COMMON/GAMEY/GH51,GM1
COMMON/C1/GAM,B,PKZ,TZ,TUMACH,NST,NEF,NTURB,PKV,KEM,KSMTK,
           KSPLN,KLE,KATCH,CMET,SLAM,NTURE,TTURB,PKAD,YSDE,KLAM,
           2MAIN,PK0F,XX(100),Y(100),PRES(100),PE(100),PEPTZ(100),
           3VOCR(100),TAWL(100),ETA(100)
COMMON/C2/PSZ,TSZ,TSZ,PTZ,MSZ,MUTZ,NUSZ,NUTZ,CP,
           TPR,TC,ARCL
COMMON/C3/XCM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
           1TAWL(100),TAWT(100),TBAP(100),RM(100),SW(100),SUTHL(100),
           2RHSP(100),RHSF(100),HEADW(100),HEADC(100),N&W(100),M&TAR(100),
           3AA(100),BB(100),PF(100),DUDS(100),DMDS(100),DMDL(100)

REAL MUSZ,NUSZ,MUTZ,NUTZ,MF,NUW,MUPAR

INF = 0
B1 = 1. + GM1/2.*ME(1)**2
B2 = 1. + (2.70 - 1.7H**P)**.5*((1.+SW(1))*B1-1.)
B3 = -NUTZ*SUTHL(1)*ARCL/ATZ/DMDL(1)*EM**((3.-GM)/2.*G)
B4 = -1.138*B2
B5 = 2.3811*B2*(4.65*PR**(-1./3.) - 3.69*PR**.5)*PR**.5*(B1-1.)
FX = (B3*XX)**.5*(B4*XX+B5)+B4*(B3*XX)**.5
       RETURN
10 INF = 1
DFX = 1.E10
       RETURN
END
```

$\text{IEFTC ROTB DECK}$

**SUBROUTINE ROTB (A, B, Y, FUNCT, TCLPEY, Y, DFX)**

```
C
C ROOT FINDS A ROOT FOR (FUNCT-Y) IN THE INTERVAL (A, B)
C
X1 = A
X2 = B
CALL FUNCT(X1,FX1,DFX,INF)
10 DO 30 I=1,20
    X = (X1+X2)/2.
    CALL FUNCT(X,FX,DFX,INF)
    IF ((FX1-Y)*(FX-Y).GT.0.) GO TO 20
    X2 = X
    GO TO 30
20 X1 = X
    FX1 = FX
30 CONTINUE
    IF (ABS(Y-FX).LT.TOLERY) RETURN
    WRITE(6,1900) A,B,Y
    STOP
1900 FORMAT(///4X,G9.4) "ROOT HAS FAILED TO CONVERGE IN THE GIVEN INTERVAL"
1/4X,3HA =,G14.6,10X,3HB =,G14.6,10X,3BY =,G14.6
END
```
$IEFTC INTG1 DECK
REAL FUNCTION INT1(XX)

COMMON/GAMMP,GM1
COMMON/C1/GAM, R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPV,M,KEM,KSMTH,
1KSPRN,KLE,KATCH,CTHI,T,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
2KMAIN,KPROF,X(100),Y(100),PPFS(100),UE(100),ME(100),POPTZ(100),
3VOVCR(100),TWAL(100),ETA(100)
COMMON/C3/XCM(100),YCM(100),S(100),SOL(100),AF(100),TSE(100),
1TAWL(100),TAWT(100),TBAR(100),PA(100),SW(100),SUTHL(100),
2PHSW(100),RHS(100),HEDW(100),HEAE(100),NUW(100),MUBAR(100),
3AA(100),EB(100),TP(100),MDS(100),MDM(100),MDL(100)
COMMON/C5/SHAPL(100),SHAPK(100),B,NS

REAL ME,NUW,MUBAR,INT1

CALL LGVRGE(SOL,ME,NST,XX,ANS)
INT1 = ANS**(F-1)/(1+GMM1/2.*ANS**2)**
1((3.*GAM-1.)/(2.*GMM1))
RETURN

$IEFTC INTG2 DECK
REAL FUNCTION INT2(XX)

COMMON/C1/GAM, R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPV,M,KEM,KSMTH,
1KSPRN,KLE,KATCH,CTHI,T,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
2KMAIN,KPROF,X(100),Y(100),PPFS(100),UE(100),ME(100),POPTZ(100),
3VOVCR(100),TWAL(100),ETA(100)
COMMON/C3/XCM(100),YCM(100),S(100),SOL(100),AF(100),TSE(100),
1TAWL(100),TAWT(100),TBAR(100),PA(100),SW(100),SUTHL(100),
2PHSW(100),RHS(100),HEDW(100),HEAE(100),NUW(100),MUBAR(100),
3AA(100),EB(100),TP(100),MDS(100),MDM(100),MDL(100)
COMMON/C5/SHAPL(100),SHAPK(100),B,NS

REAL ME,NUW,MUBAR,INT2

IF (NS.LT.4) GO TO 10
CALL LGVRGE(SOL,SHAPK,NS,XX,INT2)
RETURN

10 DO 20 J=2,NS
   IF (SOL(J).LT.XX) GO TO 20
   INT2 = SHAPK(J-1) + (SHAPK(J) - SHAPK(J-1)) * (XX - SOL(J-1))/(SOL(J) - SOL(J-1))
   RETURN
20 CONTINUE
RETURN
END

Lewis Research Center,
National Aeronautics and Space Administration,
Cleveland, Ohio, September 8, 1971,
113-34.
APPENDIX A

PROGRAM CHANGES FOR A GAS OTHER THAN AIR

The program gas properties are set up for air. This program can be easily changed so that it applies to gases other than air. The changes are all made in subroutine PRECAL.

The coefficients read in by the DATA statements of CMU, CPR, and CTC arrays must be changed. The equations for the curve fits have the following form:

\[
\frac{\mu}{\mu_{sl}} = a_1 + b_1 \left( \frac{T}{T_{sl}} \right) + c_1 \left( \frac{T}{T_{sl}} \right)^2 + d_1 \left( \frac{T}{T_{sl}} \right)^3 + e_1 \left( \frac{T}{T_{sl}} \right)^4
\]

\[
Pr = a_2 + b_2 \left( \frac{T}{T_{sl}} \right) + c_2 \left( \frac{T}{T_{sl}} \right)^2 + d_2 \left( \frac{T}{T_{sl}} \right)^3 + e_2 \left( \frac{T}{T_{sl}} \right)^4
\]

\[
\frac{k}{k_{sl}} = a_3 + b_3 \left( \frac{T}{T_{sl}} \right) + c_3 \left( \frac{T}{T_{sl}} \right)^2 + d_3 \left( \frac{T}{T_{sl}} \right)^3 + e_3 \left( \frac{T}{T_{sl}} \right)^4
\]

If the number of coefficients changes from five in any case, this must be reflected both in the DATA statements and later in the calls on CURVFT where CMU, CPR, and CTC are used. If the properties are put in a different form, these cards must be removed.

The sea-level reference values in U. S. customary and SI units of temperature (TSLE, TSLM), viscosity (MUSLE, MUSLM), and thermal conductivity (TCSLE, TCSLM) must be changed.

The value of Sutherland's constant (TCON) and the computation of \(k_{su}\) (SUTHL (1)) will have to be changed. A temperature-viscosity law of the following form was used:

\[
\frac{\mu}{\mu_o} = k_{su} \left( \frac{T}{T_o} \right)
\]

Where \(k_{su}\) for air was used:

\[
k_{su} = \left( \frac{T}{T_o} \right)^{1/2} \left( \frac{T_o + TCON}{T + TCON} \right)
\]

Where TCON is Sutherland's constant for air.
APPENDIX B

ADDITIONAL OUTPUT

This output is obtained when KPRE, KGRAD, KSDE, KLAM, and KPROF are set equal to 1.

The output corresponding to KPRE are the geometric variables:

- X  X-coordinate, m; ft
- Y  Y-coordinate, m; ft
- S  surface length, x, m; ft
- XOM ratio of X to C*
- YOM ratio of Y to C*
- SOL ratio of surface length to total arc length

The next part of the output gives the local speed of sound and several temperatures:

- AE  local free-stream speed of sound, m/sec; ft/sec
- TSE static temperature, K; °R
- TWAL wall temperature, K; °R
- TAWL laminar recovery temperature, K; °R
- TAWT turbulent recovery temperature, K; °R
- TBAR reference temperature, K; °R

The final part of this output gives:

- RW  Reynolds number at wall, RW = (UE)(S)/NUW
- SW  temperature function at wall
- SUTHL value of the coefficient in Sutherland's viscosity temperature
- RHSW static density based on the wall temperature, kg/m³; slug/ft³
- RHSE static density based on free-stream temperature, kg/m³; slug/ft³
- HEADW velocity head based on the density at wall, N/m²; lbf/ft²
- HEADE velocity head based on the free-stream density, N/m²; lbf/ft²
NUW  kinematic viscosity at wall, m²/sec; ft²/sec

MUBAR  dynamic viscosity based on reference temperature, (N)(sec)/m²;
        (lbf)(sec)/ft²

Output corresponding to KGRAD contains the three gradients of velocity and Mach number along the surface computed by finite difference methods:

DUDS  dUE/dx, sec⁻¹

DMDS  dME/dx, m⁻¹; ft⁻¹

DMDL  dME/dSOL

Output corresponding to KSDE contains the numerical solution of the laminar and turbulent differential boundary-layer equations. In the laminar case, the solution is the correlation number CORLN. In the turbulent case, the solution is the incompressible form factor FORMI, and a function F of the momentum thickness. These solutions are printed with respect to the surface length S.

Output corresponding to KLAM contains the variables used in the laminar subroutine to check for the position of instability and transition. The three variables, RTHI (increasing from station to station) and RCRIT and RTRAN (decreasing from station to station), are used in this analysis. When RTHI becomes larger than RCRIT, instability has occurred. When RTHI becomes larger than RTRAN, transition is assumed to occur. The variables listed are

CORN  correlation number

SHEAR  shear parameter

DTH  ratio of transformed displacement thickness to transformed momentum thickness

FORMTR  transformed form factor

SHAPL  Pohlhausen shape factor based on boundary-layer thickness

RTHI  incompressible momentum-thickness Reynolds number

SHAPK  dimensionless shape factor based on momentum thickness

RCRIT  critical incompressible momentum-thickness Reynolds number

KBAR  mean shape factor based on momentum thickness

DIFF  difference between transition and instability momentum-thickness Reynolds numbers

RTRAN  incompressible momentum-thickness Reynolds number used in checking for transition point
Output corresponding to KPROF contains the velocity profiles at each station along the surface. The output listed is:

- **Y/DELTA**: ratio of distance normal to surface in y-direction in boundary-layer profile to boundary-layer thickness.
- **Y**: distance normal to surface in y-direction in boundary-layer profile, m; ft
- **Y/XMAX**: ratio of Y to XMAX.
- **U**: velocity within boundary layer, m/sec; ft/sec.
- **U/UE**: ratio of U of free-stream velocity.
REFERENCES


"The aeronautical and space activities of the United States shall be conducted so as to contribute ... to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—National Aeronautics and Space Act of 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D.C. 20546